

**IMPLEMENTASI ALGORITMA *FUZZY TYPE-2* UNTUK
MENENTUKAN PERILAKU NPC DALAM GAME
*VIRTUAL REALITY SURVIVAL SHOOTER***

SKRIPSI

Oleh :
MUHAMMAD FAARIS SAIFUDDIEN
NIM : 11650090



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2018**

HALAMAN PENGAJUAN

**IMPLEMENTASI ALGORITMA *FUZZY TYPE-2* UNTUK
MENENTUKAN PERILAKU NPC DALAM *GAME*
*VIRTUAL REALITY SURVIVAL SHOOTER***

SKRIPSI

**Diajukan Kepada :
Fakultas Sains dan Teknologi
Universitas Islam Negeri
Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh :
MUHAMMAD FAARIS SAIFUDDIEN
NIM : 11650090**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2018**

HALAMAN PERSETUJUAN

**IMPLEMENTASI ALGORITMA FUZZY TYPE-2 UNTUK
MENENTUKAN PERILAKU NPC DALAM GAME
VIRTUAL REALITY SURVIVAL SHOOTER**

SKRIPSI

Oleh :
MUHAMMAD FAARIS SAIFUDDIEN
NIM : 11650090

Telah Diperiksa dan Disetujui untuk Diuji :
Tanggal : 30 November 2017

Pembimbing I

Fachrul Kurniawan, M.M.T.
NIP. 19771020 200901 1 001

Pembimbing II

Fresy Nugroho, M.T.
NIP. 19710722 201101 1 001

Mengetahui,
**Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang**



Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

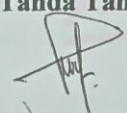
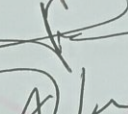
HALAMAN PENGESAHAN**IMPLEMENTASI ALGORITMA *FUZZY TYPE-2* UNTUK
MENENTUKAN PERILAKU NPC DALAM *GAME*
*VIRTUAL REALITY SURVIVAL SHOOTER*****SKRIPSI**

Oleh :
MUHAMMAD FAARIS SAIFUDDIEN
NIM : 11650090

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima Sebagai Salah Satu Persyaratan Untuk
Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal : 05 Januari 2018

Susunan Dewan Penguji**Tanda Tangan**

- | | | |
|-----------------------|--|---|
| 1. Penguji Utama | : <u>Yunifa Miftachul Arif, M.T.</u>
NIP. 19830616 201101 1 004 | () |
| 2. Ketua Penguji | : <u>Dr. Muhammad Faisal, M.T.</u>
NIP. 19740510 200501 1 007 | () |
| 3. Sekretaris Penguji | : <u>Fachrul Kurniawan, M.MT.</u>
NIP. 19771020 200901 1 001 | () |
| 4. Anggota Penguji | : <u>Fresy Nugroho, M.T.</u>
NIP. 19710722 201101 1 001 | () |

Mengetahui dan Mengesahkan
Ketua Jurusan Teknik Informatika



Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

**HALAMAN PERNYATAAN
ORISINALITAS PENELITIAN**

Nama : MUHAMMAD FAARIS SAIFUDDIEN

NIM : 11650090

Jurusan : Teknik Informatika

Fakultas : Sains dan Teknologi

Judul Skripsi : **IMPLEMENTASI ALGORITMA *FUZZY TYPE-2* UNTUK
MENENTUKAN PERILAKU NPC DALAM *GAME*
*VIRTUAL REALITY SURVIVAL SHOOTER***

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang,

Yang membuat pernyataan,



Muhammad Faaris Saifuddien
NIM : 11650090

MOTTO

*“Berusahalah untuk menjadi yang terbaik
saat ini maka kesuksesan akan menyusulmu
di hari esok”*



HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahim, kupersembahkan sebuah karya sederhana ini untuk orang-orang yang paling kusayangi, kubanggakan, dan selalu memberikan energi semangat untukku..

Seluruh keluarga besarku khususnya Ayah dan Ibu tercinta Subandriyo, Susanti Hari Pratiwi, serta Adik-Adikku Tersayang Qonita Sabrina, Amalina Syadza Nabiela, dan Muhammad Zidan Akmal al-Hanif

yang selalu ikhlas mendoakan dan mengarahkan dalam kebaikan yang dengan sabar membimbing. Semoga Allah SWT melindungi dan menjaga mereka dalam naunganNya..

Aamiin

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji bagi Allah SWT yang Maha Pengasih lagi Maha Penyayang atas Rahmat dan Hidayah-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi ini. Sholawat serta Salam tetap tercurahkan kepada junjungan kita, kekasih Allah, Nabi Muhammad SAW, sang pemberi syafaat kelak di hari akhir, beserta seluruh keluarga, sahabat, dan para pengikutnya.

Penelitian skripsi yang berjudul **“Implementasi Algoritma Fuzzy Type-2 Untuk Menentukan Perilaku NPC dalam Game Virtual Reality Survival Shooter”** ini ditulis untuk memnuhi salah satu syarat guna memperoleh gelar Sarjana Strata Satu (S1) Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Karya penelitian skripsi ini tidak akan pernah ada tanpa bantuan baik moral maupun spiritual dari berbagai pihak yang telah terlibat. Untuk itu dengan segala kerendahan hati, penulis mengucapkan rasa terimakasih yang sebesar-besarnya kepada:

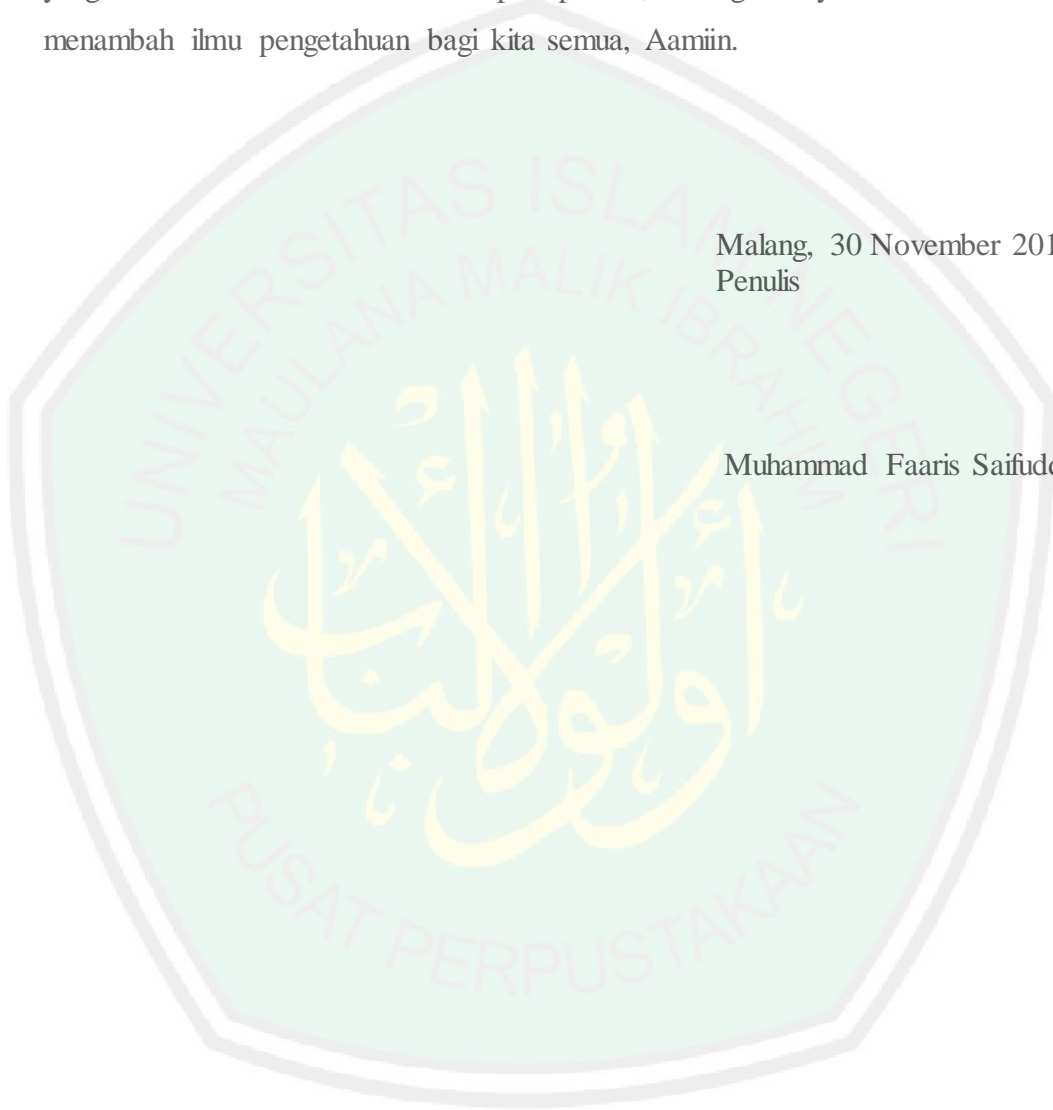
1. Prof. Dr. Abdul Haris, M.Ag., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Bapak Fachrul Kurniawan, M.MT., selaku Dosen Pembimbing I yang telah bersedia meluangkan waktu, tenaga dan pikiran untuk memberikan bimbingan, berbagai pengalaman, arahan, nasihat, motivasi dan pengarahan dalam pembangunan program hingga penyusunan skripsi ini.
3. Bapak Fresy Nugroho, S.T., M.T., selaku dosen pembimbing 2 yang selalu memberi masukan, serta pengarahan dalam penyusunan laporan skripsi ini.
4. Bapak Dr. Cahyo Crysdiyan, selaku dosen wali yang juga selalu memberi pengarahan terkait akademik selama masa studi.
5. Bapak Dr. Cahyo Crysdiyan selaku ketua jurusan Teknik Informatika yang mendukung dan mengarahkan skripsi ini.
6. Segenap civitas akademika Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang terutama seluruh dosen, terimakasih atas segala ilmu dan bimbingannya.
7. Ayah, Ibu, dan Adik serta seluruh keluarga besar yang selalu memberikan doa, kasih sayang, semangat, dukungan moril, serta motivasi sampai saat ini.

Harapan penulis semoga semua amal kebaikan dan jasa-jasa dari semua pihak yang telah membantu hingga skripsi ini selesai diterima oleh Allah SWT, serta mendapatkan balasan yang lebih baik dan berlipat ganda.

Penulis juga menyadari bahwa skripsi ini masih jauh dari kesempurnaan yang disebabkan keterbatasan harapan penulis, semoga karya ini bermanfaat dan menambah ilmu pengetahuan bagi kita semua, Aamiin.

Malang, 30 November 2017
Penulis

Muhammad Faaris Saifuddin



DAFTAR ISI

HALAMAN JUDUL	
HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
HALAMAN PERNYATAAN ORISINALITAS PENELITIAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiv
ABSTRAK	xv
ABSTRACT	xvi
المخلص	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
BAB II TINJAUAN PUSTAKA	5
2.1 Teori Penunjang	5
2.1.1 <i>Game</i>	5
a. Pengertian <i>Game</i>	5
b. Elemen Dasar <i>Game</i>	6
c. Jenis-Jenis <i>Game</i>	7
2.1.2 <i>Virtual Reality</i>	9
2.1.3 Himpunan <i>Fuzzy</i>	10
2.1.4 Algoritma <i>Fuzzy Type-2</i>	13
2.2 Penelitian Terkait	20

2.3 Metode Penelitian.....	21
BAB III DESAIN DAN RANCANGAN GAME	24
3.1 Deskripsi <i>Game</i>	24
3.2 <i>Storyline</i>	24
3.3 <i>Fuzzy State Machine</i> (FSM) NPC Teroris.....	26
3.4 Rancangan <i>Interface Game</i>	27
3.5 Deskripsi Karakter dan Objek	28
3.6 Perancangan Algoritma <i>Fuzzy Type-2</i>	30
3.6.1 Variabel <i>Fuzzy</i>	30
3.6.2 Nilai Linguistik	31
3.6.3 <i>Fuzzyfikasi</i>	31
3.6.4 Proses Inferensi	39
3.6.5 Reduksi Tipe	41
3.6.6 <i>Defuzzifikasi</i>	42
3.6.7 Contoh Perhitungan.....	42
3.6.7.1 <i>Fuzzyfikasi</i>	42
3.6.7.2 Inferensi dan Reduksi Tipe	45
3.6.7.3 <i>Defuzzifikasi</i>	50
BAB IV IMPLEMENTASI DAN PEMBAHASAN	51
4.1 Implementasi	51
4.1.1 Penggunaan Perangkat Keras	51
4.1.2 Penggunaan Perangkat Lunak	52
4.1.3 Implementasi Algoritma <i>Fuzzy Type-2</i>	52
4.1.4 Implementasi <i>Game</i>	70
4.2 Pengujian Algoritma <i>Fuzzy Type-2</i>	71
4.3 Integrasi dalam Islam	78
BAB V KESIMPULAN DAN SARAN	80
5.1 Kesimpulan	80
5.2 Saran	81
DAFTAR PUSTAKA	82

DAFTAR GAMBAR

Gambar 2.1 Representasi linier naik	11
Gambar 2.2 Representasi linier turun.....	12
Gambar 2.3 Kurva segitiga.....	12
Gambar 2.4 Kurva trapesium	12
Gambar 2.5 Sistem logika <i>fuzzy</i> bertipe 2	13
Gambar 2.6 Fungsi keanggotaan <i>interval</i> tipe 2	15
Gambar 2.7 Operasi <i>meet</i> pada himpunan <i>fuzzy</i> bertipe 2 <i>interval</i>	16
Gambar 2.8 Operasi <i>join</i> pada himpunan <i>fuzzy</i> bertipe 2 <i>interval</i>	16
Gambar 2.9 Operasi <i>meet</i> pada SLF bertipe 2 <i>interval</i>	17
Gambar 3.1 Peta kota	25
Gambar 3.2 <i>Spawn point player</i> dan NPC	26
Gambar 3.3 <i>Fuzzy State Machine</i> NPC Teroris	26
Gambar 3.4 Karakter teroris 1	28
Gambar 3.5 Karakter teroris 2	29
Gambar 3.6 Markas musuh 1	29
Gambar 3.7 Markas musuh 2	29
Gambar 3.8 Markas musuh 3	30
Gambar 3.9 Perancangan algoritma <i>fuzzy type-2</i>	30
Gambar 3.10 <i>Fuzzyfikasi</i>	31
Gambar 3.11 Grafik <i>input</i> variabel kesehatan.....	32
Gambar 3.12 Grafik <i>input</i> variabel jarak	34
Gambar 3.13 Grafik <i>input</i> variabel amunisi.....	37
Gambar 4.1 Tampilan awal <i>game</i>	70
Gambar 4.2 Tampilan menu <i>game</i>	70
Gambar 4.3 NPC patroli.....	70
Gambar 4.4 NPC menyerang	71
Gambar 4.5 NPC melarikan diri.....	71
Gambar 4.6 Tampilan simulasi <i>output</i> pada Matlab	72
Gambar 4.7 Sumbu kartesian <i>input</i> jarak dan kesehatan	73

Gambar 4.8 Sumbu kartesian *input* jarak dan amunisi..... 73
Gambar 4.9 Sumbu kartesian *input* kesehatan dan amunisi..... 74



DAFTAR TABEL

Tabel3.1 <i>Interface game</i>	27
Tabel4.1 Penggunaan perangkat keras	51
Tabel4.2 Penggunaan perangkat lunak.....	52
Tabel4.3 Keterangan <i>class fuzzy type-2</i>	52
Tabel4.4 Pengujian <i>fuzzy type-2</i>	75



ABSTRAK

Saifuddin, Muhammad Faaris. 2017. **Implementasi Algoritma Fuzzy Type-2 Untuk Menentukan Perilaku NPC Dalam Game Virtual Reality Survival Shooter**, Skripsi, Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang,

Pembimbing : (I) Fachrul Kurniawan, M.MT. (II) Fresy Nugroho, S.T., M.T.

Kata kunci : Algoritma Fuzzy Type-2, NPC

Sistem logika *fuzzy* bertipe 2 merupakan perluasan dari sistem logika *fuzzy* bertipe 1 dimana fungsi keanggotaan sistem logika *fuzzy* bertipe 2 memiliki dua derajat keanggotaan, yaitu derajat keanggotaan primer dan sekunder. Konsep *fuzzy* bertipe 2 ini diperkenalkan oleh Zadeh pada tahun 1970. Konsep utama dari logika *fuzzy* bertipe 2 ini adalah “kata dapat diartikan berbeda oleh orang yang berbeda”. *Non-Player Characters* (NPC) atau disebut juga agen adalah suatu entitas dalam *game* yang tidak dikendalikan secara langsung oleh pemain. NPC dikendalikan secara otomatis oleh komputer. Untuk dapat memperoleh perilaku cerdas dari NPC digunakan kecerdasan buatan atau *Artificial Intelligence* (AI). Penggunaan AI pada NPC dilakukan dengan pemberian Algoritma khusus sesuai dengan perilaku cerdas yang diharapkan.

Penelitian ini membahas mengenai implementasi Algoritma *Fuzzy Type-2* untuk menentukan perilaku NPC pada *game* “*Virtual Reality Survival Shooter*” berbasis *mobile*. Hasil dari penelitian ini adalah, Algoritma *Fuzzy Type-2* berhasil mengatur perilaku yang diterapkan pada NPC secara dinamis dengan 3 variabel *fuzzy*. *Persentase* keputusan yang dihasilkan yaitu, melarikan diri sebesar 17,65% menyerang 54,41% dan patroli sebesar 27,94% dengan 68 *input* yang berbeda.

ABSTRACT

Saifuddin, Muhammad Faaris. 2017. **Implementation of Fuzzy Type-2 Algorithm For Determining NPC Behavior In Virtual Reality Survival Shooter Game**, Thesis, Informatics Engineering Department, Faculty of Science and Technology, Islamic State University Maulana Malik Ibrahim Malang,

Counselor : (I) Fachrul Kurniawan, M.MT. (II) Fresy Nugroho, S.T., M.T.

Keywords: Fuzzy Type-2 Algorithm, NP

The fuzzy type 2 logic system is an extension of a fuzzy logic system of type 1 where the membership function of the fuzzy type 2 logic system has two degrees of membership, ie the degree of primary and secondary membership. This type 2 fuzzy concept was introduced by Zadeh in the 1970. The main concept of fuzzy type 2 logic is "words can be interpreted differently by different people". Non-Player Characters (NPC) or also called agents are an in-game entity that is not directly controlled by the player. NPC is automatically controlled by computer. To be able to obtain intelligent behavior from NPC used Artificial Intelligence (AI). The use of AI in NPC is done by giving special Algorithm according to the expected intelligent behavior.

This study discusses the implementation of Fuzzy Type-2 Algorithm to determine the behavior of NPC in mobile based Virtual Reality Survival Shooter game. The result of this research is, Fuzzy Type-2 algorithm managed to manage behavior applied to NPC dynamically with 3 fuzzy variables. The percentage of decisions that resulted, is, escape by 17.65% attack 54.41% and patrol by 27.94% with 68 different inputs.

المخلص

محمد فارس سيف الدين، ٢٠١٧، تنفيذ غامض من النوع ٢ خوارزمية لتحديد سلوك مجلس الشعب في الواقع الافتراضي بقاء مطلق النار لعبة، الرسالة، قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا، مالك جامعة مولانا ابراهيم الدولة الإسلامية في مالانج، المستشار : (١) فاهرول كورنياوان الماجستير . (٢) فريسي نوغروهو الماجستير.

كلمات البحث : غامض نوع ٢ خوارزمية، مجلس الشعب

نظام المنطق الضبابي من نوع ٢ هو امتداد لنظام المنطق الضبابي من نوع ١ حيث وظيفة عضوية نظام المنطق الضبابي من النوع ٢ واثنين من درجات العضوية، أي درجة من عضوية الابتدائية والثانوية. هذا المفهوم الغامض من نوع ٢ قدمه زاده في ١٩٧٠. المفهوم الرئيسي من المنطق الضبابي من نوع ٢ هو "يمكن أن تفسر الكلمات بشكل مختلف من قبل أشخاص مختلفين". أحرف غير لاعب (الشخصيات) ويسمى أيضا وكلاء أو في اللعبة هي الكيان الذي لا تسيطر مباشرة من قبل اللاعب. يتم التحكم نيك تلقائيا بواسطة الكمبيوتر. لتكون قادرة على الحصول على سلوك ذكي من مجلس الشعب تستخدم الذكاء الاصطناعي أو الذكاء الاصطناعي (منظمة العفو الدولية). يتم استخدام منظمة العفو الدولية في مجلس الشعب عن طريق إعطاء خوارزمية خاصة وفقا للسلوك ذكي المتوقع.

تتناول هذه الدراسة تنفيذ من النوع ٢ خوارزمية غامض لتحديد سلوك الشخصيات في مجال الهواتف المحمولة القائمة على الواقع الافتراضي بقاء مطلق النار لعبة. ونتيجة لهذا البحث، غامض من النوع ٢ خوارزمية تمكنت من إدارة السلوك ينطبق على مجلس الشعب حيوي مع المتغيرات ٣ غامض. هاجمت النسبة المئوية للقرارات التي أسفرت عن هروب بنسبة ١٧,٦٥٪ / ٥٤,٤١٪ ودوريات بنسبة ٢٧,٩٤٪ مع ٦٨ مدخلات مختلفة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Istilah teroris oleh para ahli dikatakan merujuk kepada para pelaku yang tidak tergabung dalam angkatan bersenjata yang dikenal atau tidak menuruti peraturan angkatan bersenjata tersebut. Aksi terorisme juga mengandung makna serangan yang dilakukan secara membabi buta dengan tujuan tertentu tanpa pertimbangan yang jelas dengan seringkali mengorbankan penduduk sipil dan infrastruktur dengan melanggar pasal-pasal tertentu dalam suatu negara untuk tujuan menyebarkan teror.

Terorisme di dunia bukanlah merupakan hal baru, namun menjadi aktual terutama sejak terjadinya peristiwa *World Trade Center* (WTC) di New York, Amerika Serikat pada tanggal 11 September 2001, dikenal sebagai September Kelabu, yang memakan lebih dari 3000 korban.

Kejadian ini merupakan isu global yang mempengaruhi kebijakan politik seluruh Negara di dunia, sehingga menjadi titik tolak persepsi untuk memerangi Terorisme sebagai musuh Internasional. Pembunuhan massal tersebut telah mempersatukan dunia melawan Terorisme Internasional. Terlebih lagi dengan diikuti terjadinya Tragedi Bali, tanggal 12 Oktober 2002 yang merupakan tindakan teror yang menimbulkan korban sipil yang terbilang tidak sedikit. Perang terhadap Terorisme yang dipimpin oleh Amerika mendapat sambutan dari seluruh dunia. Dengan dikeluarkannya *Anti Terrorism, Crime and Security Act* pada Desember 2001, diikuti tindakan-tindakan dari Negara lain yang pada intinya adalah melakukan perang atas tindak Terorisme di dunia.

Indonesia sebagai Negara Demokratis juga mengancam segala tindakan Terorisme salah satunya dengan disahkannya Undang Undang Republik Indonesia nomor 15 tahun 2003 sebagai penetapan pengganti Undang Undang nomor 1 tahun 2001 tentang Pemberantasan Tindak Pidana Terorisme sebagai Undang Undang di Indonesia.

Pemerintah Republik Indonesia juga membentuk tim khusus dalam penanganan kasus ini yaitu di bentuknya Detasemen Khusus 88 atau yang disebut dengan Densus 88 yang dilatih untuk mengatasi segala bentuk terorisme termasuk teror bom.

Indonesia sebagai Negara dengan mayoritas masyarakat beragama Islam yang berpegang teguh pada Al-Quran dan Sunnah, maka segala perilaku harus sesuai dengan syariat Islam,. Sebagaimana pelarangan terhadap sikap Terorisme yang tertuang dalam Ayat berikut :

وَابْتَغِ فِيمَا آتَاكَ اللَّهُ الدَّارَ الْآخِرَةَ ۖ وَلَا تَنْسَ نَصِيبَكَ مِنَ الدُّنْيَا ۗ وَأَحْسِنَ
 كَمَا أَحْسَنَ اللَّهُ إِلَيْكَ ۖ وَلَا تَبْغِ الْفُسَادَ فِي الْأَرْضِ ۗ إِنَّ اللَّهَ لَا يُحِبُّ
 الْمُفْسِدِينَ

“Dan carilah pada apa yang telah dianugerahkan Allah kepadamu (kebahagiaan) negeri akhirat, dan janganlah kamu melupakan bahagianmu dari (kenikmatan) duniawi dan berbuat baiklah (kepada orang lain) sebagaimana Allah telah berbuat baik, kepadamu, dan janganlah kamu berbuat kerusakan di (muka) bumi. Sesungguhnya Allah tidak menyukai orang-orang yang berbuat kerusakan.” (QS. Al-Qasas : 77)

Oleh sebab itu penelitian kali ini akan mengembangkan sebuah *game First Person Shooter* (FPS) mengenai perlawanan terhadap Terorisme di suatu kota berbasis *mobile*. Dimana *game* ini bertujuan untuk meningkatkan kesadaran masyarakat akan bahaya Terorisme, dampak serta kerugian yang ditimbulkan.

Dalam penelitian ini, peneliti menggunakan Algoritma *Fuzzy Type-2* untuk menentukan perilaku teroris, dimana Algoritma *Fuzzy Type-2* sendiri merupakan sebuah Algoritma yang mempresentasikan pengambilan keputusan dalam suatu kondisi tertentu.

1.2 Identifikasi Masalah

Berdasarkan latar belakang diatas, maka identifikasi masalah dari penelitian ini adalah :

1. Bagaimana membuat *game First Person Shooter* (FPS) berbasis *mobile*.
2. Bagaimana membuat *game Virtual Reality* berbasis *mobile*.
3. Bagaimana mengimplementasikan Algoritma *Fuzzy Type-2* untuk menentukan perilaku musuh.

1.3 Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut :

1. Penelitian ini difokuskan pada cara kerja Algoritma *Fuzzy Type-2* yang diterapkan pada NPC musuh.
2. *Game* ini berbasis *mobile*.
3. *Game* ini menggunakan sensor *gyroscope* dan *bluetooth controller* sebagai arah gerak karakter.

1.4 Tujuan Penelitian

Adapun tujuan penelitian ini adalah sebagai berikut :

1. Membuat sebuah *game First Person Shooter (FPS)* berbasis *mobile*.
2. Membuat *game Virtual Reality (VR)* berbasis *mobile*.
3. Mengimplementasikan Algoritma *Fuzzy Type-2* untuk mengatur perilaku NPC dalam menghadapi *player*.

1.5 Manfaat Penelitian

Manfaat dari pembuatan aplikasi ini bagi peneliti adalah untuk memberikan pengetahuan tentang bahaya Terorisme serta anjuran untuk memeranginya dalam bentuk *game First Person Shooter (FPS)* yang lebih menarik dan menyenangkan untuk dimainkan. Dan dapat mempelajari secara lebih rinci dari penggunaan Algoritma *Fuzzy Type-2* dalam pengaplikasiannya.

BAB II

TINJAUAN PUSTAKA

2.1 Teori Penunjang

2.1.1 *Game*

a. Pengertian *Game*

Teori permainan (*game*) pertama kali ditemukan oleh sekelompok ahli Matematika pada tahun 1944. Teori itu dikemukakan oleh John von Neumann dan Oskar Morgenstern yang berisi “ Permainan terdiri atas sekumpulan peraturan yang membangun situasi bersaing dari dua sampai beberapa orang atau kelompok dengan memilih strategi yang dibangun untuk memaksimalkan kemenangan lawan. Peraturan-peraturan menentukan kemungkinan tindakan untuk setiap pemain, sejumlah keterangan diterima setiap pemain sebagai kemajuan bermain, dan sejumlah kemenangan atau kekalahan dalam berbagai situasi.” (J. Von Neumann and O. Morgenstern, 1953).

Menurut Agustinus Nilwan dalam bukunya Pemrograman Animasi dan *Game* Profesional terbitan Elex Media Komputindo, *Game* merupakan permainan komputer yang dibuat dengan teknik dan metode animasi. Jika ingin mendalami penggunaan animasi haruslah memahami pembuatan *game*. Atau jika ingin membuat *game*, maka haruslah memahami teknik dan metode animasi, sebab keduanya saling berkaitan.

Menurut Clark C. Abt, *game* adalah kegiatan yang melibatkan keputusan pemain, berupaya mencapai tujuan dengan dibatasi oleh konteks tertentu (misalnya, dibatasi oleh peraturan).

Menurut Bernard Suits, *game* adalah upaya sukarela untuk mengatasi rintangan yang tidak perlu.

Menurut Greg Costikyan, *game* adalah sebarang karya seni di mana peserta, yang disebut pemain, membuat keputusan untuk mengelola sumberdaya yang dimilikinya melalui benda di dalam *game* demi mencapai tujuan .

b. Elemen Dasar *Game*

Menurut Teresa Dillon (futurelab.com, 2005) elemen-elemen dasar sebuah *game* adalah :

i. Game Rule

Game rule merupakan aturan perintah cara menjalankan, fungsi objek dan karakter di dunia permainan dunia *game*.

ii. Plot

Plot biasanya berisi informasi tentang hal-hal yang akan dilakukan oleh *player* dalam *game* dan secara detail.

iii. Theme

Di dalam biasanya ada pesan moral yang akan disampaikan.

iv. Character

Pemain sebagai karakter utama maupun karakter yang lain yang memiliki ciri dan sifat tertentu.

v. Object

Merupakan sebuah hal yang penting dan biasanya digunakan pemain untuk memecahkan masalah, adakalanya pemain harus punya keahlian dan pengetahuan untuk bisa memainkannya.

vi. *Text, graphic and sound*

Game biasanya merupakan kombinasi dari media teks, grafik maupun suara, walaupun tidak harus semuanya ada dalam *game*.

vii. *Animation*

Animasi ini selalu melekat pada dunia *game*, khususnya untuk gerakan karakter-karakter yang ada dalam *game*, properti dari objek.

viii. *User Interface*

Merupakan fitur-fitur yang mengkomunikasikan *user* dengan *game*. Mengenai komponen *game*, Sadiman (2008: 76) mengungkapkan empat komponen utama permainan yaitu :

- a. Adanya pemain,
- b. Adanya lingkungan di mana para *player* berinteraksi,
- c. Adanya aturan main,
- d. Adanya tujuan tertentu yang ingin dicapai.

c. *Jenis-Jenis Game*

i. *Shooting* (Tembak-tembak)

Video game jenis ini memerlukan kecepatan refleks, koordinasi mata-tangan, juga *timing*. Inti dari *game* jenis ini adalah tembak, tembak dan tembak. Contoh : Call OF Duty, Crysis.

ii. *Fighting* (Pertarungan)

Game yang permainannya memerlukan refleks dan koordinasi mata dan tangan dengan cepat, tetapi inti dari *game* ini adalah penguasaan hafalan jurus. Contoh : Mortal Kombat, Tekken.

iii. *Adventure* (Petualangan)

Game yang lebih menekankan pada jalan cerita dan kemampuan berfikir pemain dalam menganalisa tempat secara *visual*, memecahkan teka-teki maupun menyimpulkan berbagai peristiwa. Contoh : Kings Quest.

iv. Simulasi, Konstruksi, Manajemen

Video game jenis ini seringkali menggambarkan dunia di dalamnya sedekat mungkin dengan dunia nyata dan memperhatikan dengan detail berbagai faktor. Contoh : The Sims.

v. Strategi

Game jenis ini memerlukan koordinasi dan strategi dalam memainkan permainan ini. Kebanyakan *game* strategi adalah *game* perang. Contoh : Age Of Empire.

vi. Olahraga

Game ini merupakan adaptasi dari olahraga nyata, membutuhkan kelincahan dan juga strategi dalam memainkannya. Contoh : Pro Evolution Soccer, NBA.

vii. Puzzle

Game teka-teki, pemain diharuskan memecahkan teka-teki dalam *game* tersebut. Contoh : Tetris, Minesweeper.

viii. *Edugames* (Edukasi)

Video game jenis ini dibuat dengan tujuan spesifik sebagai alat pendidikan, entah untuk belajar mengenal warna untuk balita, mengenal huruf dan angka, Matematika, sampai belajar bahasa asing. Developer yang membuatnya,

harus memperhitungkan berbagai hal agar *game* ini benar-benar dapat mendidik, menambah pengetahuan dan meningkatkan keterampilan yang memainkannya. Target segmentasi pemain harus pula disesuaikan dengan tingkat kesulitan dan desain *visual* ataupun animasinya. Contoh : Bobi Bola, Dora the Explorer.

2.1.2 *Virtual Reality*

Virtual reality atau yang disebut dengan realitas maya adalah teknologi yang memungkinkan pengguna dapat berinteraksi dengan objek nyata yang disimulasikan menggunakan komputer. *Virtual memory* adalah lingkungan atau objek yang hanya ada dalam imajinasi yang mampu membangkitkan suasana 3 dimensi sehingga membuat pengguna seolah-olah terlibat secara fisik.

Piranti *Virtual reality* :

- *Glove*
Piranti masukan yang dapat menangkap gerakan tangan dan mengirim informasi ke sistem.
- *Headset*
Piranti yang berfungsi untuk memonitor gerakan kepala.
- *Walker*
Piranti yang digunakan untuk memantau gerakan kaki.

Cara Kerja *Virtual Reality* :

Cara kerja sistem *Virtual Reality* adalah pengguna memperhatikan suatu dunia semu, yang sebetulnya berbentuk gambar-gambar yang bersifat dinamis. Dengan media perangkat *headphone* atau *speaker*, pengguna bisa mendengar suara yang realistis. Dengan media *headset*, *glove* dan *walker*, semua gerakan pengguna dipantau oleh sistem

kemudian sistem memberikan reaksi yang sesuai. Sehingga seolah-olah pengguna merasakan sedang berada pada situasi yang nyata, dan dapat dirasakan baik secara fisik maupun psikologis.

2.1.3 Himpunan *Fuzzy*

Untuk memahami logika *fuzzy*, sebelumnya perhatikan dulu tentang konsep-konsep himpunan *fuzzy*. Himpunan *fuzzy* memiliki 2 atribut yaitu (T.Sutojo dkk, 2011):

1. *Linguistik*, yaitu nama suatu kelompok yang mewakili suatu keadaan tertentu dengan menggunakan bahasa alami, misalnya dingin, sejuk mewakili variabel temperatur.
2. *Numeris*, yaitu suatu nilai yang menunjukkan ukuran dari suatu variabel, misalnya 10, 35, 40 dan sebagainya.

Disamping itu, ada beberapa hal yang harus dipahami dalam memahami logika *fuzzy*, yaitu (T.Sutojo dkk, 2011):

1. Variabel *Fuzzy*, yaitu variabel yang akan dibahas dalam suatu sistem *fuzzy*. Contoh : penghasilan, temperatur, umur, dan sebagainya.
2. Himpunan *Fuzzy*, yaitu suatu kelompok yang mewakili suatu keadaan tertentu dalam suatu variabel *fuzzy*. Contoh: variabel umur terbagi menjadi 2 himpunan *Fuzzy* yaitu tua dan muda.
3. Semesta pembicaraan, yaitu seluruh nilai yang diizinkan untuk dioperasikan dalam suatu variabel *fuzzy*.

Contoh : Semesta pembicaraan untuk variabel temperatur [-10 90]

4. Domain Himpunan *Fuzzy*, yaitu seluruh nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*.

Fungsi Keanggotaan

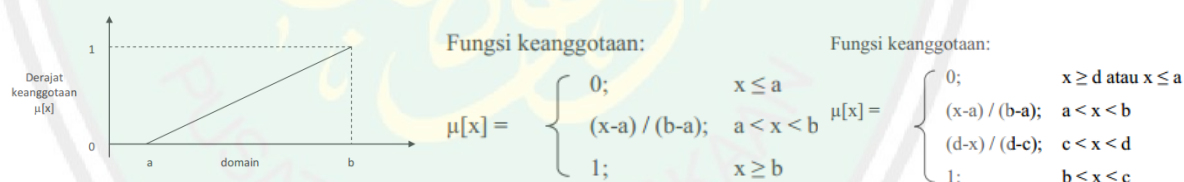
Fungsi keanggotaan adalah grafik yang mewakili besar dari derajat keanggotaan masing-masing variabel *input* yang berada dalam *interval* antara 0 dan 1. Derajat keanggotaan sebuah variabel x dilambangkan dengan symbol $\mu(x)$. Rule-rule nilai menggunakan nilai keanggotaan sebagai faktor bobot untuk menentukan pengaruhnya pada saat melakukan inferensi untuk menarik kesimpulan. (T.Sutojo dkk, 2011).

Ada beberapa fungsi yang bisa digunakan:

a. Representasi Linear

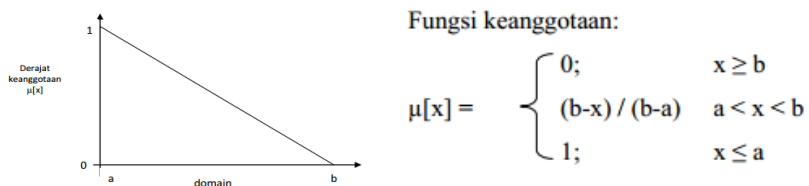
Pada representasi linear, pemetaan *input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan menjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas.

Ada 2 grafik keanggotaan linier. Pertama, grafik keanggotaan kurva linier naik, yaitu kenaikan himpunan *fuzzy* dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi. (T.Sutojo dkk, 2011).



Gambar 2.1 Representasi linier naik

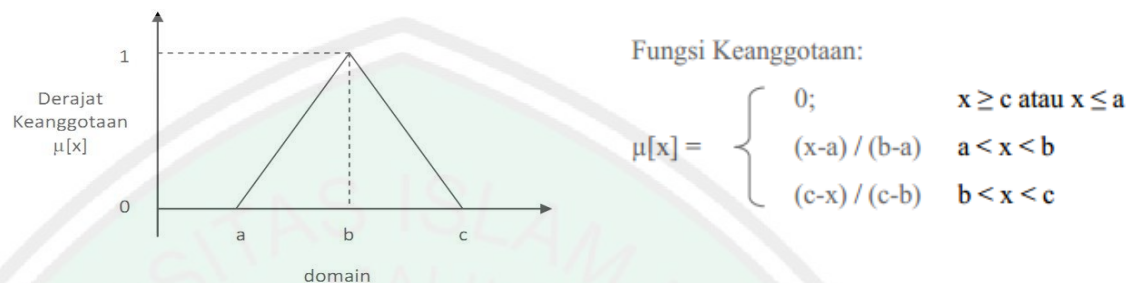
Kedua, merupakan kebalikan dari yang pertama. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah.



Gambar 2.2 Representasi linier turun

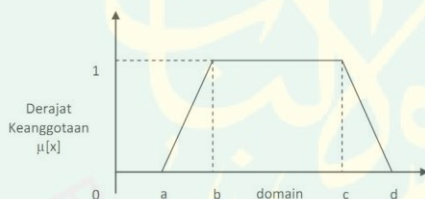
b. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis linear.

**Gambar 2.3** Kurva segitiga

c. Representasi Kurva Trapesium

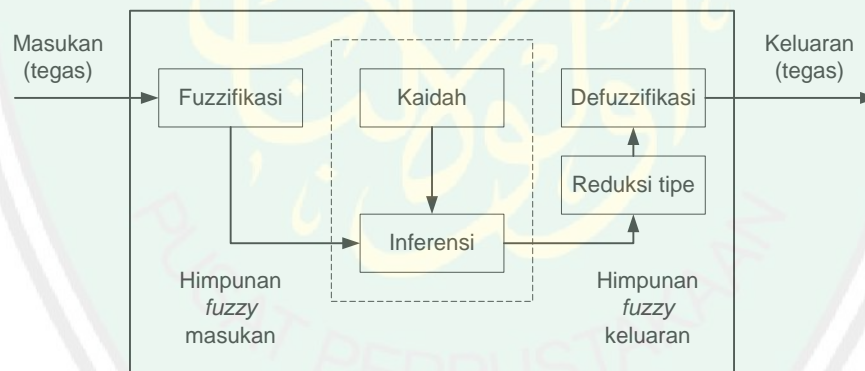
Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada titik yang memiliki nilai keanggotaan.

**Gambar 2.4** Kurva trapesium**2.1.4** Algoritma *Fuzzy Type-2*

Fuzzy type-2 merupakan bentuk pengembangan dari *fuzzy type-1* dikarenakan *fuzzy type-1* menggunakan *input* variabel dengan nilai yang pasti dimana hal itu bertentangan dengan kaidah *fuzzy* itu sendiri, maka muncullah konsep pengembangan *fuzzy* yaitu *fuzzy type-2* yang mulai diperkenalkan pada tahun 1960 oleh Zadeh.

Sama seperti pada sistem logika *fuzzy* bertipe 1, sistem logika *fuzzy* bertipe 2 juga terdiri dari *fuzzifikasi*, sekumpulan kaidah, mesin inferensi *fuzzy* dan pengolah keluaran.

Pengolah keluaran ini terdiri atas *type-reducer* (yang mengubah himpunan *fuzzy* bertipe 2 menjadi beberapa himpunan *fuzzy* bertipe 1) dan *defuzzifikasi* yang akan menghasilkan nilai tegas. Sistem logika *fuzzy* bertipe 2 juga disifati oleh aturan *IF-THEN*, tetapi himpunan keanggotaan *antecedent* dan atau *consequence*-nya adalah bertipe 2. Menurut Jerry M. Mendel dalam jurnalnya yang berjudul “*Type-2 Fuzzy Sets : Some Questions and Answers*” *IEEE Aug. 2003*, *type-2 fuzzy logic* memiliki performa yang lebih baik dari *type-1 fuzzy logic system* dalam mengatasi ketidakpastian karena perhitungannya yang lebih kompleks, meskipun membutuhkan komputasi yang lebih berat penggunaan interval *type-2 fuzzy* lebih mudah dipahami, disamping itu komputasi dengan *type-2 fuzzy logic* sangat mudah diatur sehingga menjadikannya mudah dalam pengaplikasiannya. Secara umum sistem logika *fuzzy* bertipe 2 digambarkan sebagai berikut :



Gambar 2.5 Sistem logika *fuzzy* bertipe 2

Fungsi Keanggotaan *Upper* dan *Lower*

Footprint of uncertainty (FOU) adalah daerah terbatas yang memuat ketidakpastian derajat keanggotaan primer dari fungsi keanggotaan tipe 2. *Upper* dan *lower membership function* adalah dua buah fungsi keanggotaan tipe 1 yang membatasi *footprint of uncertainty* (FOU) fungsi keanggotaan *interval* tipe 2. *Upper* adalah himpunan bagian yang memiliki derajat keanggotaan tertinggi dalam FOU. Sedangkan *lower* adalah derajat keanggotaan yang merupakan bagian dari derajat keanggotaan primer.

Dengan mengambil

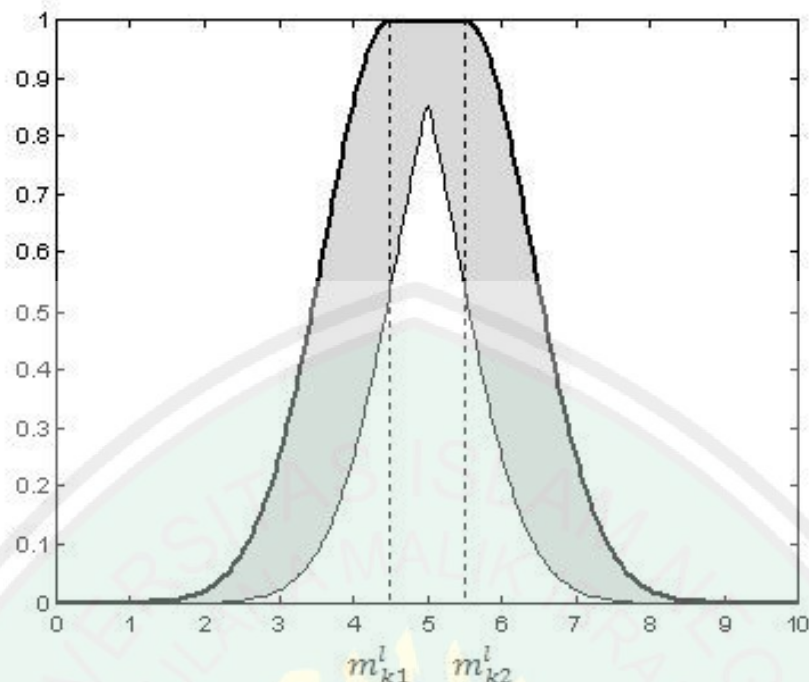
$$N(m_{k1}^l, \sigma_k^l; x_k) \cong \exp\left(-\frac{1}{2}\left(\frac{x_k - m_{k1}^l}{\sigma_k^l}\right)^2\right)$$

maka fungsi keanggotaan *upper* didefinisikan sebagai berikut :

$$\bar{\mu}_k^l(x_k) = \begin{cases} N(m_{k1}^l, \sigma_k^l; x_k), & x_k < m_{k1}^l \\ 1, & m_{k1}^l \leq x_k \leq m_{k2}^l \\ N(m_{k2}^l, \sigma_k^l; x_k), & x_k > m_{k2}^l \end{cases}$$

Sedangkan fungsi keanggotaan *lower* didefinisikan sebagai berikut :

$$\underline{\mu}_k^l(x_k) = \begin{cases} N(m_{k2}^l, \sigma_k^l; x_k), & x_k \leq \frac{m_{k1}^l + m_{k2}^l}{2} \\ N(m_{k1}^l, \sigma_k^l; x_k), & x_k > \frac{m_{k1}^l + m_{k2}^l}{2} \end{cases}$$



Gambar 2.6 Fungsi keanggotaan *interval* tipe 2

Berdasarkan diagram diatas, garis tebal adalah *upper* MF, sedangkan garis tipis adalah *lower* MF

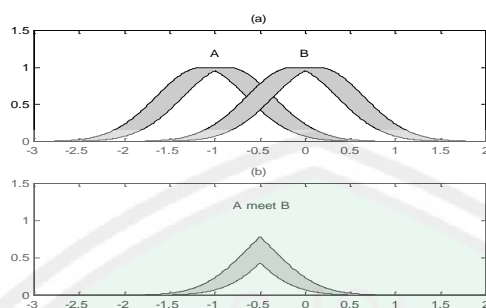
Operasi *Meet* untuk Himpunan *Interval*

Untuk operasi irisan pada logika *fuzzy* bertipe 2 disebut sebagai operasi *meet*. Misalkan $F = \int_{v \in F} 1/v$ dan $G = \int_{w \in G} 1/w$ adalah dua himpunan *interval* tipe 1 dengan domain $v \in [l_f, r_f] ([l_f, r_f] \subseteq [0,1])$ dan $w \in [l_r, r_r] ([l_r, r_r] \subseteq [0,1])$, maka operasi *meet* antara F dan G adalah: $Q = FG$ ($Q = \int_{v \in Q} 1/q$), dalam *T-norm* minimum atau produk yang didefinisikan sebagai berikut:

$$Q = FG = \int_{q \in [l_f * l_g, r_f * r_g]} 1/q$$

dimana $q = v * w$.

Operasi *meet* dari dua himpunan *fuzzy* tipe 2 *interval*, secara grafis dapat dilihat pada gambar berikut :

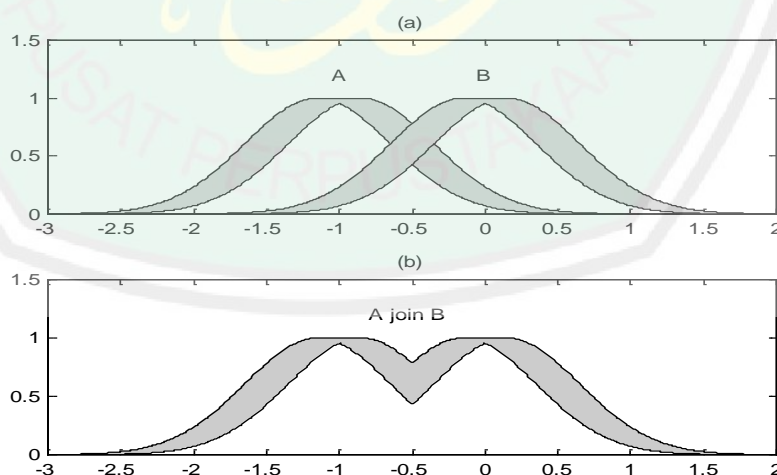


Gambar 2.7 Operasi *meet* pada himpunan *fuzzy* bertipe 2 *interval*

Operasi *Join* untuk Himpunan *Interval*

Untuk operasi union pada logika *fuzzy* bertipe 2 disebut sebagai operasi *join*. Misalkan himpunan F dan G seperti yang telah didefinisikan pada bagian sebelumnya, maka operasi *join* antara F dan G adalah $Q = FG$ ($Q = \int_{v \in Q} 1/q$), yang didefinisikan sebagai berikut:

$$Q = FG = \int_{q \in [l_f \vee l_g, r_f \vee r_g]} 1/q$$



dimana $q = v \vee w$.

Gambar 2.8 Operasi *join* pada himpunan *fuzzy* bertipe 2 *interval*

Inferensi Sistem Logika Fuzzy Bertipe 2 Interval

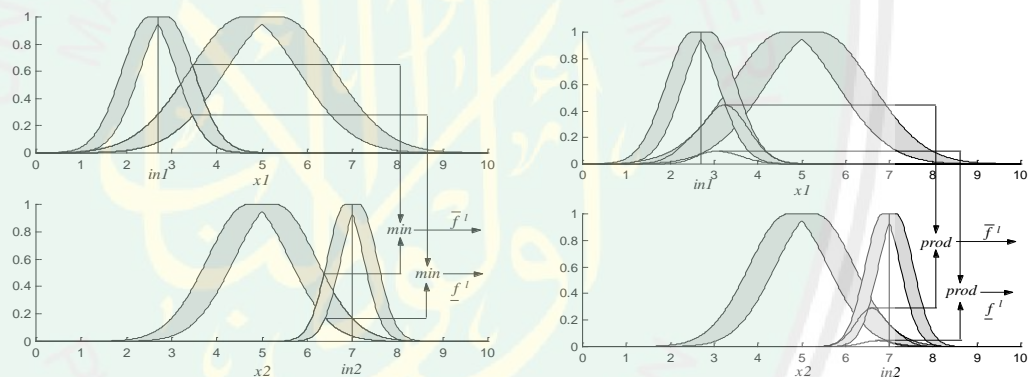
Ketika besaran masukan *defuzzifikasi* menggunakan himpunan fuzzy bertipe 2 interval maka F^l adalah himpunan interval tipe 1 yakni $F^l = [f^l, \bar{f}^l]$, dimana

$$\underline{f}^l = \sup \int_{X_1} \dots \int_{X_P} [\underline{\mu}_{\tilde{X}_1(x_1)} * \underline{\mu}_{\tilde{F}_k^l(x_1)}] * \dots * [\underline{\mu}_{\tilde{X}_P(x_P)} * \underline{\mu}_{\tilde{F}_k^l(x_P)}] / X$$

dan

$$\bar{f}^l = \sup \int_{X_1} \dots \int_{X_P} [\bar{\mu}_{\tilde{X}_1(x_1)} * \bar{\mu}_{\tilde{F}_k^l(x_1)}] * \dots * [\bar{\mu}_{\tilde{X}_P(x_P)} * \bar{\mu}_{\tilde{F}_k^l(x_P)}] / X$$

supremum diperoleh ketika setiap suku dalam kotak mencapai *supremum*, hal ini dapat dilihat pada berikut



Gambar 2.9 Operasi *meet* pada SLF bertipe 2 interval

Reduksi Tipe dan Defuzzifikasi

Setelah proses *fuzzifikasi*, inferensi *fuzzy*, reduksi tipe dan *defuzzifikasi*, maka keluaran dari SLF tipe 2 adalah besaran nyata. Ada beberapa cara reduksi tipe, diantaranya adalah *centroid*, *height* dan *center of set*. Secara umum keluaran dari SLF tipe 2 interval adalah

$$Y(Z_1, \dots, Z_M, W_1, \dots, W_M) = \int_{Z_1} \dots \int_{Z_M} \int_{W_1} \dots \int_{W_M} 1 / \frac{\sum_{i=1}^M w_i z_i}{\sum_{i=1}^M w_i}$$

Setiap Z_l ($l=1, \dots, M$) adalah himpunan tipe 1 *interval*, yang memiliki pusat c_l dan lebar (*spread*) s_l ($s_l \geq 0$). Setiap W_l juga merupakan himpunan tipe 1 *interval* yang memiliki pusat h_l dan lebar (*spread*) Δ_l ($\Delta_l \geq 0$), dengan anggapan $h_l \geq \Delta_l$ sehingga $w_l \geq 0$ untuk $l=1, \dots, M$. demikian juga Y adalah himpunan tipe 1 *interval*, sehingga yang dihitung adalah titik paling kiri (y_l) dan paling kanan (y_r). Nilai y_l memiliki ketergantungan terhadap nilai $c_l - s_l$ dan salah satu titik batas dari W_i , sedangkan nilai y_r tergantung pada nilai $c_l + s_l$ dan salah satu titik batas dari W_i . Persamaan berikut adalah persamaan yang digunakan selama iterasi dalam menghitung nilai y_l dan y_r .

$$S(w_1, \dots, w_M) \cong \frac{\sum_{l=1}^M w_l z_l}{\sum_{l=1}^M w_l}$$

Dengan $w_l \in [h_l - \Delta_l, h_l + \Delta_l]$ dan $h_l \geq \Delta_l$ untuk $l=1, \dots, M$ sedangkan $z_l \in [c_l - s_l, c_l + s_l]$. S memiliki nilai maksimum pada y_r . Untuk menghitung y_r , pertama kita memakai $z_l = c_l + s_l$ untuk ($l=1, \dots, M$) dengan tanpa menghilangkan asas umum, z_l dianggap telah tersusun dari nilai terkecil hingga terbesar. yaitu $z_1 \leq z_2 \leq \dots \leq z_M$. Langkah selanjutnya adalah :

1. menghitung $S' = S(h_1, \dots, h_M)$ menggunakan persamaan (II.59), dengan $w_l = h_l$ untuk $l = 1, \dots, M$;
2. mencari k ($1 \leq k \leq M-1$) demikian sehingga $z_k \leq S' \leq z_{k+1}$;
3. menghitung $S'' = S(h_1 - \Delta_1, \dots, h_k - \Delta_k, h_{k+1} + \Delta_{k+1}, \dots, h_M + \Delta_M)$ menggunakan Persamaan (II.59), dengan $w_l = h_l - \Delta_l$ untuk $l \leq k$ dan $w_l = h_l + \Delta_l$ untuk $l \geq k+1$;
4. memeriksa apakah $S'' = S'$, jika ya maka iterasi berhenti; jika tidak, selanjutnya ke langkah 5;
5. mengganti nilai S' dengan S'' , selanjutnya ke langkah 2.

Nilai minimum $S(w_1, \dots, w_M)$ yaitu y_l , dapat dihitung menggunakan prosedur yang sama seperti prosedur di atas. Hanya ada dua perubahan yaitu menggunakan $z_l = c_l - s_l$ untuk $(l=1, \dots, M)$. Kemudian pada langkah 3, untuk menghitung $S'' = S(h_1 + \Delta_1, \dots, h_k + \Delta_k, h_{k+1} - \Delta_{k+1}, \dots, h_M - \Delta_M)$ menggunakan $w_l = h_l + \Delta_l$ untuk $l \leq k$ dan $w_l = h_l - \Delta_l$ untuk $l \geq k+1$. Prosedur ini dapat digunakan untuk semua pereduksi.

A. Reduksi Tipe Centroid (*centroid type reduction*)

Yang dihitung dalam pereduksi ini adalah hasil *join* dari *antecedent* kaidah-kaidah yang aktif. Dengan menggunakan prosedur di atas, maka M adalah banyaknya pembagian (*diskritisasi*) semesta pembicaraan Y sehingga $Y(y_1, \dots, y_M)$, $c_l = y_l$; $s_l = 0$. $\mu_{\tilde{B}}(y_l)$ memiliki domain $[L_l, R_l]$, sehingga $h_l = (L_l + R_l)/2$ dan $\Delta_l = (R_l - L_l)/2$.

$$Y_c(x) = \int_{\theta_1} \dots \int_{\theta_M} 1 / \frac{\sum_{i=1}^M y_i \theta_i}{\sum_{i=1}^M \theta_i}$$

B. Reduksi Tipe Ketinggian (*height type reduction*)

\bar{y}^l adalah titik dalam semesta pembicaraan Y yang memiliki nilai derajat keanggotaan paling tinggi dalam kaidah ke- l . Untuk memakai prosedur reduksi di atas, maka $c_l = \bar{y}^l$, $s_l = 0$. $\mu_{\tilde{B}}(\bar{y}^l)$ memiliki domain $[L_l, R_l]$, sehingga $h_l = (L_l + R_l)/2$ dan $\Delta_l = (R_l - L_l)/2$.

$$Y_h(x) = \int_{\theta_1} \dots \int_{\theta_M} 1 / \frac{\sum_{i=1}^M y^l \theta_i}{\sum_{i=1}^M \theta_i}$$

C. Reduksi Tipe Pusat himpunan (*center of sets type reduction*)

Untuk pereduksi jenis ini, yang pertama dilakukan adalah mencari nilai *centroid* (C_l) dari himpunan *interval consequence* setiap kaidah yang aktif. Domain dari C_l adalah

$[L_i^c, R_i^c]$, sehingga $c_i = (L_i^c + R_i^c)/2$, $s_i = (R_i^c - L_i^c)/2$ sedangkan domain derajat keanggotaan *consequent*-nya adalah $[L_i, R_i]$, sehingga $h_i = (L_i + R_i)/2$ dan $\Delta_i = (R_i - L_i)/2$.

Nilai tegas yang diperoleh dari ketiga jenis pereduksi adalah :

$$y = \frac{y_l + y_r}{2}$$

2.2 Penelitian Terkait

Berikut adalah penelitian terkait dengan penelitian yang telah dilakukan, yaitu :

1 Simulasi Sistem Kendali *Heat Exchanger* Berbasis *Interval Type-2 Logic Controller*.

Penelitian ini dilakukan oleh Putri Nurul Jayanti, Mahasiswa Jurusan Teknik Elektro, Fakultas Teknologi Industri, Universitas Islam Indonesia, Yogyakarta pada tahun 2013. Penelitian tersebut bertujuan untuk membandingkan sistem kerja *Type-1 Fuzzy Logic Controller* (T1FLC) dan *Type-2 Fuzzy Logic Controller* (T2FLC) pada *Stirred Tank Heat Exchanger* sehingga akan diperoleh desain sistem kendali dan perbandingan efisiensi komputasi terhadap kedua Algoritma.

Berdasarkan penelitian tersebut, dapat diambil beberapa kesimpulan, yaitu sebagai berikut :

1. Sistem kendali *stirred tank heat exchanger* dengan T2FLC memiliki performa yang lebih baik dibandingkan T1FLC pada kondisi sistem dengan *noise*.
2. Sistem kendali *stirred tank exchanger* dengan T2FLC memiliki respon yang lebih baik atau lebih cepat dalam menangani adanya gangguan berupa perubahan parameter *plant*, antara lain perubahan kecepatan aliran panas (F), suhu cairan pendingin (T_{cin}), dan suhu cairan panas (T_0) dibandingkan dengan T1FLC.

2 Sistem Kendali *Fuzzy Bertipe-2 Interval* dengan Struktur Adaptif Beracuan Model.

Penelitian ini dilakukan oleh Bambang Riyanto dan Wakhyu Dwiono, Mahasiswa Jurusan Teknik Elektro, Institut Teknologi Bandung pada Tahun 2006. Penelitian tersebut bertujuan untuk merancang suatu sistem kendali yang mekanismenya dilakukan secara adaptif dengan acuan model terhadap *Fuzzy bertipe-2* dan *Fuzzy bertipe-1*. Berdasarkan hasil simulasi yang telah dilakukan, dapat diambil kesimpulan bahwa *Fuzzy adaptif bertipe-2* memiliki kinerja yang lebih baik dibandingkan pengendali *Fuzzy adaptif bertipe-1*. Meskipun demikian, peningkatan kinerja pada pengendali adaptif bertipe-2 memerlukan beban komputasi yang relatif lebih tinggi.

3 *Fuzzy Controller Type-2* berbasis Metode *Cycle-to-Cycle* untuk Rehabilitasi *Swing Gait Phase* dengan FES.

Penelitian ini dilakukan oleh Hendi Wicaksono Agung D dan Achmad Arifin, Mahasiswa Pasca Sarjana Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember, Surabaya pada Tahun 2009. Penelitian tersebut bertujuan untuk mengaplikasikan Algoritma *Fuzzy Type-2* untuk mengontrol *Functional Electrical Stimulation* (FES) sehingga menghasilkan *swing gait phase*, yaitu kemampuan untuk meningkatkan motorik jaringan syaraf otot pasien penderita stroke.

2.3 Metode Penelitian

Peneliti membagi pengerjaan penelitian ini menjadi beberapa tahap, antara lain :

1. Studi literatur

Pada tahap ini dilakukan berbagai pengumpulan data literatur-literatur terkait penelitian ini sebagai berikut:

- Literatur di dapatkan dari buku, jurnal, atau skripsi terdahulu
- Literatur berisi informasi tentang pembuatan *game virtual reality* dan juga tentang algoritma *fuzzy type-2* yang akan diterapkan pada *NPC* yang ada dalam *game*.

Untuk selanjutnya akan dilakukan analisis terhadap hasil pengumpulan data dari literatur yang telah didapatkan.

2. Perancangan dan desain aplikasi

Pada tahap ini, perancangan aplikasi terdiri atas perancangan proses-proses utama dan desain aplikasi yang terdiri atas desain *menu game* dan desain utama dari *game* itu sendiri.

3. Pembuatan aplikasi

Pada tahap ini, akan dilakukan pembangunan aplikasi *game* dengan menuliskan bahasa pemrograman pada *compiler* sehingga menghasilkan *game* yang sesuai dengan hasil perancangan.

4. Uji coba dan evaluasi

Pada tahap ini, akan diketahui hasil dari implementasi algoritma *fuzzy type-2* pada *game* apakah sudah optimal atau belum, terjadi kesalahan atau tidak pada *game* tersebut. Apabila tidak terjadi kesalahan, akan dilakukan pemeliharaan secara berkala dan aplikasi *game* tersebut disesuaikan dengan kondisi lingkungan tersebut diaplikasikan serta akan dilakukan pengembangan pada aplikasi *game* tersebut.

5. Penyusunan Laporan

Dalam pembuatan laporan ini nantinya diharapkan bisa bermanfaat bagi penelitian-penelitian lebih lanjut yang mana penelitian ini berisi hasil dari seluruh dokumentasi pelaksanaan penelitian. Berikut adalah permodelan dari pembagian pengerjaan penelitian.



Diagram 2.1 Permodelan pengerjaan penelitian

BAB III

DESAIN DAN PERANCANGAN *GAME*

3.1 Deskripsi *Game*

Game ini merupakan *game* bergenre *First Person Shooter (FPS) Survival Shooter* berbasis *mobile* yang menggunakan *joystick* dan sensor *gyroscope* sebagai arah gerak karakter. Sebagai *game survival shooter*, *player* dihadapkan dengan kondisi *survival* atau bertahan hidup. Skenario pada *game* ini adalah *player* diharuskan untuk bertahan selama mungkin melawan serangan teroris yang akan di *spawn* pada *interval* waktu tertentu, teroris akan bergerak berdasarkan penerapan dari *input* algoritma *fuzzy type-2* baik berupa mengejar, menyerang, berpatroli ataupun melarikan diri, sehingga diharapkan *game* akan berjalan senyata mungkin dengan kemungkinan berbeda pada setiap gerakan musuh. *Game* ini akan menjadi semakin susah seiring waktu berjalan, dimana *player* akan dihadapkan pada musuh yang semakin banyak, sehingga akan membuat situasi semakin seru dan menantang.

3.2 *Storyline*

Game ini berlatar pada suatu kota yang telah terjadi peperangan, peperangan tersebut melibatkan pihak teroris dan tentara setempat. Karena sengitnya perang secara mengejutkan pihak teroris berhasil menguasai kota.

Sebagai bentuk tindakan untuk merebut kembali kota, maka *player* ditugaskan untuk menyerang 3 bangunan besar yaitu markas logistik, gudang persenjataan serta markas utama teroris.

Berikut Skenario *game* yang akan dimainkan :

- Terdapat 3 markas musuh yaitu 3 bangunan besar di tiga sudut kota yang berupa gudang logistik, gudang persenjataan, serta markas utama teroris.
- Teroris akan berpusat disekitar markas sehingga keamanan markas lebih terjaga.
- Tingkat keamanan tiap markas berbeda beda berdasarkan tingkatannya, yaitu sebagai berikut :
 - Gudang logistik dengan tingkat penjagaan rendah.
 - Gudang persenjataan dengan tingkat penjagaan sedang.
 - Markas utama dengan tingkat penjagaan tinggi.
- Terdapat teroris yang bertugas berkeliling kota untuk berpatroli dan mengejar maupun menyerang musuh yang terlihat.
- *Player* akan di-*spawn* pada sudut lain kota, semakin dekat *player* dengan markas maka semakin banyak musuh yang akan dilawan.
- *Score* / nilai pada *game* ini ditentukan oleh seberapa banyak musuh yang berhasil dikalahkan.

Berikut adalah peta kota beserta *spawn point player* dan NPC :



Gambar 3.1 Peta kota

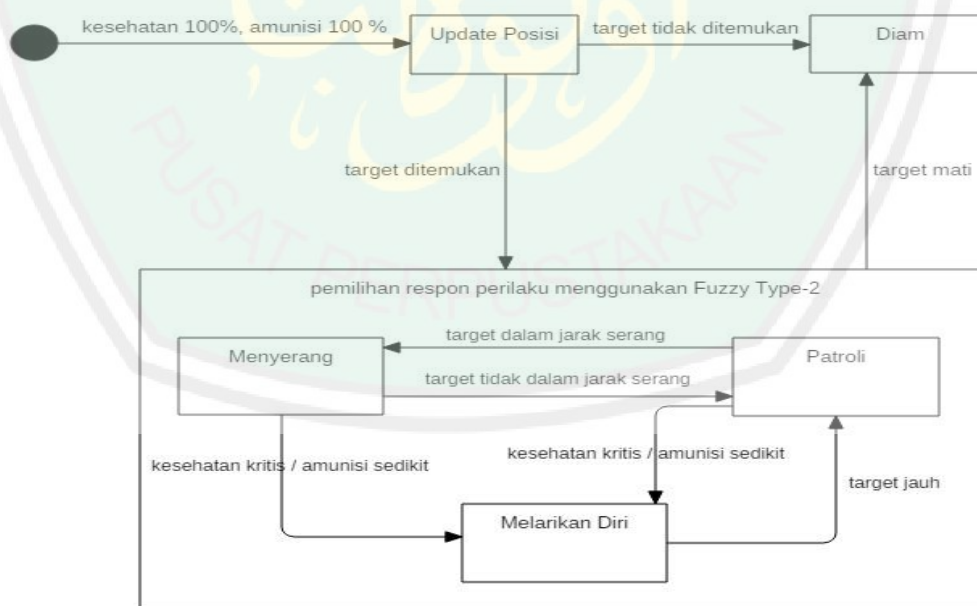


Gambar 3.2 *Spawn point player dan NPC*

Keterangan Gambar :

1. Markas musuh beserta *spawn point* teroris
2. *Spawn point* player

3.3 *Fuzzy State Machine (FSM) NPC Teroris*




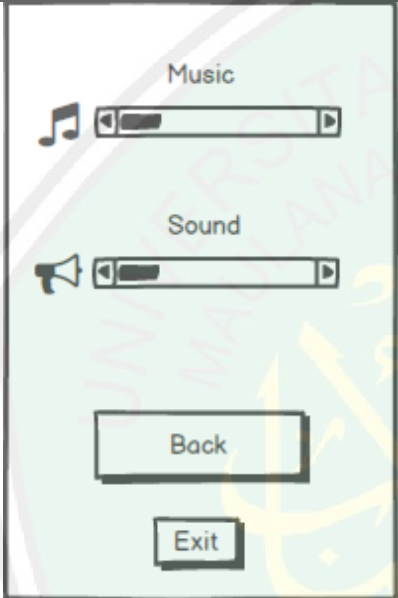
Gambar 3.3 *Fuzzy State Machine NPC teroris*

Keterangan : Ketika *game* dimulai, *NPC* teroris akan di *spawn* pada titik tertentu dalam kota pada interval tertentu (*update* posisi), kemudian *NPC* teroris akan mencari target (*player*). Ketika menemukan posisi target ataupun ketika ada serangan dari target maka *NPC* teroris akan melakukan pemilihan respon perilaku berdasarkan Algoritma *Fuzzy Type-2*, ketika target mati atau tidak ditemukan *NPC* teroris akan diam.

3.4 Rancangan Interface Game

Tabel 3.1 Interface game

1.		<p>Tampilan Menu <i>Game</i>. Terdapat 5 <i>button</i> yaitu : <i>New Game</i>, <i>Continue</i>, <i>Setting</i>, <i>Back</i>, <i>Quit</i></p>	<p><i>Button New Game</i> merupakan tombol untuk memulai permainan baru <i>Button Continue</i> merupakan tombol untuk melanjutkan permainan <i>Button Setting</i> merupakan tombol untuk membuka jendela pengaturan <i>Button Back</i> merupakan tombol untuk berpindah ke</p>
----	--	---	---

		menu sebelumnya <i>Button Quit</i> merupakan tombol untuk keluar dari permainan
2		<p>Jendela <i>Setting</i> untuk mengatur <i>volume</i> musik pada permainan</p> <p>Terdapat 2 <i>slide bar</i> yaitu <i>Music</i> dan <i>Sound</i> untuk mengatur <i>volume</i> suara pada permainan</p>

3.5 Deskripsi Karakter dan Objek

A. Karakter Musuh



Gambar 3.4 Karakter teroris 1

Bertugas patroli kota dan akan menyerang musuh yang terlihat, memiliki *hitpoint* serta daya serang rendah, tetapi memiliki kecepatan gerak tinggi.



Gambar 3.5 Karakter teroris 2

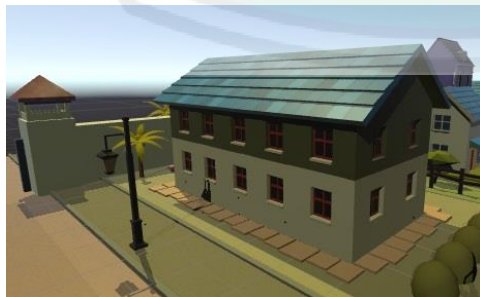
Bertugas menjaga markas, memiliki *hitpoint* dan daya serang tinggi, tetapi memiliki kecepatan gerak yang lambat.

B. Karakter Kota



Gambar 3.6 Markas musuh 1

Gudang logistik dengan tingkat penjagaan rendah.



Gambar 3.7 Markas musuh 2

Gudang persenjataan dengan tingkat penjagaan sedang.

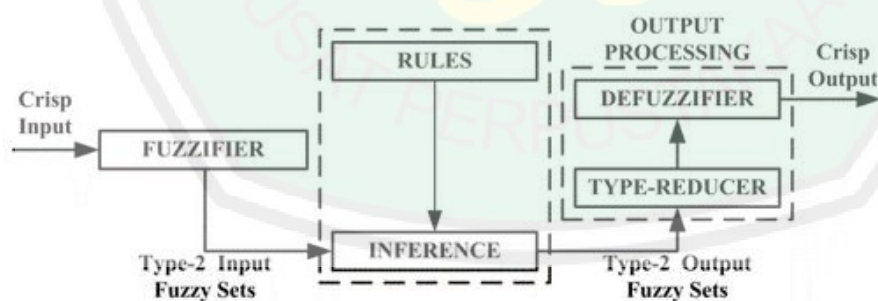


Gambar 3.8 Markas musuh 3

Markas utama dengan tingkat penjagaan tinggi.

3.6 Perancangan Algoritma *Fuzzy Type-2*

Pada *game* ini algoritma *fuzzy type-2* digunakan untuk memprediksi dan mengatur perilaku teroris, dimana teroris akan mengambil keputusan patroli, menyerang ataupun melarikan diri ketika dihadapkan pada suatu kondisi. *Fuzzy type-2* dikenal memiliki performa yang lebih baik dari *fuzzy type-1* dalam mengatasi ketidakpastian sehingga diharapkan hasil yang diperoleh akan menjadi lebih akurat, disamping itu komputasi dengan *interval type-2 fuzzy logic* lebih mudah diatur sehingga menjadikannya lebih mudah dalam implementasinya. Adapun tahapan-tahapan dalam perancangan algoritma *fuzzy type-2* adalah sebagai berikut :



Gambar 3.9 Perancangan algoritma *fuzzy type-2*

3.6.1 Variabel *Fuzzy*

Terdapat 3 variabel dalam fungsi *fuzzy* yang digunakan pada *game* ini, yaitu

variabel kesehatan, variabel amunisi dan variabel jarak sebagai *input*. Sedangkan variabel *output*nya adalah variabel perilaku.

3.6.2 Nilai Linguistik

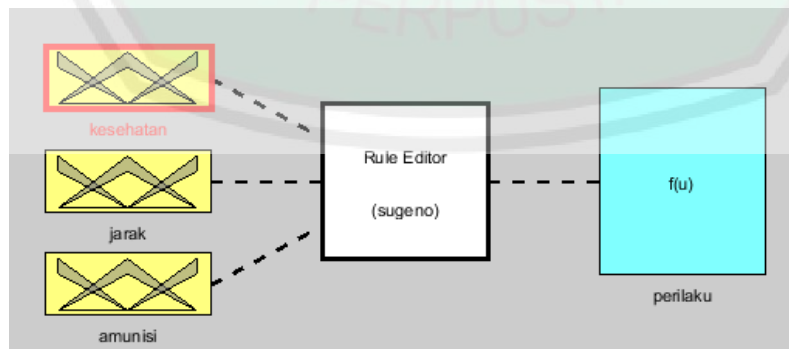
Dari 4 variabel yang digunakan, maka nilai linguistinya adalah sebagai berikut

:

1. Variabel Kesehatan NPC dibagi menjadi tiga himpunan *fuzzy* yaitu : buruk, sedang dan baik.
2. Variabel Jarak NPC dengan *player* dibagi menjadi tiga himpunan *fuzzy* yaitu : jauh, sedang dan dekat.
3. Variabel Amunisi NPC dibagi menjadi tiga himpunan *fuzzy* yaitu : banyak, sedang dan sedikit.
4. Variabel *output* Perilaku dibagi menjadi tiga himpunan *fuzzy* yaitu : patroli, menyerang dan melarikan Diri.

3.6.3 Fuzzyfikasi

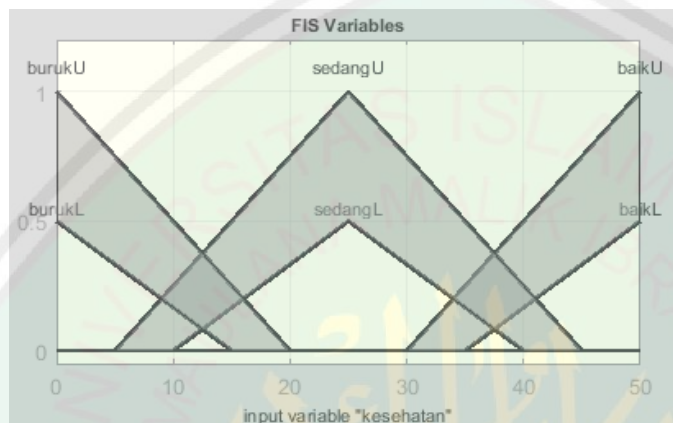
Fuzzyfikasi yaitu suatu proses untuk mengubah suatu masukan dari bentuk tegas (*crisp*) menjadi *fuzzy* (variabel linguistik) yang biasanya disajikan dalam bentuk himpunan-himpunan *fuzzy* dengan suatu fungsi kenggotaannya masing-masing.



Gambar 3.10 *Fuzzyfikasi*

Berdasarkan *fuzzy interface system* diatas maka pemetaan himpunan *fuzzy* adalah sebagai berikut :

1. Variabel kesehatan, terbagi menjadi tiga himpunan yaitu: baik, sedang dan buruk. *Range* nilai untuk variabel kesehatan dengan rentang nilai 0 - 50 akan dijelaskan sebagai berikut ini :



Gambar 3.11 Grafik *input* variabel kesehatan

Dari grafik *input* variabel kesehatan diatas, fungsi kurva bahu kiri untuk variabel linguistik “buruk”, fungsi kurva segitiga untuk variabel linguistik “sedang” dan fungsi kurva bahu kanan mewakili variabel linguistik “baik”, didapatkan nilai *range* pada setiap himpunan yaitu :

- a. Buruk (0 – 20) dengan nilai fungsi keanggotaan *Upper* (0 – 20) dan nilai fungsi keanggotaan *Lower* (0 – 15).
- b. Sedang (5 – 45) dengan nilai fungsi keanggotaan *Upper* (5 – 45) dan nilai fungsi keanggotaan *Lower* (10 – 40)
- c. Baik (30 – 50) dengan nilai fungsi keanggotaan *Upper* (30 – 50) dan nilai fungsi keanggotaan *Lower* (35 – 50)

Berikut perhitungan manual dari fungsi diatas :

Bahu Kiri : buruk

$$\mu_{burukUpper}[x] = \begin{cases} 0; & x \geq 20 \\ \frac{20 - x}{20 - 0}; & 0 < x < 20 \\ 1; & x \leq 0 \end{cases}$$

$$\mu_{burukLower}[x] = \begin{cases} 0; & x \geq 15 \\ \frac{15 - x}{15 - 0}; & 0 < x < 15 \\ 1; & x \leq 0 \end{cases}$$

Segitiga : sedang

$$\mu_{SedangUpper}[x] = \begin{cases} 0; & x > 45 \text{ atau } x < 5 \\ \frac{x - 5}{25 - 5}; & 5 \leq x \leq 25 \\ \frac{45 - x}{45 - 25}; & 25 < x \leq 45 \end{cases}$$

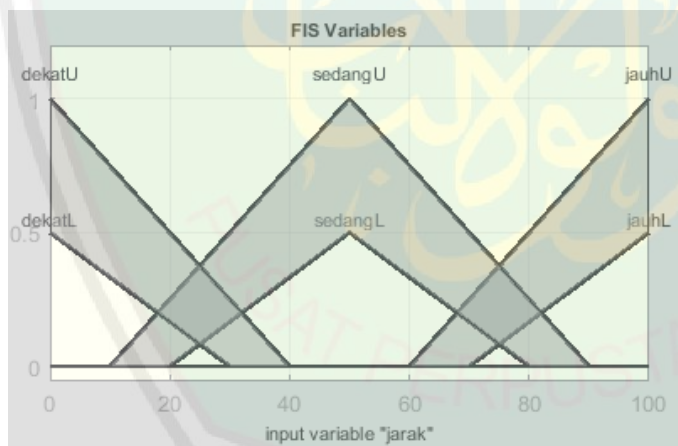
$$\mu_{SedangLower}[x] = \begin{cases} 0; & x \geq 10 \text{ atau } x \leq 40 \\ \frac{x - 10}{25 - 10}; & 10 \leq x \leq 25 \\ \frac{40 - x}{40 - 25}; & 25 < x \leq 40 \end{cases}$$

Bahu Kanan : baik

$$\mu_{baikUpper}[x] = \begin{cases} 0; & x \leq 30 \\ \frac{x - 30}{50 - 30}; & 30 < x < 50 \\ 1; & x \geq 50 \end{cases}$$

$$\mu_{baikLower}[x] = \begin{cases} 0; & x \leq 35 \\ \frac{x - 35}{50 - 35}; & 35 < x < 50 \\ 1; & x \geq 50 \end{cases}$$

2. Variabel jarak, terbagi menjadi tiga himpunan yaitu: dekat, sedang dan jauh. *Range* nilai untuk variabel kesehatan dengan rentang nilai 0 - 100 akan dijelaskan sebagai berikut ini :



Gambar 3.12 Grafik *input* variabel jarak

Dari grafik *input* variabel jarak diatas, fungsi kurva bahu kiri untuk variabel linguistik “dekat”, fungsi kurva segitiga untuk variabel linguistik “sedang” dan fungsi kurva bahu kanan mewakili variabel linguistik “jauh”, didapatkan nilai range pada setiap himpunan yaitu :

- Dekat (0 – 40) dengan nilai fungsi keanggotaan *Upper* (0 – 40) dan nilai fungsi keanggotaan *Lower* (0 – 30)
- Sedang (10 – 90) dengan nilai fungsi keanggotaan *Upper* (10 – 90) dan nilai fungsi keanggotaan *Lower* (20 – 80)
- Jauh (60 – 100) dengan nilai fungsi keanggotaan *Upper* (60 – 100) dan nilai fungsi keanggotaan *Lower* (70 – 100)

Berikut perhitungan manual dari fungsi diatas :

Bahu Kiri : dekat

$$\mu_{dekatUpper}[x] = \begin{cases} 0 ; & x \geq 40 \\ \frac{40 - x}{40 - 0} ; & 0 < x < 40 \\ 1 ; & x \leq 0 \end{cases}$$

$$\mu_{dekatLower}[x] = \begin{cases} 0 ; & x \geq 30 \\ \frac{30 - x}{30 - 0} ; & 0 < x < 30 \\ 1 ; & x \leq 0 \end{cases}$$

Segitiga : sedang

$$\mu_{SedangUpper}[x] = \begin{cases} 0 ; & x > 90 \text{ atau } x < 10 \\ \frac{x - 10}{50 - 10} ; & 10 \leq x \leq 50 \\ \frac{90 - x}{90 - 50} ; & 50 < x \leq 90 \end{cases}$$

$$\mu_{\text{SedangLower}}[x] = \begin{cases} 0; & x < 20 \text{ atau } x > 80 \\ \frac{x - 20}{50 - 20}; & 20 \leq x \leq 50 \\ \frac{80 - x}{80 - 50}; & 50 < x \leq 80 \end{cases}$$

Bahu Kanan : jauh

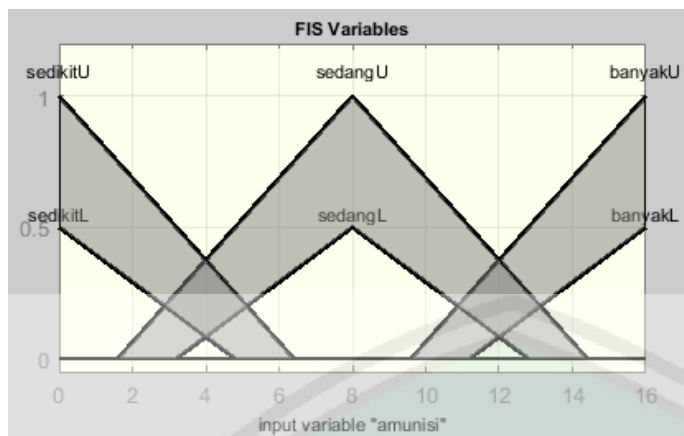
$$\mu_{\text{jauhUpper}}[x] = \begin{cases} 0; & x \leq 60 \\ \frac{x - 60}{100 - 60}; & 60 < x < 100 \\ 1; & x \geq 100 \end{cases}$$

$$\mu_{\text{jauhLower}}[x] = \begin{cases} 0; & x \leq 70 \\ \frac{x - 70}{100 - 70}; & 70 < x < 100 \\ 1; & x \geq 100 \end{cases}$$

3. Variabel amunisi, terbagi menjadi tiga himpunan yaitu: sedikit, sedang dan banyak.

Range nilai untuk variabel kesehatan dengan rentang nilai 0 - 16 akan dijelaskan sebagai

berikut ini :



Gambar 3.13 Grafik *input* variabel amunisi

Dari grafik *input* variabel jarak diatas, fungsi kurva bahu kiri untuk variabel linguistik “sedikit”, fungsi kurva segitiga untuk variabel linguistik “sedang” dan fungsi kurva bahu kanan mewakili variabel linguistik “banyak” didapatkan nilai range pada setiap himpunan yaitu :

- a. Sedikit (0 – 6) dengan nilai fungsi keanggotaan (0 – 6) dan nilai fungsi keanggotaan *Lower* (0 – 4)
- b. Sedang (2 – 14) dengan nilai fungsi keanggotaan *Upper* (2 – 14) dan nilai fungsi keanggotaan *Lower* (3 – 12)
- c. Banyak (10 – 16) dengan nilai fungsi keanggotaan *Upper* (10 – 16) dan nilai fungsi keanggotaan *Lower* (11 – 16)

Berikut perhitungan manual dari fungsi diatas :

Bahu Kiri : sedikit

$$\mu_{\text{sedikitUpper}}[x] = \begin{cases} 0; & x \geq 6 \\ 6 - x; & 0 < x < 6 \\ 6 - 0; & \\ 1; & x \leq 0 \end{cases}$$

$$\mu_{\text{sedikitLower}}[x] = \begin{cases} 0; & x \geq 4 \\ \frac{4-x}{4-0}; & 0 < x < 4 \\ 1; & x \leq 0 \end{cases}$$

Segitiga : sedang

$$\mu_{\text{SedangUpper}}[x] = \begin{cases} 0; & x < 2 \text{ atau } x > 14 \\ \frac{x-2}{8-2}; & 2 \leq x \leq 8 \\ \frac{14-x}{14-8}; & 8 < x \leq 14 \end{cases}$$

$$\mu_{\text{SedangLower}}[x] = \begin{cases} 0; & x < 3 \text{ atau } x > 12 \\ \frac{x-3}{8-3}; & 3 \leq x \leq 8 \\ \frac{12-x}{12-8}; & 8 < x \leq 12 \end{cases}$$

Bahu Kanan : banyak

$$\mu_{\text{banyakUpper}}[x] = \begin{cases} 0; & x \leq 10 \\ \frac{x-10}{16-10}; & 10 < x < 16 \\ 1 & x \geq 16 \end{cases}$$

$$\mu_{\text{banyakLower}}[x] = \begin{cases} 0; & x \leq 11 \\ \frac{x-11}{16-11}; & 11 < x < 16 \\ 1 & x \geq 16 \end{cases}$$

4. Variabel perilaku atau aksi memiliki tiga nilai linguistik, yaitu : patroli, menyerang, dan melarikan diri.

3.6.4 Proses Inferensi

Proses inferensi berupa perhitungan *rulebase* dari *input* yang telah ada untuk menentukan *output*, dimana didapatkan *rule-rule* dalam *game* yang akan dibuat sebagai berikut :

1. If (kesehatan is buruk) and (jarak is dekat) and (amunisi is sedikit) then (perilaku is melarikan diri)
2. If (kesehatan is buruk) and (jarak is dekat) and (amunisi is sedang) then (perilaku is melarikan diri)
3. If (kesehatan is buruk) and (jarak is dekat) and (amunisi is banyak) then (perilaku is menyerang)
4. If (kesehatan is buruk) and (jarak is sedang) and (amunisi is sedikit) then (perilaku is melarikan diri)
5. If (kesehatan is buruk) and (jarak is sedang) and (amunisi is sedang) then (perilaku is menyerang)
6. If (kesehatan is buruk) and (jarak is sedang) and (amunisi is banyak) then (perilaku is menyerang)
7. If (kesehatan is buruk) and (jarak is jauh) and (amunisi is sedikit) then (perilaku is patroli)
8. If (kesehatan is buruk) and (jarak is jauh) and (amunisi is sedang) then (perilaku is patroli)
9. If (kesehatan is buruk) and (jarak is jauh) and (amunisi is banyak) then (perilaku is

- patroli)
10. If (kesehatan is sedang) and (jarak is dekat) and (amunisi is sedikit) then (perilaku is melarikan diri)
 11. If (kesehatan is sedang) and (jarak is dekat) and (amunisi is sedang) then (perilaku is menyerang)
 12. If (kesehatan is sedang) and (jarak is dekat) and (amunisi is banyak) then (perilaku is menyerang)
 13. If (kesehatan is sedang) and (jarak is sedang) and (amunisi is sedikit) then (perilaku is melarikan diri)
 14. If (kesehatan is sedang) and (jarak is sedang) and (amunisi is sedang) then (perilaku is menyerang)
 15. If (kesehatan is sedang) and (jarak is sedang) and (amunisi is banyak) then (perilaku is menyerang)
 16. If (kesehatan is sedang) and (jarak is jauh) and (amunisi is sedikit) then (perilaku is patroli)
 17. If (kesehatan is sedang) and (jarak is jauh) and (amunisi is sedang) then (perilaku is patroli)
 18. If (kesehatan is sedang) and (jarak is jauh) and (amunisi is banyak) then (perilaku is patroli)
 19. If (kesehatan is baik) and (jarak is dekat) and (amunisi is sedikit) then (perilaku is melarikan diri)
 20. If (kesehatan is baik) and (jarak is dekat) and (amunisi is sedang) then (perilaku is menyerang)

21. If (kesehatan is baik) and (jarak is dekat) and (amunisi is banyak) then (perilaku is menyerang)
22. If (kesehatan is baik) and (jarak is sedang) and (amunisi is sedikit) then (perilaku is menyerang)
23. If (kesehatan is baik) and (jarak is sedang) and (amunisi is sedang) then (perilaku is menyerang)
24. If (kesehatan is baik) and (jarak is sedang) and (amunisi is banyak) then (perilaku is menyerang)
25. If (kesehatan is baik) and (jarak is jauh) and (amunisi is sedikit) then (perilaku is patroli)
26. If (kesehatan is baik) and (jarak is jauh) and (amunisi is sedang) then (perilaku is patroli)
27. If (kesehatan is baik) and (jarak is jauh) and (amunisi is banyak) then (perilaku is patroli)

3.6.5 Reduksi Tipe (*type-reduction*)

Type reduction merupakan tahap proses untuk mereduksi himpunan *interval fuzzy type-2* menjadi himpunan *fuzzy type-1*. Pada perancangan ini metode reduksi tipe yang digunakan adalah *center-of-set* dengan persamaan

$$Y = \frac{\sum_{n=1}^N f^n y^n}{\sum_{n=1}^N f^n} \equiv [yl, yr]$$

f^n merupakan hasil proses penyulutan yang berupa nilai *interval* $[f^n, \bar{f}^n]$. Dan y^n merupakan nilai *output* konsekuen yaitu $[y^n, \bar{y}^n]$.

Jika metode *type reduction* yang digunakan adalah *center-of-set*, maka dalam komputasinya bagian konsekuen digantikan oleh pusat luasannya.

3.6.6 Defuzzifikasi

Proses *defuzzifikasi* bertujuan menentukan nilai *output* tegas dari komputasi *fuzzy*.

Berikut adalah *source code* proses *defuzzifikasi* :

```
For (i = 0; I < rule.Length; i++) {
    sigmaAI += rule[i];
    sigmaAIZI += rule[i] * keputusan[i];
}
hasil = sigmaAIZI / sigmaAI;
```

3.6.7 Contoh Perhitungan

Dari 27 *rule* yang terbentuk akan dilakukan percobaan perhitungan manual terhadap algoritma *fuzzy type-2* yang telah dimodelkan, dengan nilai kesehatan 45, jarak 55 dan amunisi 8, maka tahapan untuk mendapatkan hasil keputusan adalah sebagai berikut :

3.6.7.1 Fuzzifikasi

Pada proses *fuzzifikasi* akan dilakukan proses pemetaan nilai *crisp* dari kesehatan, jarak, dan amunisi pada himpunan *fuzzy* dan menentukan derajat keanggotaan dengan fungsi *upper* dan *lower*.

Perhitungan *fuzzifikasi* untuk variabel kesehatan dengan nilai 45 :

1. Fungsi Keanggotaan *Upper* :

$$\mu_{\text{Kesehatan Buruk}} [45] = 0 ; \text{Kesehatan} \leq 20$$

$$\mu_{\text{Kesehatan Sedang}} [45] = 0 ; \text{Kesehatan} < 45 \text{ atau } \text{Kesehatan} > 5$$

$$\mu \text{ Kesehatan Baik [45]} = \frac{(45 - 30)}{50 - 30} = 0,75 ; 30 \leq \text{Kesehatan} \leq 50$$

2. Fungsi Keanggotaan *Lower* :

$$\mu \text{ Kesehatan Buruk [45]} = 0 ; \text{Kesehatan} \leq 15$$

$$\mu \text{ Kesehatan Sedang [45]} = 0 ; \text{Kesehatan} < 40 \text{ atau } \text{Kesehatan} > 10$$

$$\mu \text{ Kesehatan Baik [45]} = \frac{(45 - 35)}{50 - 35} = 0,67 ; 35 \leq \text{Kesehatan} \leq 50$$

Dari hasil perhitungan berdasarkan rumus *linear* turun, *linear* segitiga dan *linear* naik diperoleh derajat keanggotaan *upper* dan *lower* dari variabel kesehatan buruk, sedang dan baik sebagai berikut :

- a. Derajat keanggotaan *Upper* buruk [45] = 0
- b. Derajat keanggotaan *Lower* buruk [45] = 0
- c. Derajat keanggotaan *Upper* sedang [45] = 0
- d. Derajat keanggotaan *Lower* sedang [45] = 0
- e. Derajat keanggotaan *Upper* baik [45] = 0,75
- f. Derajat keanggotaan *Lower* baik [45] = 0,67

Perhitungan *fuzzifikasi* untuk variabel jarak dengan nilai 55 :

1. Fungsi Keanggotaan *Upper* :

$$\mu \text{ Jarak Dekat [55]} = 0 ; \text{Jarak} \leq 40$$

$$\mu \text{ Jarak Sedang [55]} = \frac{(90 - 55)}{90 - 10} = 0,44 ; \text{Jarak} < 90 \text{ atau } \text{Jarak} > 10$$

$$\mu \text{ Jarak Jauh [55]} = 0 ; 60 \leq \text{Jarak} \leq 100$$

2. Fungsi Keanggotaan *Lower* :

$$\mu \text{ Jarak Dekat [55]} = 0 ; \text{Jarak} \leq 30$$

$$\mu \text{ Jarak Sedang [55]} = \frac{(80-55)}{80-20} = 0,42 ; \text{ Jarak} < 80 \text{ atau } \text{ Jarak} > 20$$

$$\mu \text{ Jarak Jauh [55]} = 0 ; 70 \leq \text{ Jarak} \leq 100$$

Dari hasil perhitungan berdasarkan rumus *linear* turun, *linear* segitiga dan *linear* naik diperoleh derajat keanggotaan *upper* dan *lower* dari variabel jarak dekat, sedang dan jauh sebagai berikut :

- a. Derajat keanggotaan *Upper* dekat [55] = 0
- b. Derajat keanggotaan *Lower* dekat [55] = 0
- c. Derajat keanggotaan *Upper* sedang [55] = 0,44
- d. Derajat keanggotaan *Lower* sedang [55] = 0,42
- e. Derajat keanggotaan *Upper* jauh [55] = 0
- f. Derajat keanggotaan *Lower* jauh [55] = 0

Perhitungan *fuzzifikasi* untuk variabel amunisi dengan nilai 8 :

3. Fungsi Keanggotaan *Upper* :

$$\mu \text{ Amunisi Sedikit [8]} = 0 ; \text{ Amunisi} \leq 6$$

$$\mu \text{ Amunisi Sedang [8]} = \frac{(14-8)}{14-2} = 0,5 ; \text{ Amunisi} < 14 \text{ atau } \text{ Amunisi} > 2$$

$$\mu \text{ Amunisi Banyak [8]} = 0 ; 10 \leq \text{ Amunisi} \leq 16$$

4. Fungsi Keanggotaan *Lower* :

$$\mu \text{ Amunisi Sedikit [8]} = 0 ; \text{ Amunisi} \leq 4$$

$$\mu \text{ Amunisi Sedang [8]} = \frac{(12-8)}{12-3} = 0,44 ; \text{ Amunisi} < 12 \text{ atau } \text{ Amunisi} > 3$$

$$\mu \text{ Amunisi Banyak [8]} = 0 ; 11 \leq \text{ Amunisi} \leq 16$$

Dari hasil perhitungan berdasarkan rumus *linear* turun, *linear* segitiga dan *linear* naik diperoleh derajat keanggotaan *upper* dan *lower* dari variabel amunisi sedikit, sedang dan banyak sebagai berikut :

- a. Derajat keanggotaan *Upper* sedikit [8] = 0
- b. Derajat keanggotaan *Lower* sedikit [8] = 0
- c. Derajat keanggotaan *Upper* sedang [8] = 0,5
- d. Derajat keanggotaan *Lower* sedang [8] = 0,44
- e. Derajat keanggotaan *Upper* banyak [8] = 0
- f. Derajat keanggotaan *Lower* banyak [8] = 0

3.6.7.2 Inferensi dan Reduksi Tipe

Pada tahap ini akan dihitung nilai dari tiap variabel sesuai dengan *rule fuzzy type-2* yang telah dibuat dan mereduksi himpunan *upper* dan *lower* dengan metode *center-of-set*, yaitu sebagai berikut :

1. *If* (Kesehatan *is* Buruk) *and* (Jarak *is* Dekat) *and* (Amunisi *is* Sedikit) *then*

(Keputusan *is* Melarikan Diri)

$$Upper = (0, 0, 0) = (0)$$

$$Lower = (0, 0, 0) = (0)$$

2. *If* (Kesehatan *is* Buruk) *and* (Jarak *is* Dekat) *and* (Amunisi *is* Sedang) *then*

(Keputusan *is* Melarikan Diri)

$$Upper = (0, 0, 0.5) = (0)$$

$$Lower = (0, 0, 0.44) = (0)$$

3. *If* (Kesehatan *is* Buruk) *and* (Jarak *is* Dekat) *and* (Amunisi *is* Bayak) *then*

(Keputusan *is* Menyerang)

$$Upper = (0, 0, 0) = (0)$$

$$Lower = (0, 0, 0) = (0)$$

4. *If (Kesehatan is Buruk) and (Jarak is Sedang) and (Amunisi is Sedikit) then*

(Keputusan is Melarikan Diri)

$$Upper = (0, 0.44, 0) = (0)$$

$$Lower = (0, 0.42, 0) = (0)$$

5. *If (Kesehatan is Buruk) and (Jarak is Sedang) and (Amunisi is Sedang) then*

(Keputusan is Menyerang)

$$Upper = (0, 0.44, 0.5) = (0)$$

$$Lower = (0, 0.42, 0.44) = (0)$$

6. *If (Kesehatan is Buruk) and (Jarak is Sedang) and (Amunisi is Banyak) then*

(Keputusan is Menyerang)

$$Upper = (0, 0.44, 0) = (0)$$

$$Lower = (0, 0.42, 0) = (0)$$

7. *If (Kesehatan is Buruk) and (Jarak is Jauh) and (Amunisi is Sedikit) then*

(Keputusan is Patroli)

$$Upper = (0, 0, 0) = (0)$$

$$Lower = (0, 0, 0) = (0)$$

8. *If (Kesehatan is Buruk) and (Jarak is Jauh) and (Amunisi is Sedang) then*

(Keputusan is Patroli)

$$Upper = (0, 0, 0.5) = (0)$$

$$Lower = (0, 0, 0.44) = (0)$$

9. *If (Kesehatan is Buruk) and (Jarak is Jauh) and (Amunisi is Banyak) then*

(Keputusan is Patroli)

$$Upper = (0, 0, 0) = (0)$$

$$Lower = (0, 0, 0) = (0)$$

10. *If (Kesehatan is Sedang) and (Jarak is Dekat) and (Amunisi is Sedikit) then*

(Keputusan is Melarikan Diri)

$$Upper = (0, 0, 0) = (0)$$

$$Lower = (0, 0, 0) = (0)$$

11. *If (Kesehatan is Sedang) and (Jarak is Dekat) and (Amunisi is Sedang) then*

(Keputusan is Menyerang)

$$Upper = (0, 0, 0.5) = (0)$$

$$Lower = (0, 0, 0.44) = (0)$$

12. *If (Kesehatan is Sedang) and (Jarak is Dekat) and (Amunisi is Banyak) then*

(Keputusan is Menyerang)

$$Upper = (0, 0, 0) = (0)$$

$$Lower = (0, 0, 0) = (0)$$

13. *If (Kesehatan is Sedang) and (Jarak is Sedang) and (Amunisi is Sedikit) then*

(Keputusan is Melarikan Diri)

$$Upper = (0, 0.44, 0) = (0)$$

$$Lower = (0, 0.42, 0) = (0)$$

14. *If (Kesehatan is Sedang) and (Jarak is Sedang) and (Amunisi is Sedang) then*

(Keputusan is Menyerang)

$$Upper = (0, 0.44, 0.5) = (0)$$

$$\text{Lower} = (0, 0.42, 0.44) = (0)$$

15. *If (Kesehatan is Sedang) and (Jarak is Sedang) and (Amunisi is Banyak) then*

(Keputusan is Menyerang)

$$\text{Upper} = (0, 0.44, 0) = (0)$$

$$\text{Lower} = (0, 0.42, 0) = (0)$$

16. *If (Kesehatan is Sedang) and (Jarak is Jauh) and (Amunisi is Sedikit) then*

(Keputusan is Patroli)

$$\text{Upper} = (0, 0, 0) = (0)$$

$$\text{Lower} = (0, 0, 0) = (0)$$

17. *If (Kesehatan is Sedang) and (Jarak is Jauh) and (Amunisi is Sedang) then*

(Keputusan is Patroli)

$$\text{Upper} = (0, 0, 0.5) = (0)$$

$$\text{Lower} = (0, 0, 0.44) = (0)$$

18. *If (Kesehatan is Sedang) and (Jarak is Jauh) and (Amunisi is Banyak) then*

(Keputusan is Patroli)

$$\text{Upper} = (0, 0, 0) = (0)$$

$$\text{Lower} = (0, 0, 0) = (0)$$

19. *If (Kesehatan is Baik) and (Jarak is Dekat) and (Amunisi is Sedikit) then*

(Keputusan is Melarikan Diri)

$$\text{Upper} = (0.75, 0, 0) = (0)$$

$$\text{Lower} = (0.67, 0, 0) = (0)$$

20. *If (Kesehatan is Baik) and (Jarak is Dekat) and (Amunisi is Sedang) then*

(Keputusan is Menyerang)

$$Upper = (0.75, 0, 0) = (0)$$

$$Lower = (0.67, 0, 0) = (0)$$

21. *If (Kesehatan is Baik) and (Jarak is Dekat) and (Amunisi is Banyak) then (Keputusan is Menyerang)*

$$Upper = (0.75, 0, 0) = (0)$$

$$Lower = (0.67, 0, 0) = (0)$$

22. *If (Kesehatan is Baik) and (Jarak is Sedang) and (Amunisi is Sedikit) then (Keputusan is Menyerang)*

$$Upper = (0.75, 0.44, 0) = (0)$$

$$Lower = (0.67, 0.42, 0) = (0)$$

23. *If (Kesehatan is Baik) and (Jarak is Sedang) and (Amunisi is Sedang) then (Keputusan is Menyerang)*

$$Upper = (0.75, 0.44, 0.5) = (0.165)$$

$$Lower = (0.67, 0.42, 0.44) = (0.124)$$

$$\text{Reduksi Tipe} = \text{if (Upper} \neq 0 \ \&\& \ \text{Lower} \neq 0)$$

$$\{ (0.165 + 0.124) / 2 \} = 0.145$$

24. *If (Kesehatan is Baik) and (Jarak is Sedang) and (Amunisi is Banyak) then (Keputusan is Menyerang)*

$$Upper = (0.75, 0.44, 0) = (0)$$

$$Lower = (0.67, 0.42, 0) = (0)$$

25. *If (Kesehatan is Baik) and (Jarak is Jauh) and (Amunisi is Sedikit) then (Keputusan is Patroli)*

$$Upper = (0.75, 0, 0) = (0)$$

$$Lower = (0.67, 0, 0) = (0)$$

26. *If (Kesehatan is Baik) and (Jarak is Jauh) and (Amunisi is Sedang) then*

(Keputusan is Patroli)

$$Upper = (0.75, 0, 0.5) = (0)$$

$$Lower = (0.67, 0, 0.44) = (0)$$

27. *If (Kesehatan is Baik) and (Jarak is Jauh) and (Amunisi is Banyak) then*

(Keputusan is Patroli)

$$Upper = (0.75, 0, 0) = (0)$$

$$Lower = (0.67, 0, 0) = (0)$$

3.6.7.3 Defuzzifikasi

Proses *defuzzifikasi* bertujuan menentukan nilai *output* tegas dari komputasi yaitu sebagai berikut :

```

For (i = 0; i < rule.Length; i++) {
    sigmaAI += rule[i];
    sigmaAIZI += rule[i] * keputusan[i];
}
hasil = sigmaAIZI / sigmaAI;
if (hasil == 3){keputusan = "Menyerang" ;
} else if (hasil == 2){keputusan = "Patroli" ;
} else if (hasil == 1){keputusan = "Melarikan Diri" ;}

```

$rule[22] = sigmaAI = 0.145 ;$

$sigmaAIZI = 0.145 * 3 = 0.435 ;$

$hasil = 0.435 / 0.145 = 3 ;$

keputusan = Menyerang

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

4.1 Implementasi

Pada bab ini membahas tentang implementasi dari perencanaan *game* yang telah dibuat, serta melakukan pengujian terhadap penerapan algoritma yang digunakan untuk mengetahui apakah *game* tersebut telah berjalan sesuai dengan yang diharapkan.

4.1.1 Penggunaan Perangkat Keras

Perangkat keras yang digunakan untuk mengimplementasikan perangkat lunak dari aplikasi *game* ini, sebagai berikut:

Tabel 4.1 Penggunaan perangkat keras

No.	Perangkat Keras	Spesifikasi
1.	Processor	Intel® Core™ i3-4030U CPU @ 1.90 GHz (2 CPUs)
2.	RAM	4 GB
3.	VGA	Nvidia® GeForce 820M
4.	HDD	500 Gb
5.	Monitor	14"
6.	Speaker	On
7.	Mouse & Keyboard	On

4.1.2 Penggunaan Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan pembangunan *game* ini, sebagai berikut :

Tabel 4.2 Penggunaan perangkat lunak

No.	Perangkat Lunak	Spesifikasi
1.	Sistem Operasi	Windows 10 64-bit
2.	<i>Game Engine</i>	<i>Unity3D 5.5</i>
3.	Konsep desain 2D	Photoshop CS3
4.	Desain 3D	Blender 2.78 a
5.	<i>Script Writer</i>	Mono Develop

4.1.3 Implementasi Algoritma *Fuzzy Type-2*

Berikut akan dijelaskan penggunaan *method* dan fungsi pada tabel 4.3

Tabel 4.3 Keterangan *class fuzzy type-2*

No	Method / Fungsi	Keterangan
1	<pre>rule = new float[27]; keputusan = new float[27] {1, 1, 3, 1, 3, 3, 2, 2, 2, 1, 3, 3, 1, 3, 3, 2, 2, 2, 1, 3, 3, 3, 3, 3, 2, 2, float kesehatanUpper; float kesehatanLower; float amunisiUpper; float amunisiLower; float jarakUpper; float jarakLower; float kesehatanBaik; float kesehatanSedang;</pre>	<p><i>Method</i> yang dijalankan pertama kali saat <i>class</i> dipanggil, berguna mengambil transformasi <i>object</i> yang dimiliki oleh <i>class</i>,</p>

float kesehatanBuruk; float amunisiBanyak; float amunisiSedang; float amunisiSedikit; float jarakJauh; float jarakSedang; float jarakDekat; float kesehatanBaikUpper; float kesehatanSedangUpper; float kesehatanBurukUpper; float kesehatanBaikLower; float kesehatanSedangLower; float kesehatanBurukLower; float amunisiSedikitUpper; float amunisiSedangUpper; float amunisiBanyakUpper; float amunisiSedikitLower; float amunisiSedangLower; float amunisiBanyakLower; float jarakDekatUpper; float jarakSedangUpper; float jarakJauhUpper; float jarakDekatLower; float jarakSedangLower; float jarakJauhLower;	membangkitkan variabel yang akan digunakan untuk proses <i>fuzzy</i> , dan menentukan variabel outpun dari rule yang akan digunakan
---	---

2	<pre>void Update () { kesehatanPlayer = player.GetComponent<PlayerHealth>().currentHealth; inputKesehatan = GetComponent<EnemyHealth>().currentHealth; inputJarak = 2*(Vector3.Distance(player.transform.position, transform.position)); inputAmunisi = 1; if(inputKesehatan > 0 && kesehatanPlayer > 0){ fuzzyType2(inputKesehatan, inputJarak, inputAmunisi); } }</pre>	<p><i>Method</i> yang digunakan untuk menjalankan proses <i>fuzzy</i> secara terus-menerus (<i>update</i>) dengan melakukan inialisasi dari <i>input</i> untuk digunakan sebagai parameter <i>fuzzy</i></p>
3	<pre>void fuzzyType2(float Kesehatan, float Jarak, float Amunisi){ kesehatanUpper = Kesehatan; kesehatanLower = Kesehatan; jarakUpper = Jarak; jarakLower = Jarak; amunisiUpper = Amunisi; amunisiLower = Amunisi; }</pre>	<p><i>Method</i> yang digunakan untuk menginisialisasi parameter <i>input fuzzy</i></p>
4	<pre>if(kesehatanUpper >= 20){</pre>	<p><i>Method</i> yang</p>

	<pre> kesehatanBurukUpper = 0; } else if(kesehatanUpper < 20 && kesehatan Upper > 0){ kesehatanBurukUpper= (20 - kesehatanUpper)/(20 - 0); } else if(kesehatanUpper <= 0){ kesehatanBurukUpper = 1; } if(kesehatanLower >= 15){ kesehatanBurukLower = 0; } else if(kesehatanLower < 15 && kesehatan Lower > 0){ kesehatanBurukLower= (15 - kesehatanLower)/(15 - 0); } else if(kesehatanLower <= 0){ kesehatanBurukLower = 1; } </pre>	<p>digunakan untuk <i>fuzzyfikasi</i> trapesium turun kesehatan buruk <i>Upper</i> dan <i>Lower</i></p>
5	<pre> if(kesehatanUpper > 45 kesehatanUpper < 5){ kesehatanSedangUpper = 0; } else if(kesehatanUpper >= 5 && kesehatan Upper <= 25){ </pre>	<p><i>Method</i> yang digunakan untuk <i>fuzzyfikasi</i> segitiga kesehatan sedang <i>Upper</i> dan <i>Lower</i></p>

	<pre> kesehatanSedangUpper = (kesehatanUpper - 5)/(25 - 5); }else if(kesehatanUpper <= 45 && kesehata nUpper > 25){ kesehatanSedangUpper = (45 - kesehatanUpper) / (45 - 25); } if(kesehatanLower > 40 kesehatanLower < 10){ kesehatanSedangLower = 0; }else if(kesehatanLower >= 10 && kesehata nLower <= 25){ kesehatanSedangLower = (kesehatanLower - 10)/(25 - 10); }else if(kesehatanLower <= 40 && kesehata nLower > 25){ kesehatanSedangLower = (40 - kesehatanLower) / (40 - 25); } </pre>	
6	<pre> if(kesehatanUpper <= 30){ kesehatanBaikUpper = 0; }else if(kesehatanUpper > 30 && kesehatan </pre>	<p><i>Method yang digunakan untuk fuzzyfikasi trapesium</i></p>

	<pre> Upper < 50){ kesehatanBaik Upper = (kesehatanUpper - 30)/(50 - 30); } else if(kesehatanUpper >= 50){ kesehatanBaik Upper = 1; } if(kesehatanLower <= 35){ kesehatanBaikLower = 0; } else if(kesehatanLower > 35 && kesehatan Lower < 50){ kesehatanBaikLower = (kesehatanLower - 35)/(50 - 35); } else if(kesehatanLower >= 50){ kesehatanBaikLower = 1; } </pre>	<p>naik kesehatan baik</p> <p><i>Upper</i> dan <i>Lower</i></p>
7	<pre> if(jarakUpper >= 40){ jarakDekatUpper = 0; } else if(jarakUpper < 40 && jarakUpper > 0){ jarakDekatUpper= (40 - jarakUpper)/(40 - 0); } else if(jarakUpper <= 0){ </pre>	<p><i>Method</i> yang digunakan untuk <i>fuzzyfikasi</i> trapesium turun jarak dekat</p> <p><i>Upper</i> dan <i>Lower</i></p>

	<pre> jarakDekatUpper = 1; } Jarak Dekat Lower if(jarakLower >= 30){ jarakDekatLower = 0; } else if(jarakLower < 30 && jarakLower > 0){ jarakDekatLower= (30 - jarakLower)/(30 - 0); } else if(jarakLower <= 0){ jarakDekatLower = 1; } </pre>	
8	<pre> if(jarakUpper > 90 jarakUpper < 10){ jarakSedangUpper = 0; } else if(jarakUpper >= 10 && jarakUpper <= 50){ jarakSedangUpper = (jarakUpper - 10)/(50 - 10); } else if(jarakUpper <= 90 && jarakUpper > 50){ jarakSedangUpper = (90 - jarakUpper) / (90 - 50); </pre>	<p><i>Method yang digunakan untuk fuzzyfikasi segitiga jarak sedang Upper dan Lower</i></p>

	<pre> } if(jarakLower > 80 jarakLower < 20){ jarakSedangLower = 0; }else if(jarakLower >= 20 && jarakLower <= 50){ jarakSedangLower = (jarakLower - 20)/(50 - 20); }else if(jarakLower <= 80 && jarakLower > 50){ jarakSedangLower = (80 - jarakLower) / (80 - 50); } </pre>	
9	<pre> if(jarakUpper <= 60){ jarakJauhUpper = 0; }else if(jarakUpper > 60 && jarakUpper < 100){ jarakJauhUpper = (jarakUpper - 60)/(100 - 60); }else if(jarakUpper >= 100){ jarakJauhUpper = 1; } </pre>	<p><i>Method yang digunakan untuk fuzzyfikasi trapesium naik jarak jauh Upper dan Lower</i></p>

	<pre> if(jarakLower <= 70){ jarakJauhLower = 0; } else if(jarakLower > 70 && jarakLower < 100){ jarakJauhLower = (jarakLower - 70)/(100 - 70); } else if(jarakLower >= 100){ jarakJauhLower = 1; } </pre>	
10	<pre> if(amunisiUpper >= 6){ amunisiSedikitUpper = 0; } else if(amunisiUpper < 6 && amunisiUpper > 0){ amunisiSedikitUpper= (6 - amunisiUpper)/(6 - 0); } else if(amunisiUpper <= 0){ amunisiSedikitUpper = 1; } if(amunisiLower >= 4){ amunisiSedikitLower = 0; } else if(amunisiLower < 4 && amunisiLower > 0){ </pre>	<p><i>Method</i> yang digunakan untuk <i>fuzzyfikasi</i> trapesium turun amunisi Sedikit <i>Upper</i> dan <i>Lower</i></p>

	<pre> amunisiSedikitLower= (4 - amunisiLower)/(4 - 0); }else if(amunisiLower <= 0){ amunisiSedikitLower = 1; } </pre>	
11	<pre> if(amunisiUpper > 14 amunisiUpper < 2){ amunisiSedangUpper = 0; }else if(amunisiUpper >= 2 && amunisiUpper <= 8){ amunisiSedangUpper = (amunisiUpper - 2)/(8 - 2); }else if(amunisiUpper <= 14 && amunisiUpper > 8){ amunisiSedangUpper = (14 - amunisiUpper) /(14 - 8); } if(amunisiLower > 12 amunisiLower < 3){ amunisiSedangLower = 0; }else if(amunisiLower >= 3 && amunisiLower <= 8){ amunisiSedangLower = (amunisiLower - 3)/(8 - 3); </pre>	<p><i>Method yang digunakan untuk fuzzyfikasi segitiga amunisi sedang Upper dan Lower</i></p>

	<pre> }else if(amunisiLower <= 12 && amunisiLower > 8){ amunisiSedangLower = (12 - amunisiLower) / (12 - 8); } </pre>	
12	<pre> if(amunisiUpper <= 10){ amunisiBanyakUpper = 0; }else if(amunisiUpper > 10 && amunisiUpper < 16){ amunisiBanyakUpper = (amunisiUpper - 10)/(16 - 10); }else if(amunisiUpper >= 16){ amunisiBanyakUpper = 1; } if(amunisiLower <= 11){ amunisiBanyakLower = 0; }else if(amunisiLower > 11 && amunisiLower < 16){ amunisiBanyakLower = (amunisiLower - 11)/(16 - 11); }else if(amunisiLower >= 16){ amunisiBanyakLower = 1; } </pre>	<p><i>Method yang digunakan untuk fuzzyfikasi trapesium naik amunisi banyak Upper dan Lower</i></p>

	<pre> }</pre>	
13	<pre> if (amunisiSedikitUpper != 0 && amunisiSedikitLower != 0){ amunisiSedikit = (amunisiSedikitUpper + amunisiSedikitLower) / 2; } else if (amunisiSedikitUpper == 0 amunisiSedikitLower == 0){ amunisiSedikit = 0; } if (amunisiSedangUpper != 0 && amunisiSedangLower != 0){ amunisiSedang = (amunisiSedangUpper + amunisiSedangLower) / 2; } else if (amunisiSedangUpper == 0 amunisiSedangLower == 0){ amunisiSedang = 0; } if (amunisiBanyakUpper != 0 && amunisiBanyakLower != 0){ amunisiBanyak = (amunisiBanyakUpper + amunisiBanyakLower) / 2; } else if (amunisiBanyakUpper == 0 amunisiBanyakLower == 0){</pre>	<p><i>Method</i> yang digunakan untuk proses reduksi tipe</p>

<pre>amunisiBanyak = 0; } if (jarakDekatUpper != 0 && jarakDekatLower != 0){ jarakDekat = (jarakDekatUpper + jarakDekatLower) / 2; } else if (jarakDekatUpper == 0 jarakDekatLower == 0){ jarakDekat = 0; } if (jarakSedangUpper != 0 && jarakSedangLower != 0){ jarakSedang = (jarakSedangUpper + jarakSedangLower) / 2; } else if (jarakSedangUpper == 0 jarakSedangLower == 0){ jarakSedang = 0; } if (jarakJauhUpper != 0 && jarakJauhLower != 0){ jarakJauh = (jarakJauhUpper + jarakJauhLower) / 2; } else if (jarakJauhUpper == 0 </pre>	
---	--

<pre>jarakJauhLower == 0){ jarakJauh = 0; } if (kesehatanBurukUpper != 0 && kesehatanBurukLower != 0){ kesehatanBuruk = (kesehatanBurukUpper + kesehatanBurukLower) / 2; } else if (kesehatanBurukUpper == 0 kesehatanBurukLower == 0){ kesehatanBuruk = 0; } if (kesehatanSedangUpper != 0 && kesehatanSedangLower != 0){ kesehatanSedang = (kesehatanSedangUpper + kesehatanSedangLower) / 2; } else if (kesehatanSedangUpper == 0 kesehatanSedangLower == 0){ kesehatanSedang = 0; } if (kesehatanBaikUpper != 0 && kesehatanBaikLower != 0){ kesehatanBaik = (kesehatanBaikUpper + kesehatanBaikLower) / 2;</pre>	
---	--

	<pre> } else if (kesehatanBaikUpper == 0 kesehatanBaikLower == 0){ kesehatanBaik = 0; } </pre>	
14	<pre> rule[0] = Mathf.Min (kesehatanBuruk, jarakDekat, amunisiSedikit); rule[1] = Mathf.Min (kesehatanBuruk, jarakDekat, amunisiSedang); rule[2] = Mathf.Min (kesehatanBuruk, jarakDekat, amunisiBanyak); rule[3] = Mathf.Min (kesehatanBuruk, jarakSedang, amunisiSedikit); rule[4] = Mathf.Min (kesehatanBuruk, jarakSedang, amunisiSedang); rule[5] = Mathf.Min (kesehatanBuruk, jarakSedang, amunisiBanyak); rule[6] = Mathf.Min (kesehatanBuruk, jarakJauh, amunisiSedikit); rule[7] = Mathf.Min (kesehatanBuruk, jarakJauh, amunisiSedang); rule[8] = Mathf.Min (kesehatanBuruk, jarakJauh, amunisiBanyak); rule[9] = Mathf.Min (kesehatanSedang, </pre>	<p>Proses Inferensi untuk menentukan keputusan dari rule yang dibuat</p>

<pre>jarakDekat, amunisiSedikit); rule[10] = Mathf.Min (kesehatanSedang, jarakDekat, amunisiSedang); rule[11] = Mathf.Min (kesehatanSedang, jarakDekat, amunisiBanyak); rule[12] = Mathf.Min (kesehatanSedang, jarakSedang, amunisiSedikit); rule[13] = Mathf.Min (kesehatanSedang, jarakSedang, amunisiSedang); rule[14] = Mathf.Min (kesehatanSedang, jarakSedang, amunisiBanyak); rule[15] = Mathf.Min (kesehatanSedang, jarakJauh, amunisiSedikit); rule[16] = Mathf.Min (kesehatanSedang, jarakJauh, amunisiSedang); rule[17] = Mathf.Min (kesehatanSedang, jarakJauh, amunisiBanyak); rule[18] = Mathf.Min (kesehatanBaik, jarakDekat, amunisiSedikit); rule[19] = Mathf.Min (kesehatanBaik, jarakDekat, amunisiSedang); rule[20] = Mathf.Min (kesehatanBaik, jarakDekat, amunisiBanyak);</pre>	
--	--

	<pre> rule[21] = Mathf.Min (kesehatanBaik, jarakSedang, amunisiSedikit); rule[22] = Mathf.Min (kesehatanBaik, jarakSedang, amunisiSedang); rule[23] = Mathf.Min (kesehatanBaik, jarakSedang, amunisiBanyak); rule[24] = Mathf.Min (kesehatanBaik, jarakJauh, amunisiSedikit); rule[25] = Mathf.Min (kesehatanBaik, jarakJauh, amunisiSedang); rule[26] = Mathf.Min (kesehatanBaik, jarakJauh, amunisiBanyak); </pre>	
15	<pre> for(i = 0; i<rule.Length; i++){ sigmaAI += rule[i]; sigmaAIZI += rule[i] * keputusan[i]; } hasil = sigmaAIZI / sigmaAI; sigmaAI = 0; sigmaAIZI = 0; </pre>	<p><i>Method</i> yang digunakan untuk Defuzzifikasi berdasarkan rule yang telah ditetapkan sebelumnya</p>
16	<pre> if (hasil == 3) { print ("Amunisi : " + amunisiUpper + " Kesehatan : " + kesehatanUpper + " Jarak : " + jarakUpper + " Keputusan : </pre>	<p>Proses yang digunakan untuk menginisialisasi hasil keputusan dari</p>

<pre>Menyerang"); GetComponent<navMeshPathFinder> ().keputusan = 1; } else if (hasil == 2) { print ("Amunisi : " + amunisiUpper + " Kesehatan : " + kesehatanUpper + " Jarak : " + jarakUpper + " Keputusan : Patroli"); GetComponent<navMeshPathFinder> ().keputusan = 2; } else if (hasil == 1) { print ("Amunisi : " + amunisiUpper + " Kesehatan : " + kesehatanUpper + " Jarak : " + jarakUpper + " Keputusan : Melarikan Diri"); GetComponent<navMeshPathFinder> ().keputusan = 3; }</pre>	<p>proses defuzzifikasi</p>
--	-----------------------------

4.1.4 Implementasi *Game*

Berikut adalah tampilan *game* yang telah selesai dibuat :



Gambar 4.1 Tampilan awal *game*



Gambar 4.2 Tampilan menu *game*



Gambar 4.3 NPC patroli



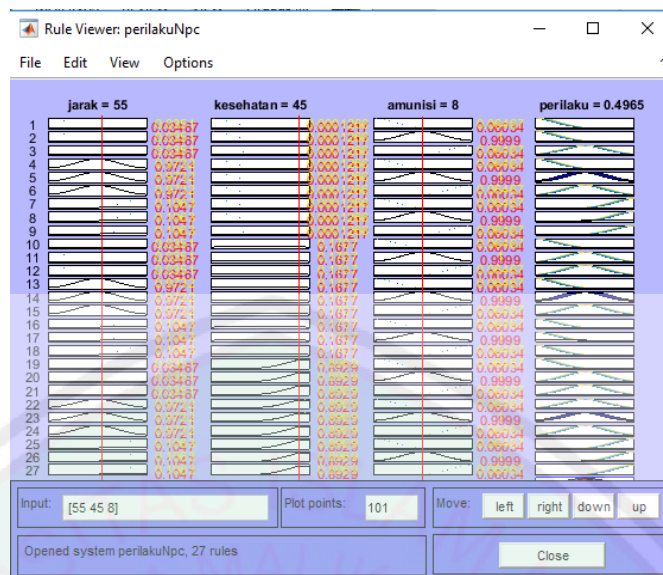
Gambar 4.4 NPC menyerang



Gambar 4.5 NPC melarikan diri

4.2 Pengujian Algoritma *Fuzzy Type-2*

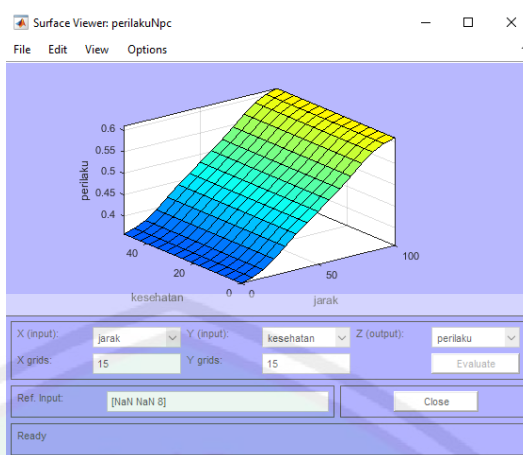
Pengujian algoritma *fuzzy type-2* dengan tiga variabel yang digunakan untuk menemukan *output* perilaku terhadap *NPC* , contoh *input* Jarak = 55, Kesehatan = 45, Amunisi = 8 disimulasikan dalam aplikasi Matlab. Berikut hasil simulasi sesuai dengan *input* di atas :



Gambar 4.6 Tampilan simulasi *output* pada Matlab

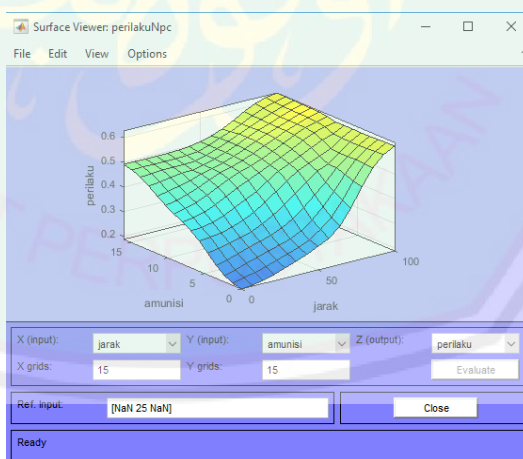
Pada gambar 4.6 menunjukkan bahwa *rule viewer* berguna untuk melihat alur penalaran *fuzzy* pada sistem, meliputi pemetaan *input* yang di berikan ke tiap variabel *input*, fungsi inferensi, reduksi tipe (*type-reduction*), sampai pada penentuan *output* pada metode defuzzifikasi. Dari gambar di atas variabel *input* di bagi menjadi tiga variabel yang digunakan dalam aturan yaitu jarak, kesehatan dan amunisi, sedangkan variabel *output* memiliki satu variabel yang di gunakan dalam aturan yaitu perilaku.

Tampilan simulasi di atas dalam pengujian algoritma *fuzzy type-2* menggunakan *rule viewer* yang disimulasikan pada aplikasi Matlab dengan menggunakan tiga variabel *input* yaitu jarak = 55, kesehatan = 45, amunisi = 8. Maka hasil keputusan perilaku terhadap *NPC* adalah 0.4965 (menyerang).



Gambar 4.7 Sumbu kartesian *input* jarak dan kesehatan

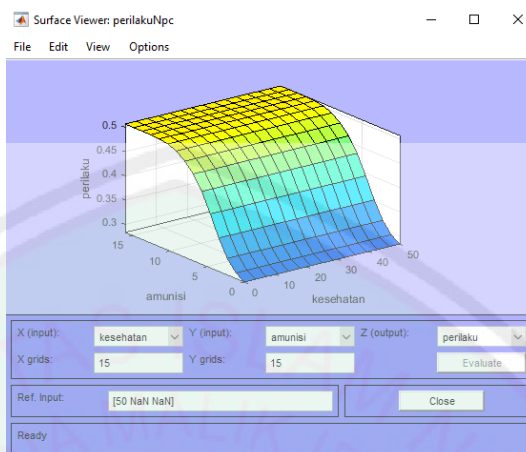
Pada gambar 4.7 menunjukkan bahwa *surface viewer* berguna untuk melihat gambar pemetaan antara variabel *input* dan variabel output. Dari *output* perhitungan Matlab dengan *input* jarak = 55, kesehatan = 45, dan amunisi = 8, menunjukkan hasil dari sumbu kartesian untuk masukan jarak terhadap kesehatan dalam *surface* dimana hasil keputusannya menunjukkan *range* 1 dari penerapan *rules* yang sudah dimasukkan



Gambar 4.8 Sumbu kartesian *input* jarak dan amunisi

Sumber Gambar 4.8 adalah hasil yang menunjukkan sumbu kartesian untuk masukan amunisi terhadap jarak dalam *surface* dimana hasil dari penerapan *rules* yang

sudah dimasukkan adalah rentang jarak terhadap amunisi pada nilai sedikit ke besar tidak terlalu berpengaruh terhadap keputusan.



Gambar 4.9 Sumbu kartesian *input* kesehatan dan amunisi

Pada Gambar 4.9 menunjukkan hasil dari sumbu kartesian untuk masukan kesehatan terhadap amunisi dalam *surface* hasil dari penerapan *rules* yang sudah dimasukkan. Kesimpulannya adalah ketika kesehatan terhadap amunisi dalam posisi banyak maka akan menghasilkan keputusan yang sangat signifikan.

Berdasarkan hasil dari pengujian dengan aplikasi Matlab di atas didapatkan beberapa perhitungan untuk menggambarkan perilaku *NPC* terhadap *player* dengan melihat tiga masukan yaitu : jarak, kesehatan dan amunisi. Berikut akan dijelaskan tentang hasil pengujian dari algoritma *fuzzy type-2* dalam bentuk tabel pada tabel 4.4 sebagai berikut :

Tabel 4.4 Pengujian *fuzzy type-2*

No	Jarak	Kesehatan	Amunisi	<i>Fuzzy Type-2</i>	<i>NPC</i>
1	25	40	8	0.4072	Menyerang
2	70	30	13	0.5599	Menyerang
3	10	20	15	0.4865	Menyerang
4	55	45	2	0.3327	Menyerang
5	40	35	11	0.4839	Menyerang
6	80	10	15	0.5877	Patroli
7	35	30	5	0.3494	Menyerang
8	40	10	2	0.2838	Melarikan Diri
9	25	45	10	0.4436	Menyerang
10	12	25	10	0.4114	Menyerang
11	78	20	7	0.5544	Patroli
12	34	39	12	0.4748	Menyerang
13	45	20	25	0.4959	Menyerang
14	85	35	10	0.5945	Patroli
15	77	48	9	0.5711	Patroli
16	90	40	14	0.6077	Patroli
17	50	35	3	0.3381	Melarikan Diri
18	35	27	9	0.4592	Menyerang
19	20	49	12	0.4573	Menyerang
20	58	33	4	0.4124	Menyerang

21	66	46	6	0.5047	Menyerang
22	44	20	15	0.494	Menyerang
23	33	40	15	0.4877	Menyerang
24	91	15	5	0.5762	Patroli
25	49	23	6	0.4376	Menyerang
26	9	45	4	0.2704	Melarikan Diri
27	22	40	15	0.4861	Menyerang
28	70	40	10	0.5542	Menyerang
29	20	48	14	0.4785	Menyerang
30	20	8	15	0.4861	Menyerang
31	89	8	15	0.6096	Patroli
32	80	10	10	0.5826	Patroli
33	70	49	15	0.556	Menyerang
34	71	49	14	0.5607	Menyerang
35	73	49	15	0.5656	Menyerang
36	75	49	15	0.5721	Patroli
37	45	36	10	0.4905	Menyerang
38	89	30	3	0.5669	Patroli
39	22	23	3	0.2721	Melarikan Diri
40	50	10	2	0.311	Melarikan Diri
41	20	10	4	0.2903	Melarikan Diri
42	21	15	6	0.341	Melarikan Diri
43	22	9	15	0.4861	Menyerang

44	18	30	2	0.2417	Melarikan Diri
45	10	35	8	0.3695	Menyerang
46	10	25	13	0.463	Menyerang
47	10	45	3	0.2462	Melarikan Diri
48	12	37	7	0.3496	Menyerang
49	12	42	15	0.4864	Menyerang
50	46	7	1	0.2812	Melarikan Diri
51	46	15	7	0.4478	Menyerang
52	50	17	12	0.5064	Menyerang
53	67	23	2	0.4044	Melarikan Diri
54	51	28	9	0.4962	Menyerang
55	65	30	13	0.5474	Menyerang
56	55	45	3	0.3611	Menyerang
57	69	46	8	0.5369	Menyerang
58	60	40	14	0.5328	Menyerang
59	85	10	2	0.5474	Patroli
60	91	15	6	0.5782	Patroli
61	88	20	12	0.6008	Patroli
62	88	37	3	0.5621	Patroli
63	95	25	7	0.5954	Patroli
64	85	22	15	0.6012	Patroli
65	90	45	2	0.5738	Patroli
66	93	40	8	0.6025	Patroli

67	87	47	13	0.5993	Patroli
68	8	49	1	0.2075	Melarikan Diri

Dari tabel tersebut dapat dilihat bahwa semua *output* sudah sesuai dengan *rule* yang telah di tentukan. Perilaku yang di hasilkan dari *output* tersebut adalah melarikan diri sebesar 17,65% , menyerang 54,41% dan patroli sebesar 27,94%.

4.3 Integrasi dalam Islam

Islam merupakan agama yang sesuai dengan fitrah manusia, syariatnya bukan saja mendorong manusia untuk mempelajari sains dan teknologi, kemudian membangun peradaban, bahkan mengatur umatnya agar selamat dan menyelamatkan baik di dunia maupun di akhirat kelak. Semua aktifitas termasuk mengkaji dan mengembangkan sains dan teknologi dapat bernilai ibadah bahkan menjadi nilai perjuangan di sisi Allah SWT. Keduanya mempunyai wilayah masing-masing, terpisah antara satu dan lainnya, baik dari segi objek formal-material, metode penelitian maupun kriteria kebenaran.

Dalam agama Islam, semua perkara sudah terdapat dalam Al-Qur'an dan diperjelas dalam Hadits dengan segala perkataan dan tingkah laku Nabi Muhammad Saw. Islam juga mengajarkan untuk tidak membuat kerusakan di muka bumi. Selain itu, Al-Quran juga ikut membicarakan tentang kerusakan di darat dan di laut adalah sebagai akibat perbuatan manusia yang dibenci Allah. Seperti peringatan dalam Al Quran Surat Al A'raf [7], ayat 46, Allah SWT berfirman :

وَلَا تُفْسِدُوا فِي الْأَرْضِ بَعْدَ إِصْلَاحِهَا وَادْعُوهُ خَوْفًا وَطَمَعًا ۗ إِنَّ رَحْمَتَ اللَّهِ قَرِيبٌ مِّنَ الْمُحْسِنِينَ

“Dan janganlah kamu membuat kerusakan di muka bumi, sesudah (Allah) memperbaikinya dan berdoalah kepadanya dengan rasa takut (tidak akan diterima) dan harapan (akan dikabulkan). Sesungguhnya rahmat Allah amat dekat kepada orang-orang yang berbuat baik.” (Q.S. Al-A’raf[7 : 56).

Sebagaimana juga anjuran untuk berjihad di jalan Allah demi mengharapkan ridho dan pahala dari Allah, maka segala bentuk tindakan terorisme wajib diperangi, yang dijelaskan dalam ayat berikut :

إِنَّ الَّذِينَ آمَنُوا وَالَّذِينَ هَاجَرُوا وَجَاهَدُوا فِي سَبِيلِ اللَّهِ أُولَٰئِكَ يَرْجُونَ رَحْمَتَ اللَّهِ وَاللَّهُ غَفُورٌ رَّحِيمٌ

“Sesungguhnya orang-orang yang beriman, orang-orang yang berhijrah dan berjihad di jalan Allah, mereka itu mengharapkan rahmat Allah, dan Allah Maha Pengampun lagi Maha Penyayang.” (Q.S Al-Baqarah[2]: 218)

Sebagaimana juga yang tertuang dalam Hadist riwayat Ahmad sebagai berikut :

Rasulullah SAW bersabda : *“Kejahatan dan perbuatan jahat, keduanya sama sekali bukan ajaran Islam. Dan orang yang paling baik Islamnya ialah yang paling baik akhlaqnya”.* [HR. Ahmad].

Maka hendaknya setiap umat manusia melakukan kebaikan di muka bumi, tidak saling bermusuhan dan menyebarkan kebencian. Karena setiap perilaku manusia akan mencerminkan akhlak mereka di hadapan sesama manusia dan Allah kelak di hari akhir.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil dari implementasi dan pengujian yang dilakukan peneliti, maka dapat ditarik kesimpulan sebagai berikut :

1. Penelitian ini berhasil membangun *game virtual reality survival shooter* perlawanan terhadap teroris yang berfungsi sebagai media hiburan dan pembelajaran tentang perlawanan terhadap aksi terorisme dengan menggunakan perangkat keras dengan spesifikasi *processor Intel® Core™ i3-4030U CPU @ 1.90GHz (2 CPUs)*, RAM 4GB, VGA *Nvidia® GeForce 820M 2GB*, HDD 500GB dan perangkat lunak yang digunakan adalah sistem operasi *Windows 10 64Bit*, *game engine Unity3d5.6*, konsep desain 2D menggunakan *Photoshop CS3*, desain 3D dengan *Blender 2.7*, *script writer* menggunakan *MonoDevelop* dan simulasi perhitungan menggunakan *Matlab r2015b*.
2. Algoritma *Fuzzy Type-2* ini digunakan untuk mengatur perilaku yang dilakukan oleh *NPC* teroris. Penelitian ini berhasil mengatur perilaku yang diterapkan pada *NPC* secara dinamis dengan 3 *variable fuzzy* menggunakan algoritma *fuzzy type-2*. *Persentase* keputusan yang dihasilkan yaitu, melarikan diri sebesar 17,65% , menyerang 54,41% dan patroli sebesar 27,94% dengan 68 *input* yang berbeda.

5.2. Saran

Dalam penelitian pembuatan *game* ini masih banyak kekurangan yang nantinya perlu untuk dilakukan pengembangan, diantaranya:

1. Resolusi gambar yang masih kurang sehingga pengalaman *virtual reality* terasa kurang.
2. *Game* ini perlu tambahan animasi yang lebih bagus sehingga apa yang akan disampaikan akan terasa lebih nyata.
3. Pengembangan *game* dengan *level* permainan yang lebih beragam dan menantang.



DAFTAR PUSTAKA

- Ady Wicaksono. 2013. Respon Agen Menggunakan *Fuzzy State Machine* Pada *Game First Person Shooter*, Tugas Akhir Tesis Mahasiswa Institut Teknologi Sepuluh Nopember Surabaya Jurusan Teknik Elektro.
- Bambang Riyanto dan Wakhyu Dwiono. 2006. Sistem Kendali *Fuzzy Bertipe-2 Interval* dengan Struktur Adaptif Beracuan Model.
- Hendi Wicaksono Agung D dan Achmad Arifin. 2009. *Fuzzy Controller Type-2* berbasis Metode *Cycle-to-Cycle* untuk Rehabilitasi *Swing Gait Phase* dengan FES. Tugas Akhir Tesis Program Pasca Sarjana Institut Teknologi Sepuluh Nopember Surabaya Jurusan Teknik Elektro.
- Hurd, Daniel dan Jenuings, Erin. 2009. *Standardized Educational Games Ratings: Suggested Criteria*. Karya Tulis Ilmiah.
- Iwan Setiawan, ST., MT. 2006. Perancangan *Software Embedded System* Berbasis *Finite State Machine*.
- Jayanti, Putri Nurul. 2013. Simulasi Sistem Kendali *Heat Exchanger* berbasis *Interval Type-2 Fuzzy Logic Controller*. Tugas Akhir Mahasiswa Universitas Islam Indonesia Yogyakarta Jurusan Teknik Elektro.
- Karnik, N. N., J. M. Mendel and Q. Liang. 1999. *Type-2 Fuzzy Logic Systems*. *IEEE on Fuzzy Systems*. vol. 7, pp. 643-658.
- Mendel, J. M. 2003. *Type-2 Fuzzy Sets : Some Questions and Answers*. *IEEE on Fuzzy Systems*. vol. 1, pp. 10-13.

- Mendel, J. M. and John, R. I. 2006. *Type-2 Fuzzy Sets Made Simple*. IEEE on Fuzzy Systems. vol. 10, pp. 808-821.
- N. N. Karnik and J. M. Mendel. 1998. *Introduction to Type-2 Fuzzy Logic Systems*. IEEE Fuzzy Conf. Anchorage, AK.
- Wijaya, Surya Adi. 2009. *Fuzzy State Machine* untuk Menghasilkan Variasi Respon NPC pada *Game*. Tugas Akhir Tesis Mahasiswa Institut Teknologi Sepuluh Nopember Surabaya Jurusan Teknik Elektro.

