

MEDIA INTERAKSI ANTARA MANUSIA DENGAN PERANGKAT KERAS MELALUI KINECT MENGGUNAKAN FUZZY SUGENO

Achmad Teguh Wibowo¹⁾

Tantri Windarti²⁾

1) Program Studi/Jurusan Sistem Informasi STIKOM Surabaya, email : atw@stikom.edu

2) Program Studi/Jurusan Sistem Informasi STIKOM Surabaya, email : tantri@stikom.edu

Abstract: Saat ini masih banyak ditemui interaksi manusia dengan komputer yang hanya menggunakan *keyboard* dan *mouse* saja, sehingga banyak peneliti berkeinginan membuat interaksi manusia dengan komputer yang lebih hidup dan menyenangkan untuk dilakukan. *Gesture processing* merupakan salah satu jawaban atas keinginan para peneliti untuk memberikan pengalaman baru dalam hal interaksi manusia dengan komputer. Membahas *gesture* tidak akan lengkap tanpa membahas Natural User Interface (NUI). NUI adalah istilah umum untuk beberapa teknologi seperti *speech recognition*, *multitouch* dan *kinectic interface* seperti *kinect*. Teknologi ini memunculkan ciri lain dari NUI seperti interaksi antar user dan komputer akan terjadi tanpa perantara (media interaksi tidak akan terlihat).

Penggabungan dari teknologi *kinect* dan *Fuzzy Sugeno* yang diproses oleh komputer dapat menjadi solusi dari media interaksi manusia dengan komputer yang lebih menarik dan dapat mengurangi penggunaan *keyboard* atau *mouse*, teknologi ini dapat membaca gerakan tubuh manusia yang disorot oleh kamera *kinect* menjadi kode-kode perintah yang bisa dimengerti komputer.

Keywords: *Natural User Interface (NUI), Kinect, Fuzzy Sugeno*

Gesture atau bahasa tubuh adalah gerakan tubuh secara spontan yang biasanya menyertai komunikasi verbal. Bagian tubuh yang umum digunakan adalah tangan, jari, lengan, kepala, wajah, mata, alis dan badan (Loehr, 2004).

Gesture processing merupakan salah satu jawaban atas keinginan para peneliti untuk memberikan pengalaman baru dalam hal interaksi manusia dengan komputer.

Dalam hal ini diperlukan suatu teknologi khusus yang mampu membaca dan menerjemahkan gerakan-gerakan yang dibuat oleh manusia menjadi suatu perintah yang dapat dibaca dan diproses oleh komputer. Teknologi ini biasa disebut *kinect*, alat ini merupakan proyek dari Microsoft dengan nama *Project Natal*, proyek ini bertugas untuk menciptakan alat yang mampu melakukan *depth recognition*, *motion tracking*, *facial recognition* dan *speech recognition*. Referensi alat ini dibuat oleh perusahaan PrimeSense yang terdiri dari sebuah kamera RGB, sebuah sensor infra merah dan sebuah *light source* infra merah (Webb and Ashley, 2012).

NUI adalah istilah umum untuk beberapa teknologi seperti *speech recognition*, *multitouch* dan

kinectic interface seperti *kinect*. NUI lebih unggul dari *graphical user interface* seperti interaksi menggunakan *keyboard* dan *mouse* yang umum digunakan di beberapa sistem operasi seperti windows, mac, linux dan lain-lain. NUI memunculkan ciri lain seperti interaksi antar *user* dan komputer akan terjadi tanpa perantara (media interaksi tidak akan terlihat) (Webb and Ashley, 2012).

Salah satu metode yang biasa digunakan adalah *fuzzy sugeno*. Metode ini merupakan metode inferensi *fuzzy* untuk aturan yang direpresentasikan dalam bentuk IF – THEN, dimana output (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linear. Metode ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985.

NUI dapat membaca gerakan tubuh manusia (*gesture*) yang disorot oleh kamera *kinect* menjadi kode-kode perintah yang bisa dimengerti computer, sedangkan penggunaan metode *fuzzy sugeno* digunakan untuk mencari nilai akhir yang berupa nilai konstanta sebagai *parameter* pemicu kerja perangkat keras. Penggunaan metode *fuzzy sugeno*

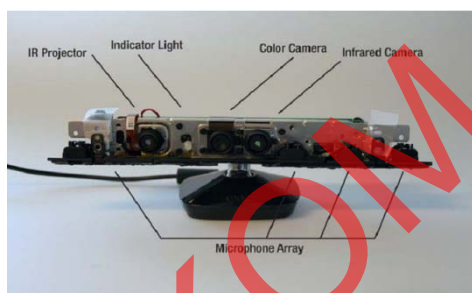
dapat meningkatkan kinerja aplikasi yang dibuat, karena metode ini mempunyai beberapa kelebihan diantaranya komputasinya lebih efisien dan bekerja paling baik untuk teknik linear.

Dengan adanya teknologi ini, interaksi antara manusia dan komputer dapat terjadi melalui *gesture*, sehingga dapat mempersingkat waktu dan lebih menyenangkan dalam hal pengendalian alat yang sudah disesuaikan dengan teknologi *kinect*. Sehingga prinsip dari HCI yaitu membuat agar sistem dapat berdialog dengan pengguna terjadi seramah dan seefisien mungkin dapat tercapai.

LANDASAN TEORI

A. Kinect

Kinect merupakan proyek dari Microsoft dengan nama *Project Natal*, proyek ini bertugas untuk menciptakan alat yang mampu melakukan *depth recognition, motion tracking, facial recognition dan speech recognition*. Struktur *kinect* tampak seperti pada Gambar 1. (Webb and Ashley, 2012).



Gambar 1. Struktur *Kinect* (Webb and Ashley, 2012).

Sensor *kinect* berbasis lensa optik dan memiliki beberapa keterbatasan, tetapi sensor *kinect* dapat bekerja baik di antara *range* nilai berikut (Catuhe, 2012) :

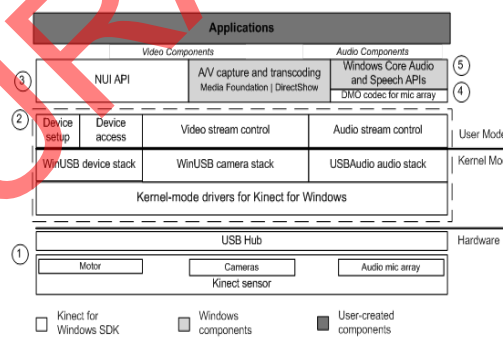
- *Horizontal viewing angle* 57°
- *Vertical viewing angle* 43°
- Jarak terbaik user dengan sensor *kinect* adalah 1.2 m
- *Depth range* 400 mm sampai 800 mm
- *Temperature* 5 sampai 35 derajat celcius

1. Kinect Windows Software Development Toolkit (SDK)

Untuk membuat aplikasi menggunakan *kinect* di sistem operasi windows, kita harus menginstall *kinect* SDK, salah satunya adalah *KinectSDK-v1.5*. SDK *kinect* sudah menyediakan *software library* dan *tools* untuk membantu membuat aplikasi menggunakan *kinect* sensor, *software library* dan *tools* pada *kinect* tampak pada Gambar 2. dan Gambar 3. (Wiranda, 2012).



Gambar 2. Gambaran Interaksi Sensor *Kinect* Dengan Aplikasi (Wiranda, 2012).



Gambar 3. Arsitektur SDK *Kinect* (Wiranda, 2012)

Kinect SDK terintegrasi dengan sistem operasi windows melalui komponen *standart* termasuk :

- *Audio, speech and media API* dapat digunakan dengan aplikasi seperti Microsoft Speech SDK.
- *DirectX Media Object (DMO)* dapat digunakan dengan aplikasi seperti *DirectShow* atau *Media Foundation*.

2. Video Stream

Seperti yang diketahui bahwa data pertama yang diberikan oleh *kinect* sensor adalah *video stream*. Meskipun *kinect* sensor berfungsi sebagai kamera 3D, pada tingkat yang paling dasar. Kamera

standart *kinect* dapat menangkap *video streams* menggunakan resolusi dan *frame rate* berikut ini :

- 640 x 480 sebanyak 30 fps menggunakan RGB format.
- 1280 x 960 sebanyak 12 fps menggunakan RGB format.
- 640 x 480 sebanyak 15 fps menggunakan YUV format.

3. Depth Camera

Seperti halnya *video stream*, *depth camera* juga dapat mengirim aliran data yang terdiri dari jarak antara kamera *kinect* dengan object terdekat yang ditemukan. Setiap pixel dari gambar yang dihasilkan mengandung nilai jarak yang dinyatakan dalam satuan millimeter (Webb and Ashley, 2012).

Resolusi yang dapat didukung oleh *depth camera* antara lain (640 x 480), (320 x 240) atau (80 x 60). Semua resolusi menggunakan *frame rate* sebesar 30 FPS. Dengan menggunakan *method DepthStream* kita dapat memilih resolusi yang disukai, gambar *DeptStream* tampak pada Gambar 4. (Webb and Ashley, 2012).

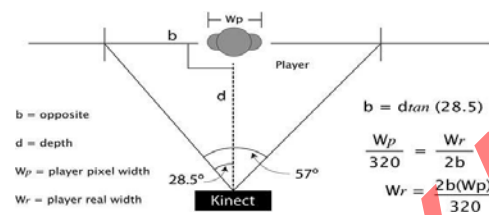


Gambar 4. Contoh *Capture* Gambar Menggunakan *Depth Camera* (Webb And Ashley, 2012)

4. Mengambil Ukuran

Salah satu hal yang menarik dari *kinect* adalah mengukur pixel dari pengguna, dimana posisi X dan Y dari pixel tidak berkoordinasi dengan lebar atau tinggi secara *actual*. Namun bisa saja kita menghitung pixel dari pengguna karena kita tahu bahwa sudut pandang kamera *kinect* sebesar 57 derajat horizontal dan 43 deratajat vertical, sehingga

kita bisa menentukan lebar dan tinggi pengguna menggunakan trigonometri, dimana cara mengitung lebar pengguna tampak pada Gambar 5. (Webb and Ashley, 2012).

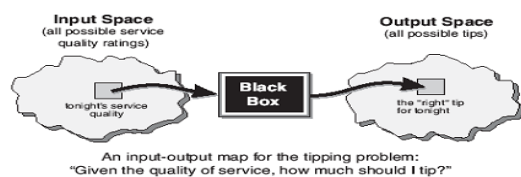


Gambar 5. Menentukan Lebar Pengguna (Webb And Ashley, 2012)

B. Logika Fuzzy

Orang yang belum pernah mengenal logika *fuzzy* pasti akan mengira bahwa logika *fuzzy* adalah sesuatu yang amat rumit dan tidak menyenangkan. Namun, sekali seseorang mulai mengenalnya, ia pasti akan sangat tertarik dan akan menjadi pendatang baru untuk ikut serta mempelajari logika *fuzzy*. Logika *fuzzy* dikatakan sebagai logika baru yang lama, sebab ilmu tentang logika *fuzzy* modern dan metodis baru ditemukan beberapa tahun yang lalu, padahal sebenarnya konsep tentang logika *fuzzy* itu sendiri sudah ada sejak lama (Kusumadewi, 2004).

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* ke dalam suatu ruang *output*. Contoh pemetaan *input-output* pada logika *fuzzy* tampak pada gambar 6.



Gambar 6. Contoh Pemetaan *Input-Output* (Kusumadewi, 2004)

1. Inferensi Fuzzy

Inferensi *Fuzzy* merupakan kerangka komputasi yang didasarkan pada teori himpunan *fuzzy*, aturan *fuzzy* berbentuk *if-then* dan penalaran *fuzzy*. Inferensi *fuzzy* telah berhasil diterapkan di bidang-bidang seperti kontrol otomatis, klasifikasi

data, analisis keputusan, dan sistem pakar. Sehingga dari penerapan yang ada dikenal beberapa istilah lain dalam inferensi *fuzzy* yaitu *fuzzy rule based*, sistem pakar *fuzzy*, pemodelan *fuzzy*, *fuzzy associative memory* dan pengendalian *fuzzy* (ketika digunakan pada proses kontrol) (Kusumadewi, 2004).

2. Metode Sugeno

Michio Sugeno mengusulkan penggunaan *singleton* sebagai fungsi keanggotaan dari konsekuen. *Singleton* adalah sebuah himpunan *fuzzy* dengan fungsi keanggotaan pada titik tertentu mempunyai sebuah nilai dan 0 di luar titik tersebut. Pada metode Sugeno dua bagian pertama dari proses penarikan kesimpulan *fuzzy*, *fuzzifikasi input* dan menerapkan operator *fuzzy* semua sama dengan metode Mamdani. Perbedaan utama antara metode Mamdani dan Sugeno adalah *output membership function* dari metode Sugeno berbentuk linier atau konstan (Kusumadewi, 2004). Aturan pada model *fuzzy sugeno* mempunyai bentuk :

**If input 1 = x and input 2 = y then Output is z
= ax + by + c**

Ada 2 model *fuzzy* dengan metode Sugeno yaitu sebagai berikut :

1. Model *fuzzy Sugeno* Orde-Nol

Secara umum bentuk model *fuzzy Sugeno* Orde-Nol adalah :

IF(x₁ is A₁) * (x₂ is A₂) * (x₃ is A₃) *.....* (x_N is A_N) THEN z = k

Dengan A_i adalah himpunan *fuzzy* ke – i sebagai antiseden, dan k adalah suatu konstanta (tegas) sebagai konsekuen.

2. Model *fuzzy Sugeno* Orde-Satu

Secara umum bentuk model *fuzzy Sugeno* Orde-Satu adalah :

IF(x₁ is A_{1i}) *.....* (x_N is A_{Ni}) THEN z = p₁ * x₁ +.....+ P_N * x_N + q

Proses penegasan (*defuzzy*) dilakukan dengan menggunakan konsep rata-rata tertimbang (*weighted average*), seperti terlihat pada persamaan di bawah (Kusumadewi, 2004).

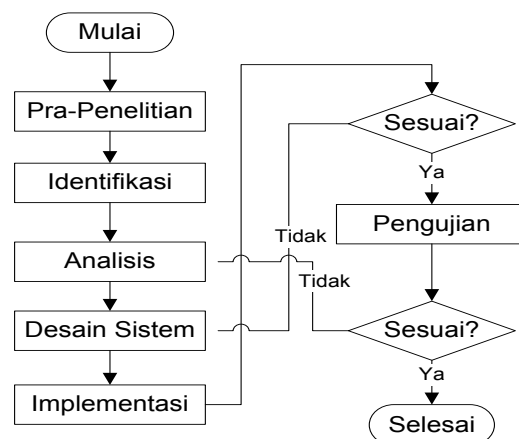
$$z = \frac{\sum_{r=1}^R (\alpha_r z_r)}{\sum_{r=1}^R \alpha_r}$$

Sistem *fuzzy Sugeno* juga memiliki kelemahan terutama pada bagian *THEN*, yaitu dengan adanya perhitungan matematika sehingga tidak dapat menyediakan kerangka alami untuk merepresentasikan pengetahuan manusia dengan sebenarnya. Permasalahan kedua adalah tidak adanya kebebasan untuk menggunakan prinsip yang berbeda dalam logika *fuzzy*, sehingga ketidakpastian dari sistem *fuzzy* tidak dapat direpresentasikan secara baik dalam kerangka ini (Kusumadewi, 2004).

METODOLOGI PENELITIAN

A. Metodologi Penelitian

Metodologi penelitian dalam penelitian media media interaksi antara manusia dengan perangkat keras melalui kinect menggunakan *fuzzy sugeno* tampak dalam gambar 7.



GamGambar 7. Metodologi Penelitian.

B. Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian ini adalah :

A. Kebutuhan Perangkat Lunak

Perkiraan kebutuhan perangkat lunak yang digunakan dalam penelitian ini agar dapat berjalan dengan baik :

1. Microsoft Visual Studio 2010 (C#).
2. Microsoft SDK *kinect* 1.5.
3. Sistem operasi windows 7 keatas

B. Kebutuhan perangkat keras

Perkiraan kebutuhan perangkat keras yang digunakan dalam penelitian ini agar dapat berjalan dengan baik :

1. *Memory* kapasitas 512 mb atau lebih.
2. *Hardisk* 20 gb atau lebih.
3. Kabel *Converter serial to usb*.
4. Mesin *Kinect*.
5. *Processor* intel Pentium IV dengan kecepatan 1.86 atau lebih.
6. *Mouse, keyboard* dan *monitor* dalam keadaan baik.
7. Perangkat keras yang terdiri dari rangkaian lampu led.

C. Pengumpulan Data

Untuk pengumpulan data yang diperlukan dalam pelaksanaan penelitian ini. Menggunakan cara:

- a) Studi literature : cara ini dilakukan dengan tujuan untuk memperoleh pengetahuan lebih dalam tentang pemrograman dengan *kinect* dimana metode ini dipilih untuk memberikan penyelesaian dalam penelitian ini.
- b) Observasi : cara ini dilakukan terhadap objek secara langsung guna mendapatkan informasi dasar terhadap objek yang diteliti.
- c) Analisis : semua data yang diperoleh melalui tahap studi literature dan observasi, dirancang menjadi inferensi *fuzzy*, desain *interface* serta melakukan akuisisi data informasi hasil pengumpulan data sebelumnya.

d) Desain sistem : merancang sistem secara keseluruhan mulai dari antar muka, pengolahan input dan menghasilkan output yang digunakan untuk memicu kerja perangkat keras.

e) Implementasi : memindahkan hasil rancangan pada tahap sebelumnya kedalam sistem komputerisasi. Pada bagian ini, membuat aplikasi hasil dari rancangan desain sistem yang dibuat.

f) Pengujian : dalam tahap ini dilakukan dengan beberapa tahap. Tahap pertama melakukan pengujian perangkat lunak sistem yang dibagi dalam 4 bagian, yaitu bagian jarak, kemiringan, pencahayaan tubuh dengan mesin *kinect* dan posisi tubuh memungungi mesin *kinect*.

D. Hipotesis

- Terdapat pengaruh nilai kemiringan tubuh terhadap mesin *kinect* yang diproses menggunakan *fuzzy Sugeno* terhadap aplikasi yang dibuat.
- Terdapat pengaruh nilai jarak telapak tangan kanan dan telapak tangan kiri terhadap mesin *kinect* yang diproses menggunakan *fuzzy Sugeno* terhadap aplikasi yang dibuat.
- Terdapat pengaruh pencahayaan ruangan terhadap aplikasi yang dibuat.

HASIL DAN PEMBAHASAN

A. Representasi Pengetahuan

Proses representasi pengetahuan dilakukan dengan cara mengumpulkan pengetahuan-pengetahuan pokok dan aturan-aturan yang digunakan dalam pemanfaatan *kinect* sebagai pemacu kerja perangkat keras yang selanjutnya disusun berdasarkan:

1. Posisi tubuh pengguna terhadap mesin *kinect*.
2. Jarak telapak tangan kiri dan kanan pengguna terhadap mesin *kinect*

Aturan-aturan yang digunakan dalam proses representasi pengetahuan direpresentasikan dalam bentuk grafik *fuzzy*. Terdapat 2 grafik *fuzzy* yang

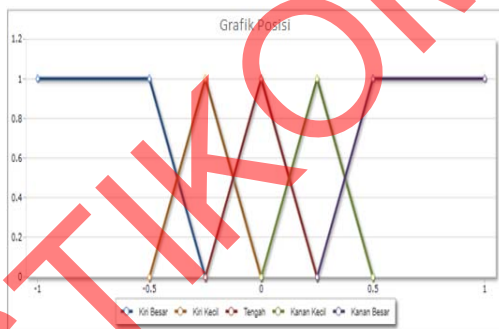
digunakan, yaitu grafik posisi dan grafik jarak telapak tangan. Kedua grafik tersebut mempunyai variabel yang berbeda, grafik posisi mempunyai 5 variabel yaitu:

1. Variabel kiri besar dimulai dari sumbu (-1, 1), (-0.5, 1), (-0.25, 0).
2. Variabel kiri kecil dimulai dari sumbu (-0.5, 0), (-0.25, 1), (0,0).
3. Variabel tengah dimulai dari sumbu (-0.25, 0), (0, 1), (0.25, 0).
4. Variabel kanan kecil dimulai dari sumbu (0, 0), (0.25, 1), (0.5, 0).
5. Variabel kanan besar dimulai dari sumbu (0.25, 0), (0.5, 1), (1,1).

Tabel 1. Nilai dan Variabel Grafik Posisi

| No | Variabel | X ₁ | Y ₁ | X ₂ | Y ₂ | X ₃ | Y ₃ |
|----|-------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | Kiri besar | -1 | 1 | -0.5 | 1 | -0.25 | 0 |
| 2 | Kiri kecil | -0.5 | 0 | -0.25 | 1 | 0 | 0 |
| 3 | Tengah | -0.25 | 0 | 0 | 1 | 0.25 | 0 |
| 4 | Kanan kecil | 0 | 0 | 0.25 | 1 | 0.5 | 0 |
| 5 | Kanan besar | 0.25 | 0 | 0.5 | 1 | 1 | 1 |

Nilai dari X_n dan Y_n dalam Tabel 1 dimasukkan ke dalam rumus fungsi keanggotaan kurva fuzzy trapesium, sehingga menghasilkan grafik fuzzy posisi yang diperlihatkan pada gambar 8.



Gambar 8. Grafik Fuzzy Posisi

Sedangkan grafik jarak telapak tangan mempunyai 3 variabel yaitu:

1. Variabel dekat dimulai dari sumbu (0, 1), (1, 1), (2,0).

2. Variabel sedang dimulai dari sumbu (1, 0), (2, 1), (3, 0).
3. Variabel jauh dimulai dari sumbu (2, 0), (3, 1), (4,1).

Tabel 2. Nilai dan Variabel Grafik Telapak Tangan

| No | Variabel | X ₁ | Y ₁ | X ₂ | Y ₂ | X ₃ | Y ₃ |
|----|----------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | Dekat | 0 | 1 | 1 | 1 | 2 | 0 |
| 2 | Sedang | 1 | 0 | 2 | 1 | 3 | 0 |
| 3 | Jauh | 2 | 0 | 3 | 1 | 4 | 1 |

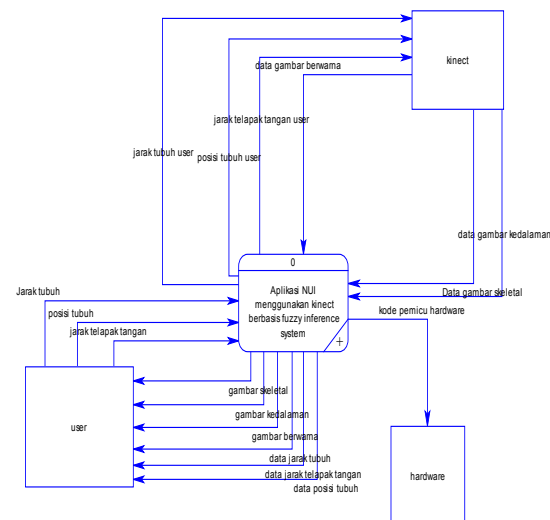
Nilai dari X_n dan Y_n dalam Tabel 2 dimasukkan ke dalam rumus fungsi keanggotaan kurva fuzzy trapesium, sehingga menghasilkan grafik fuzzy jarak telapak tangan yang diperlihatkan dalam Gambar 9.



Gambar 9. Grafik Fuzzy Jarak TelapakTangan

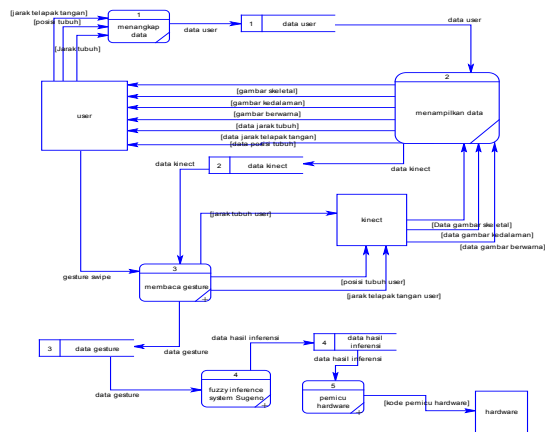
B. Data Flow Diagram (DFD)

1. Diagram Konteks



Gambar 10. Diagram Konteks

2. DFD Level 0



Gambar 11. DFD Level 0

C. Uji Coba

1. Pengujian Jarak Tubuh Terhadap Kinect

Tabel 3. Tabel Pengujian Jarak Tubuh Terhadap Kinect

| No | Jarak (meter) | Hasil Percobaan |
|----|---------------|--------------------------------|
| 1 | 0.79 m | Aplikasi tidak muncul di layar |
| 2 | 0.86 m | Aplikasi tidak muncul di layar |
| 3 | 0.98 m | Aplikasi muncul di layar |
| 4 | 1.14 m | Aplikasi muncul di layar |
| 5 | 1.53 m | Aplikasi muncul di layar |
| 6 | 2.08 m | Aplikasi muncul di layar |
| 7 | 2.17 m | Aplikasi muncul di layar |
| 8 | 2.25 m | Aplikasi muncul di layar |
| 9 | 2.32 m | Aplikasi muncul di layar |
| 10 | 2.43 m | Aplikasi muncul di layar |
| 11 | 2.47 m | Aplikasi muncul di layar |
| 12 | 2.54 m | Aplikasi muncul di layar |
| 13 | 2.68 m | Aplikasi muncul di layar |
| 14 | 2.73 m | Aplikasi muncul di layar |
| 15 | 2.77 m | Aplikasi muncul di layar |
| 16 | 2.88 m | Aplikasi muncul di layar |
| 17 | 3.50 m | Aplikasi muncul di layar |
| 18 | 3.68 m | Aplikasi muncul di layar |
| 19 | 3.83 m | Aplikasi tidak muncul di layar |
| 20 | 3.95 m | Aplikasi tidak muncul di layar |

2. Pengujian Posisi Tubuh Terhadap Kinect.

Tabel 4. Tabel Pengujian Posisi Tubuh Terhadap Kinect

| No | Posisi Tubuh (meter) | Keterangan | Hasil Percobaan |
|----|----------------------|---|--------------------------|
| 1 | 0.00 m | Tubuh tepat sejajar dengan kamera kinect | Aplikasi muncul di layar |
| 2 | -0.46 m | Tubuh condong ke sisi kiri dengan kamera kinect | Aplikasi muncul di layar |
| 3 | -0.70 m | Tubuh condong ke sisi kiri dengan kamera kinect | Aplikasi muncul di layar |

| | | | |
|----|---------|--|--------------------------------|
| 4 | -0.89 m | Tubuh condong ke sisi kiri dengan kamera kinect | Aplikasi muncul di layar |
| 5 | -1.15 m | Tubuh condong ke sisi kiri dengan kamera kinect | Aplikasi muncul di layar |
| 6 | -1.30 m | Tubuh condong ke sisi kiri dengan kamera kinect | Aplikasi muncul di layar |
| 7 | -1.37 m | Tubuh condong ke sisi kiri dengan kamera kinect | Aplikasi muncul di layar |
| 8 | -1.47 m | Tubuh condong ke sisi kiri dengan kamera kinect | Aplikasi tidak muncul di layar |
| 9 | 0.16 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 10 | 0.36 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 11 | 0.41 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 12 | 0.55 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 13 | 0.70 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 14 | 0.78 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 15 | 0.87 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 16 | 1.00 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 17 | 1.13 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 18 | 1.30 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 19 | 1.34 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi muncul di layar |
| 20 | 1.41 m | Tubuh condong ke sisi kanan dengan kamera kinect | Aplikasi tidak muncul di layar |

3. Pengujian Pencahayaan Ruang Terhadap Kinect

Tabel 5. Tabel Pengujian Pencahayaan Ruang
Terhadap Kinect

| No | Intensitas Cahaya (Cd) | Keterangan |
|----|------------------------|--------------------------|
| 1 | 0 | Aplikasi muncul di layar |
| 2 | 1 | Aplikasi muncul di layar |
| 3 | 2 | Aplikasi muncul di layar |
| 4 | 3 | Aplikasi muncul di layar |
| 5 | 4 | Aplikasi muncul di layar |
| 6 | 5 | Aplikasi muncul di layar |
| 7 | 6 | Aplikasi muncul di layar |
| 8 | 7 | Aplikasi muncul di layar |
| 9 | 8 | Aplikasi muncul di layar |
| 10 | 9 | Aplikasi muncul di layar |
| 11 | 10 | Aplikasi muncul di layar |
| 12 | 11 | Aplikasi muncul di layar |
| 13 | 12 | Aplikasi muncul di layar |
| 14 | 13 | Aplikasi muncul di layar |

KESIMPULAN DAN SARAN

A. Kesimpulan

Dari perancangan sistem, implementasi dan pengujian serta analisa, maka dapat diambil kesimpulan :

1. Kinect dapat menangkap jarak tubuh, data posisi, kedua telapak tangan serta mendeteksi *hand gesture (swipe)*.
2. Metode Fuzzy Sugeno bekerja sesuai dengan hasil yang diinginkan.
3. Jarak minimal antara kinect dengan tubuh adalah ± 0.84 m dan jarak maksimalnya adalah ± 3.83 m.

4. Posisi minimal antara kinect dengan tubuh adalah ± -1.47 m dan posisi maksimalnya adalah < 0.00 m sedangkan mendeteksi posisi tubuh condong ke kanan dengan baik posisi minimal antara kinect dengan tubuh adalah > 0.00 m dan posisi maksimalnya adalah ± 1.41 m.
5. Faktor cahaya tidak berpengaruh.

B. Saran

Berikut saran dalam penelitian ini :

1. Penelitian ini dapat ditingkatkan kebermanfaatannya, seperti menerapkan kinect untuk bidang sipil dan lain-lain.
2. Algoritma fuzzy sugeno dapat digabungkan/hybrid dengan algoritma lain untuk mendapatkan hasil yang lebih baik.

RUJUKAN

- Kusumadewi, S. 2004. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Graha Ilmu.
- Loehr, Daniel P. 2004. *Gesture and Intonation*. Disertasi tidak diterbitkan. Washington D.C : Georgetown University.
- Webb, J, and Ashley, J. 2012. *Begining Kinect Programming with the Microsoft KinectSDK*. Apress.
- Wiranda, N. 2012. *Implementasi Kinect Pada Penerjemahan Bahasa Isyarat*. Publikasi Sekolah Tinggi Manajemen Informatika Dan Komputer AMIKOM, Yogyakarta.