

***CROWD SIMULATION UNTUK VIRTUAL REALITY PENGENALAN KAMPUS  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG  
MENGUNAKAN RECIPROCAL VELOCITY OBSTACLE***

**SKRIPSI**

**Oleh:  
GEOVANNI AZAM JANITRA  
NIM. 19650089**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2023**

***CROWD SIMULATION UNTUK VIRTUAL REALITY PENGENALAN  
KAMPUS UNIVERSITAS ISLAM NEGERI MAULANA MALIK  
IBRAHIM MALANG MENGGUNAKAN RECIPROCAL VELOCITY  
OBSTACLE***

**SKRIPSI**

**Oleh:  
GEOVANNI AZAM JANITRA  
NIM. 19650089**

**Diajukan kepada:  
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang  
Untuk memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2023**

**HALAMAN PERSETUJUAN**

***CROWD SIMULATION UNTUK VIRTUAL REALITY PENGENALAN  
KAMPUS UNIVERSITAS ISLAM NEGERI MAULANA MALIK  
IBRAHIM MALANG MENGGUNAKAN RECIPROCAL VELOCITY  
OBSTACLE***

**SKRIPSI**

**Oleh:  
GEOVANNI AZAM JANITRA  
NIM. 19650089**

Telah Diperiksa dan Disetujui untuk Diuji:  
Tanggal: 17 Mei 2023

Pembimbing I,



Dr. Yunifa Miftachul Arif, M. T  
NIP. 19830616 201101 1 004

Pembimbing II,



Dr. M. Imamudin Lc, MA  
NIP. 19740602 200901 1 010

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Ma



Dr. Faehraf Kurniawan, M.MT, IPM  
NIP. 19771020 200912 1 001

## HALAMAN PENGESAHAN

### ***CROWD SIMULATION UNTUK VIRTUAL REALITY PENGENALAN KAMPUS UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG MENGGUNAKAN RECIPROCAL VELOCITY OBSTACLE***

### SKRIPSI

Oleh:  
**GEOVANNI AZAM JANITRA**  
**NIM. 19650089**

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer ( S.Kom )  
Tanggal: 09 Juni 2023

#### Susunan Dewan Penguji

Ketua Penguji : Dr. Fresy Nugroho, M. T  
NIP. 19710722 201101 1 001

Anggota Penguji I : Hani Nurhayati, M.T  
NIP. 19780625 200801 2 006

Anggota Penguji II : Dr. Yunifa Miftachul Arif, M. T  
NIP. 19830616 201101 1 004

Anggota Penguji III : Dr. M. Imamudin Lc, MA  
NIP. 19740602 200901 1 010



Mengetahui dan Mengesahkan,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi

Universitas Islam Negeri Maulana Malik Ibrahim Ma



Dr. Faehru Kurniawan, M.MT, IPM  
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Geovanni Azam Janitra

NIM : 19650089

Fakultas / Jurusan : Sains dan Teknologi / Teknik Informatika

Judul Skripsi : *Crowd Simulation* Untuk *Virtual Reality* Pengenalan Kampus  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Menggunakan *Reciprocal Velocity Obstacle*

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 merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 09 Juni 2023  
Yang membuat pernyataan,



Geovanni Azam Janitra  
NIM.19650089

## **MOTTO**

*Sometimes we must take two little steps  
back to make one big step forward then  
keep moving forward.*

## HALAMAN PERSEMBAHAN

الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ

**Puji syukur kehadiran Allah SWT, shalawat dan salam bagi Rasul-Nya**

**Penulis persembahkan sebuah karya ini kepada:**

Keluarga penulis yang sangat dicintai, Bapak Heri Susanto yang berjasa mendidik dan selalu memberi motivasi kepada penulis untuk segera menyelesaikan tanggung jawabnya, Ibu Dwi Hapsari Febriyanti yang selalu mendoakan dan memberi nasehat kepada penulis sampai saat ini dan Adik Elvira Tsuraya Izdihar yang selalu memahami kondisi penulis ketika mengerjakan skripsi serta saling membantu.

Dosen pembimbing penulis, Bapak Dr. Yunifa Miftachul Arif, M. T dan Bapak Dr. M. Imamudin Lc, MA yang telah dengan sabar memberikan bimbingan dan masukan dalam skripsi ini. Dosen penguji penulis, Bapak Dr. Fresy Nugroho, M. T dan Ibu Hani Nurhayati, M.T yang tidak banyak memberikan revisi sulit.

Seluruh dosen Teknik Informatika UIN Maulana Malik Ibrahim Malang yang telah ikhlas membimbing dengan sabar, memberikan ilmu yang sangat bermanfaat kepada penulis dan berbagai pengalaman lainnya.

Teman-teman terdekat yang menjadi *support system* penulis yang telah membantu menyelesaikan skripsi dan memberi motivasi mengerjakan skripsi ini serta menemani disaat bermain game online dikala penulis merasa suntuk. Tak lupa juga untuk angkatan TI'19 yang tak bisa penulis sebutkan namanya satu per satu

## KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Assalamu'alaikum Wr. Wb*

*Alhamdulillah* puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis mampu menyelesaikan skripsi dengan lancar dan tepat waktu. Sholawat serta salam semoga tetap tercurahkan kepada Nabi Muhammad SAW atas syafaatnya yang telah menuntun umat manusia menuju jalan yang baik. Siapapun pengikutnya menantikan syafaatnya di hari akhir. Semoga kita semua termasuk dalam golongan yang dituntun Allah SWT dan mendapat pertolongan Nabi Muhammad SAW. Aamiin. Tujuan dari penyusunan skripsi yang berjudul “*Crowd Simulation* untuk *Virtual Reality* Pengenalan Kampus Universitas Islam Negeri Maulana Malik Ibrahim Malang menggunakan *Reciprocal Velocity Obstacle*” guna memenuhi salah satu syarat untuk bisa menempuh ujian sarjana komputer pada Fakultas Sains dan Teknologi Program Studi Teknik Informatika di 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. M. Zainuddin, M.A. selaku selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Fachrul Kurniawan, M.MT, IPM selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang beserta jajarannya.
4. Dr. Yunifa Miftachul Arif, M. T selaku Dosen Pembimbing I telah dengan sabar membimbing penulis, memberi masukan dan arahan sehingga penulis tidak hanya mampu menyelesaikan pengerjaan skripsi namun juga mengambil banyak hikmah dan pelajaran.
5. Dr. M. Imamudin Lc, MA selaku Dosen Pembimbing II telah dengan sabar membimbing penulis, memberi masukan dan arahan sehingga penulis tidak hanya mampu menyelesaikan pengerjaan skripsi namun juga mengambil banyak hikmah dan pelajaran.
6. Segenap civitas akademika Fakultas Sains dan Teknologi Universitas Islam Maulana Malik Ibrahim Malang terutama seluruh dosen, terimakasih atas bimbingan dan segala ilmu yang telah diberikan.
7. Orang tua dan keluarga penulis yang telah memberikan doa, motivasi dan dukungan sehingga penulis dapat menyelesaikan skripsi.
8. Teman-teman Jurusan Teknik Informatik Universitas Islam Negeri Maulana Malik Ibrahim Malang Angkatan 2019 yang telah memberikan dukungan kepada penulis dalam menyelesaikan penelitian skripsi ini.

9. Semua pihak yang terlibat dalam penyusunan skripsi.

*10. Last but not least, I wanna thank me. I wanna thank me for believing in me, I wanna thank me for doinng all this hard work, I wanna thank me for having no days off, I wanna thank me for never quitting, I wanna thank me for always being a giver and tryna give more than I receive.*

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan dan jauh dari kata sempurna. Maka dari itu penulis selalu membuka kesempatan seluas-luasnya untuk setiap saran dan kritik yang membangun. Doa terbaik penulis sampaikan untuk siapapun yang saat ini sedang berjuang ingin merasakan bangku kuliah demi membahagiakan orang tua, keluarga, dan yang ingin menggapai cita-cita. “Tidak ada mimpi yang gagal, yang ada hanyalah mimpi yang tertunda. Sekiranya teman-teman merasa gagal dalam menggapai mimpi, jangan khawatir. Mimpi-mimpi lain bisa diciptakan. Jangan menyerah, tetaplah berjuang, bangkit dari keterpurukan. Karena saya yakin, kita semua disini petarung untuk kehidupan yang keras ini.”

*Wassalamu'alaikum Wr. Wb*

Malang, 17 Mei 2023

A handwritten signature in black ink, appearing to be 'Gade' with a small star-like mark above the 'e'. The signature is written in a cursive, somewhat stylized font.

Penulis

## DAFTAR ISI

<b>HALAMAN PERSETUJUAN</b> .....	<b>iii</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>iv</b>
<b>PERNYATAAN KEASLIAN TULISAN</b> .....	<b>v</b>
<b>MOTTO</b> .....	<b>vi</b>
<b>HALAMAN PERSEMBAHAN</b> .....	<b>vii</b>
<b>KATA PENGANTAR</b> .....	<b>viii</b>
<b>DAFTAR ISI</b> .....	<b>xi</b>
<b>DAFTAR GAMBAR</b> .....	<b>xiii</b>
<b>DAFTAR TABEL</b> .....	<b>xv</b>
<b>ABSTRAK</b> .....	<b>xvi</b>
<b>ABSTRACT</b> .....	<b>xvii</b>
<b>الملخص</b> .....	<b>xviii</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	5
1.3 Tujuan Penelitian .....	5
1.4 Manfaat Penelitian .....	5
1.5 Batasan Masalah.....	5
1.6 Sistematika Penulisan .....	6
<b>BAB II STUDI PUSTAKA</b> .....	<b>7</b>
2.1 Landasan Teori.....	7
2.1.1 Pemodelan Berbasis Agen.....	7
2.1.2 Simulasi Kerumunan.....	8
2.1.3 Algoritma <i>Velocity Obstacle</i> .....	9
2.1.4 Algoritma <i>Reciprocal Velocity Obstacle (RVO)</i> .....	12
2.2 <i>Software</i> Pendukung .....	16
2.2.1 Blender .....	16
2.2.2 Unity.....	17
2.3 Penelitian Terkait .....	17
<b>BAB III DESAIN DAN IMPLEMENTASI</b> .....	<b>21</b>
3.1 Tahapan Penelitian .....	21
3.1.1 Studi Literatur .....	22
3.1.2 Rancangan Skenario.....	22
3.1.3 Desain Agen dan <i>Environment</i> .....	22
3.1.4 Uji Coba Simulasi .....	22
3.1.5 Penyusunan Laporan .....	23
3.2 Desain Game dan Simulasi .....	23
3.3 Implementasi Algoritma RVO .....	26
3.3.1 Parameter.....	26
3.3.2 Perhitungan RVO .....	28
3.4 Pembuatan Lingkungan Simulasi.....	33
3.4.1 Deskripsi Agen.....	36
3.4.2 Deskripsi Kampus .....	38

<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>41</b>
4.1 Tahapan Penelitian .....	41
4.1.1 Perangkat Keras .....	41
4.1.2 Perangkat Lunak.....	41
4.2 Pengujian RVO .....	42
4.2.1 <i>Seeking</i> .....	42
4.2.2 <i>Steering</i> .....	42
4.2.3 <i>Collision Avoidance</i> .....	43
4.2.4 Ekstraksi Data .....	44
4.3 Pengujian Skenario Kampus UIN Malik Ibrahim Malang.....	44
4.3.1 Arah dari Gerbang Utama .....	45
4.3.2 Arah dari Gerbang Mahad Pria .....	48
4.3.3 Arah dari Gerbang Belakang.....	51
4.3.4 Arah dari Gerbang Samping .....	54
4.3.5 Arah dari Gerbang Mahad Wanita .....	58
4.4 Integrasi Islam.....	60
4.4.1 <i>Muamalah ma'a Allah</i> .....	62
4.4.2 <i>Muamalah ma'a an-nas</i> .....	64
<b>BAB V PENUTUP.....</b>	<b>66</b>
5.1 Kesimpulan .....	66
5.2 Saran.....	67
<b>DAFTAR PUSTAKA .....</b>	<b>68</b>

## DAFTAR GAMBAR

Gambar 2. 1 <i>Velocity obstacle</i> .....	9
Gambar 2. 2 Wilayah <i>velocity obstacle</i> .....	10
Gambar 2. 3 RVO dari agen B menuju agen A.....	13
Gambar 2. 4 Osilasi.....	14
Gambar 2. 5 Agen bebas osialasi .....	15
Gambar 2. 6 Navigasi multiagen dengan RVO.....	16
Gambar 3. 1 Tahapan penelitian .....	21
Gambar 3. 2 Desain Sistem.....	23
Gambar 3. 3 Flowchart pergerakan agen .....	24
Gambar 3. 4 Flowchart deteksi tabrakan .....	26
Gambar 3. 5 Agen saling berhadapan .....	30
Gambar 3. 6 Agen hijau menghitung area collider agen kuning .....	31
Gambar 3. 7 <i>Velocity obstacle</i> agen hijau pada agen kuning .....	31
Gambar 3. 8 RVO agen hijau pada agen kuning.....	32
Gambar 3. 9 RVO agen hijau bersama agen-agen kuning .....	33
Gambar 3. 10 Wilayah skenario kampus UIN Malang tampak dari selatan .....	34
Gambar 3. 11 Wilayah skenario kampus UIN Malang tampak dari utara .....	35
Gambar 3. 12 Bentuk agen.....	37
Gambar 3. 13 FSM pada tiap agen.....	37
Gambar 3. 14 Walkable area skenario kampus UIN Malang.....	38
Gambar 3. 15 Destination skenario kampus UIN Malang .....	39
Gambar 3. 16 Obstacle skenario kampus UIN Malang.....	40
Gambar 4.1 Seeking .....	42
Gambar 4.2 Steering.....	43
Gambar 4.3 Collision Avoidance .....	44
Gambar 4.4 Preview Skenario Kampus UIN Malang .....	45
Gambar 4.5 Titik awal dan tujuan dari Gerbang Utama .....	46
Gambar 4.6 Total Collision dari Gerbang Utama .....	46
Gambar 4.7 Rata-rata <i>Framerate</i> dari Gerbang Utama .....	48
Gambar 4.8 Titik awal dan tujuan dari Gerbang Mahad Pria .....	49
Gambar 4.9 Total Collision dari Gerbang Mahad Pria .....	49
Gambar 4.10 Rata-rata <i>Framerate</i> dari Gerbang Mahad Pria .....	51
Gambar 4.11 Titik awal dan tujuan dari Gerbang Belakang.....	52
Gambar 4.12 Total Collision dari Gerbang Belakang.....	52
Gambar 4.13 Rata-rata <i>Framerate</i> dari Gerbang Belakang.....	54
Gambar 4.14 Titik awal dan tujuan dari Gerbang Samping.....	55
Gambar 4.15 Total Collision dari Gerbang Samping.....	55

Gambar 4.16 Rata-rata <i>Framerate</i> dari Gerbang Samping.....	57
Gambar 4.17 Titik awal dan tujuan dari Gerbang Mahad Wanita.....	58
Gambar 4.18 Total <i>Collision</i> dari Gerbang Mahad Wanita.....	58
Gambar 4.19 Rata-rata <i>Framerate</i> dari Gerbang Mahad Wanita .....	60

## DAFTAR TABEL

Tabel 3.1 Keterangan titik awal agen.....	36
Tabel 3.2 Keterangan titik tujuan agen .....	36
Tabel 4.1 Pengujian data dan fungsi .....	44
Tabel 4.2 Jumlah Agen dari Gerbang Utama.....	47
Tabel 4.3 Jumlah Agen dari Gerbang Mahad Pria.....	50
Tabel 4.4 Jumlah Agen dari Gerbang Belakang .....	53
Tabel 4.5 Jumlah Agen dari Gerbang Samping .....	56
Tabel 4.6 Jumlah Agen dari Gerbang Mahad Wanita.....	59

## ABSTRAK

Janitra, Geovanni. 2023. *Crowd Simulation untuk Virtual Reality Pengenalan Kampus Universitas Islam Negeri Maulana Malik Ibrahim Malang menggunakan Reciprocal Velocity Obstacle*. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Yunifa Miftachul Arif, M. T (II) Dr. M. Imamudin Lc, MA.

*Kata kunci: Crowd simulation, UIN Malang, RVO (Reciprocal velocity obstacles).*

*Crowd simulation* adalah tema penelitian yang dipakai pada aspek navigasi multi agen. Tingginya mobilitas agen serta adanya beberapa rintangan dalam simulasi mampu menimbulkan masalah pada navigasi multi agen. Maka dari itu diperlukan sebuah cara pergerakan yang mampu untuk mengarahkan agen supaya berhasil menjangkau titik tujuan dengan aman. Mayoritas orang ingin memperoleh banyak ilmu untuk menambah wawasan, atau bahkan merubah keadaan. Sehingga mengharuskan mereka melalui sebuah perguruan tinggi. UIN Maulana Malik Ibrahim merupakan salah satu instansi pendidikan tinggi yang berada di kota Malang. Banyaknya SDM yang menempuh pendidikan disana mengakibatkan tingginya mobilitas di area kampus. Di zaman modern seperti sekarang, teknologi makin canggih sehingga mampu menampilkan hampir semua dalam bentuk visual. Oleh karena itu, dibutuhkan sebuah visualisasi untuk memperkirakan keramaian atau simulasi keramaian pada area kampus UIN Maulana Malik Ibrahim Malang *Reciprocal Velocity Obstacle (RVO)* merupakan salah satu cara pada pengarahan multi agen. Dengan menggunakan RVO, setiap agen mempunyai kemungkinan untuk bergerak tanpa komunikasi antar agen lalu mampu melaksanakan penghindaran tabrakan dan tidak terkena gerakan osilasi dalam *velocity obstacle*. Penggunaan RVO diharapkan mampu membagikan kontribusi pada proses pengembangan deteksi tabrakan tiap objek dan mampu difungsikan pada metode *crowd simulation*. Hasil pada penelitian ini dapat disimpulkan bahwa metode *Reciprocal Velocity Obstacle* dapat diterapkan dalam navigasi multi agen. Hal tersebut dikarenakan agen mampu melakukan pemilihan rute, menavigasi mandiri sehingga dapat menghindari tabrakan antar agen tanpa perlunya komunikasi. Pada setiap simulasi, tabrakan akan terjadi pada sekumpulan agen karena tingkat kepadatan tinggi yang menyebabkan perhitungan kompleks.



## ABSTRACT

Janitra, Geovanni. 2023. *Crowd Simulation for Virtual Reality Introduction to Maulana Malik Ibrahim State Islamic University Campus Malang using Reciprocal Velocity Obstacle*. Theses. Department of Informatics Engineering Faculty of Science and Technology Maulana Malik Ibrahim State Islamic University of Malang. Advisor: (I) Dr. Yunifa Miftachul Arif, M. T (II) Dr. M. Imamudin Lc, MA.

Crowd simulation is a research theme used in the multi-agent navigation aspect. The high mobility of agents and the presence of several obstacles in the simulation can cause problems in multi-agent navigation. Therefore, a way of movement is needed that is able to direct agents to successfully reach the destination point safely. Most people want to gain a lot of knowledge to add insight, or even change the situation. So that requires them to go through a college. UIN Maulana Malik Ibrahim is one of the higher education institutions in the city of Malang. The large number of human resources who study there results in high mobility in the campus area. In modern times like now, technology is increasingly sophisticated so that it can display almost everything in visual form. Therefore, a visualization is needed to estimate the crowd or simulate the crowd in the campus area of UIN Maulana Malik Ibrahim Malang. Reciprocal Velocity Obstacle (RVO) is one way of directing multi agents. By using RVO, each agent has the possibility to move without communication between agents and then be able to carry out collision avoidance and not be exposed to oscillating movements in velocity obstacles. The use of RVO is expected to be able to contribute to the development process of collision detection of each object and be able to function in the crowd simulation method. The results of this research can be concluded that the Reciprocal Velocity Obstacle method can be applied in multi-agent navigation. This is because agents are able to do route selection, navigate independently so as to avoid collisions between agents without the need for communication. In each simulation, a collision will occur in a set of agents due to the high density that causes complex calculations.

*Keywords: Crowd simulation, UIN Malang, RVO (Reciprocal velocity obstacles).*

## الملخص

جانيترا، جيوفاني. ٢٠٢٣. محاكاة الحشود للواقع الافتراضي من البرنامج التعريفي لحرم جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج باستخدام عقبة السرعة المتبادلة. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: د. يونيفة مفتاح العارف، الماجستير. المشرف الثاني: د. محمد إمام الدين، الماجستير.

**الكلمات الرئيسية:** محاكاة الحشود، الجامعة الإسلامية الحكومية مالانج، *RVO* (عقبة السرعة المتبادلة).

محاكاة الحشود هي موضوع بحث يستخدم في جانب التنقل متعدد العوامل. يمكن أن يسبب التنقل العالي للوكلاء ووجود العديد من العقبات في المحاكاة مثل مشاكل في التنقل متعدد العوامل. لذلك، هناك حاجة إلى طريقة للحركة قادرة على توجيه العوامل للوصول بنجاح إلى نقطة الوجهة بأمان. يرغب معظم الناس في اكتساب الكثير من المعرفة لإضافة نظرة ثاقبة، أو حتى تغيير الموقف. لذلك يتطلب منهم الذهاب إلى الجامعة. تعد جامعة مولانا مالك إبراهيم الإسلامية الحكومية من مؤسسات التعليم العالي في مدينة مالانج. يؤدي العدد الكبير من الموارد البشرية الذين يدرسون فيها إلى تنقل عال في منطقة الحرم الجامعي. في العصر الحديث مثل الآن، أصبحت التكنولوجيا متطورة بشكل متزايد بحيث تكون قادرة على عرض كل شيء تقريباً في شكل مرئي. لذلك، هناك حاجة إلى تصور لتقدير الحشد أو محاكاة الحشد في منطقة الحرم الجامعي لجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. عقبة السرعة المتبادلة (*RVO*) هي إحدى طرق الإحاطة متعددة العوامل. باستخدام *RVO*، يتمتع كل عامل بإمكانية التحرك دون اتصال بين العوامل ومن ثم يكون قادراً على تجنب الاصطدام وعدم التعرض لحركات متذبذبة في عوائق السرعة. من المتوقع أن يكون استخدام *RVO* قادراً على المساهمة في عملية تطوير اكتشاف الاصطدام لكل كائن وأن يكون قادراً على العمل بطريقة محاكاة الحشود. ن الاستنتاج من نتائج هذا البحث أنه يمكن تطبيق طريقة عقبة السرعة المتبادلة في الملاحظة متعددة العوامل. وذلك لأن العوامل قادرون على اختيار المسار، والتنقل بشكل مستقل لتجنب الاصطدامات بينهم دون الحاجة إلى الاتصال. في كل محاكاة، سيحدث تصادم في مجموعة من العوامل بسبب الكثافة العالية التي تسبب حسابات معقدة.

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pendidikan adalah sarana yang menunjang perkembangan informasi melalui proses pembelajaran. Pasal 3 UU No. 2 pada tahun 1989 mengenai Sistem Pendidikan Nasional mempresentasikan sesungguhnya fungsi dari pendidikan kebangsaan yaitu berfungsi untuk membesarkan kesanggupan dan meluaskan derajat kehidupan dan martabat penduduk Indonesia pada usaha merealisasikan tujuan bangsa. Sistem pendidikan nasional di Indonesia diawali dari sekolah dasar menuju sekolah menengah pertama lalu sekolah menengah akhir kemudian perguruan tinggi.

Pada era sekarang, semakin banyak warga Indonesia yang menempuh pendidikan tinggi guna meningkatkan keahlian dalam berbagai bidang ilmu yang diminati. Berkaitan dengan hal tersebut, maka mengakibatkan meningkatnya mahasiswa, dosen maupun praktisi lainnya yang adalah SDM pada sekolah tinggi. (Seniati, 2006)

UIN Malang ialah universitas atau sekolah tinggi negeri islam negeri yang berlokasi di Malang. Banyaknya sumber daya manusia yang menempuh Pendidikan tinggi di UIN Malang menyebabkan tingginya mobilitas di area sekitar kampus. Hal tersebut seringkali mengakibatkan kemacetan dan kerumunan. Maka dari itu, perlunya sebuah sistem pencitraan guna mempermudah mahasiswa dan warga sekitar. Visualisasi adalah suatu pemanfaatan teknologi komputer yang berfungsi

untuk menemukan metode paling baik dalam menampilkan sebuah data. (Mauludi, 2013)

Berdasarkan yang telah ada pada Al-Qur'an Surat Yunus ayat 101 yang berbunyi :

قُلْ انظُرُوا مَاذَا فِي السَّمٰوٰتِ وَالْاَرْضِ يَوْمَ تُغْنِي الْاٰيٰتُ وَالنُّذُرُ عَنْ قَوْمٍ لَا يُؤْمِنُوْنَ

*“Katakanlah, “Perhatikanlah apa yang ada di langit dan di bumi!” Tidaklah bermanfaat tanda-tanda (kebesaran Allah) dan rasul-rasul yang memberi peringatan bagi orang yang tidak beriman.” (QS. Yunus : 101).*

Makna ayat tersebut adalah sesungguhnya Allah senantiasa menginstruksikan umat manusia perlu selalu mengamati segala hal yang terjadi di langit dan bumi. Segala hal tersebut yakni bukti kebesaran Allah SWT dan dapat kita rasakan manfaatnya melalui berbagai penelitian untuk memajukan Ilmu Pengetahuan dan Teknologi (IPTEK)

Pada Tafsir Kementerian Agama dijelaskan bahwasanya tafsiran dari surat Yunus ayat 101 yakni Allah memerintahkan kepada Nabi Muhammad, katakanlah kepada mereka, Perhatikanlah ciptaan Allah, yaitu apa saja yang ada di langit dan di bumi! Jika mereka mau menggunakan akal mereka untuk memikirkan tanda-tanda kebesaran dan kekuasaan Allah, tentu mereka sudah beriman. Namun mereka enggan melakukannya, sehingga tidaklah bermanfaat tanda-tanda kebesaran Allah dan rasul-rasul yang memberi peringatan bagi orang yang tidak beriman, karena mereka menutup hati mereka untuk menerima kebenaran. Kalau keberadaan tanda-tanda kebesaran Allah di langit dan di bumi serta diutusny para rasul tidak juga menjadikan mereka beriman, maka mereka tidak menunggu-nunggu kecuali

kejadian-kejadian yang sama dengan kejadian-kejadian masa lalu, yakni azab yang menimpa orang-orang terdahulu sebelum mereka. Katakanlah, wahai Nabi Muhammad Maka tunggulah kedatangan azab itu, sesungguhnya aku pun termasuk orang yang menunggu bersama kamu untuk menyaksikan ketetapan Allah itu menimpa kamu (Kementrian Agama, 2022).

Sehubungan dengan makna tersebut, kesimpulan relasi dari ayat tersebut dengan penerapan RVO yakni senantiasa melaksanakan anjuran Allah SWT untuk mengamati sekitar. Metode RVO pada Kampus UIN Malang menyimulasikan pergerakan keramaian di dalam area kampus. Hal tersebut seiring dengan anjuran pada surat Yunus ayat 101 karena sesungguhnya setiap apa yang ada di Bumi ini merupakan tanda kebesaran Allah SWT. Melakukan pengamatan ciptaan Allah melalui metode ilmiah merupakan bentuk keterbukaan hati terhadap ciptaanNya dan memperkuat iman kita.

*Virtual Reality (VR)* adalah teknologi simulasi komputer yang saling berhubungan dengan gabungan *device input-output* yang berfungsi guna memberikan stimulus pada satu indera atau lebih supaya user mampu bersosialisasi secara mendalam dengan lingkungan *virtual* yang seolah-olah berada dalam dunia nyata (Mihelj, Novak, & Beguš, 2014) . *Virtual Reality* merupakan sarana yang terdiri dari tiruan komputer yang saling berhubungan dengan menyangkutkan satu atau lebih dari beberapa indra sehingga users mampu merasakan dan tenggelam secara mental akibat dari simulasi virtual yang terjadi (Sherman & Craig, 2003). *Virtual* yang memiliki arti dekat sedangkan *reality* memiliki arti seluruh hal yang bersifat riil atau nyata. Di kamus besar bahasa Indonesia (KBBI) ditafsirkan sebagai

realitas maya. Maka dari itu, Virtual Reality dapat diartikan sebagai salah satu teknologi yang dapat mereplika keadaan sama persis dengan kondisi aslinya. User seakan-akan masuk di dalam lingkungan nyata dalam bentuk lingkungan 3D serta memungkinkan untuk berinteraksi dengan berbagai objek didalamnya.

Pada visualisasi mobilitas sumber daya manusia di area kampus UIN Malang, sistem simulasi kerumunan merupakan salah satu cara yang efektif. Simulasi kerumunan adalah sebuah pemodelan dari suatu gerakan yang dilakukan oleh kerumunan entitas bergerak atau agen pada jumlah yang banyak dan tingkat mobilitas yang tinggi (Fachri, 2016). Simulasi kerumunan memiliki beberapa metode sesuai algoritma penghindaran tabrakan timbal balik yang optimal atau ORCA, hambatan kecepatan timbal balik atau RVO, *A star* dan *Velocity Obstacle*. Dari sekitar metode yang tercantum, RVO mempunyai keunggulan pada saat melaksanakan gerakan yang bebas osilasi. Selain itu, RVO mengasumsikan tanggapan tingkah laku dari agen dengan dugaan agen lain akan melakukan pengelakan benturan atau hantaman yang sama sehingga memperoleh respon aktivitas agen yang bebas dari hantaman dan *osilasi*. (Furqon, 2019)

Berdasarkan permasalahan dan metode yang sudah dijelaskan oleh penulis maka diperlukan sebuah penelitian untuk mencari solusi kerumunan di area kampus UIN Malang menggunakan metode *Reciprocal Velocity Obstacle* (RVO). Observasi ini memiliki maksud untuk menerapkan simulasi keramaian dalam model 3 dimensi karena penulis berasumsi bahwa visualiasi tersebut akan menjadi realistis serta sangat membantu saat diimplementasikan kedalam suatu simulasi kerumunan atau *crowd simulation* di area kampus UIN Malang.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut dapat diputuskan sebuah rumusan masalah pada riset ini yaitu : Bagaimana merealisasikan tindakan agen yang menjauhi *obstacle* berupa objek statis atau agen yang bergerak untuk menghindari saling hantaman memakai Algoritma *Reciprocal Velocity Obstacle* (RVO) ?

## 1.3 Tujuan Penelitian

Berdasarkan rumusan masalah tersebut dapat ditentukan tujuan dari observasi yang dilakukan yaitu menciptakan tindakan agen yang menjauhi *obstacle* berbentuk objek statis atau bergerak untuk menghindari saling berbenturan menggunakan Algoritma *Reprocal Velocity Obstacle* (RVO).

## 1.4 Manfaat Penelitian

Penelitian ini menghasilkan sebuah pengembangan dalam pendeteksian dari perilaku tabrakan antar objek atau agen di bidang animasi dalam teknologi komputer dan mampu dipakai sebagai metode simulasi kerumunan atau *crowd simulation* di area kampus UN Malang.

## 1.5 Batasan Masalah

Batasan masalah dalam riset yang dilakukan adalah seperti berikut :

1. Simulasi kerumunan hanya dibatasi pada kasus pergerakan di area Kampus UIN Malang
2. Simulasi hanya dilakukan dalam kasus kerumunan agen yang berjumlah 200 agen dalam waktu 200 detik

3. Simulasi yang divisualisasikan akan dikembangkan menggunakan platform Unity 3D desktop.

### **1.6 Sistematika Penulisan**

Sistematika penulisan pada penelitian ini terbagi dalam beberapa bab sebagai berikut :

**BAB I PENDAHULUAN :** Berisi tentang latar belakang, rumusan masalah, tujuan, manfaat dan batasan masalah pada penelitian yang dikerjakan.

**BAB II STUDI PUSTAKA :** Berisi tentang tinjauan pustaka, landasan teori serta metode penelitian yang digunakan

**BAB III DESAIN DAN PENERAPAN :** Berisi tentang perancangan desain simulasi serta penerapan algoritma *reciprocal velocity obstacle*.



## **BAB II**

### **STUDI PUSTAKA**

#### **2.1 Landasan Teori**

##### **2.1.1 Pemodelan Berbasis Agen**

Agen adalah komponen yang melakukan pengambilan keputusan dalam sebuah sistem yang kompleks. Biasanya, agen memiliki sebuah paket aturan atau tindakan yang digunakan untuk melakukan proses terhadap informasi. Proses yang dilakukan meliputi penerimaan, input dan pengaruh terhadap lingkungan luar.(Saputri et al., 2014)

Pemodelan berbasis agen atau lebih dikenal dengan *Agent-based Modelling* adalah simulasi yang memiliki acuan pada sebuah model yaitu proses dinamis antar agen disimulasikan berulang kali dari waktu ke waktu. Simulasi dilakukan seperti dalam dinamika sistem, langkah waktu , peristiwa yang saling tidak berhubungan dan jenis simulasi lainnya. Simulasi umumnya memiliki konsep interaksi agen yang berulang-ulang. Misalnya, agen melakukan optimalisasi perilaku kolektif dengan menukar informasi sederhana seperti dalam optimalisasi koloni semut atau dalam optimalisasi gerombolan partikel (Macal & North, 2008). Tujuan dari optimalisasi tersebut adalah untuk mencapai keadaan yang diinginkan menjadi lebih optimal dan realistis. Dalam pemodelan berbasis agen, semua yang terlibat mampu mengambil keputusan sendiri secara otonom berdasarkan dari aturan yang ada dalam sebuah ruang lingkup yang telah ditentukan.

### 2.1.2 Simulasi Kerumunan

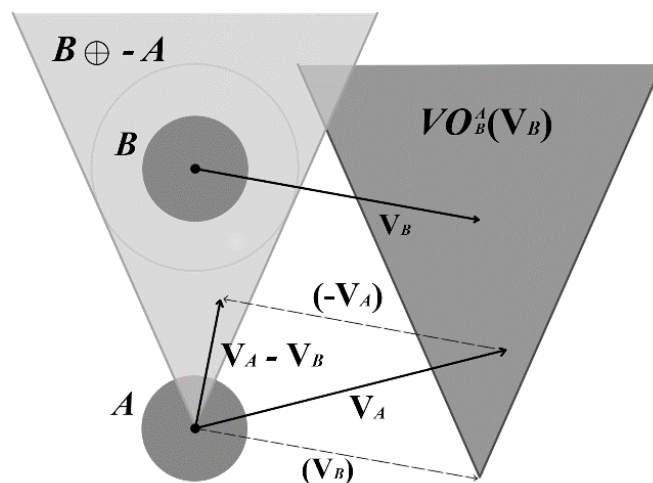
Simulasi kerumunan biasanya dimodelkan dalam kumpulan agen yang saling berinteraksi. Perilaku kerumunan juga seringkali bersifat kontinum seperti mematuhi hukum fuida dinamika. Perilaku yang dimungkinkan adalah adanya properti yang muncul dari suatu individu agen yang dipengaruhi oleh agen sekitar. Perilaku individu ini didefinisikan dengan mengaplikasikan beberapa aturan yang memiliki orientasi tujuan sederhana, diantaranya seperti bergerak dengan kecepatan rata-rata agen sekitar dan menjaga jarak optimal antar agen (Thalman & Musse, 2012)

Menurut (Furqon, 2019), pemodelan kerumunan meliputi beberapa hal yang berkaitan, salah satunya adalah pergerakan individu agen dalam lingkungannya dengan batasan atau rintangan yang tersedia. Selain itu, interaksi dinamis antar agen juga menjadi faktor utama. Namun tidak hanya kedua hal tersebut, model juga diharapkan mampu menunjukkan cerminan *intelligent path planning* atau perencanaan jalan secara intelek melalui perubahan yang terjadi. Selanjutnya, agen yang berada di dalam lingkungan tersebut harus mampu menyesuaikan jalan yang akan ditempuh secara dinamis. Berdasarkan pernyataan tersebut dapat disimpulkan bahwa simulasi kerumunan memiliki beberapa aspek utama yaitu *path planning* atau perancangan jalan, *collision avoidance* atau penghindaran hantaman, dan *separation* atau pemisahan jarak. Ketiga aspek tersebut sangat diperhatikan untuk mewujudkan realitas simulasi dan model kerumunan yang nyata.

### 2.1.3 Algoritma Velocity Obstacle

Menurut (Snape et al., 2011), dalam penelitian “*Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation*”. *Velocity obstacle* adalah gabungan dari seluruh kelancaran agen yang dapat menyebabkan hantaman dari agen lain, yang memiliki anggapan agen lain menjaga kelancaran yang terkini sekarang. *Velocity obstacle* biasanya digunakan untuk menavigasikan objek diantara rintangan yang bergerak.

Dapat diasumsikan bahwa  $A$  merupakan agen yang melakukan translasi pada titik point yang direfrensikan di  $P_A$  dan  $B$  menjadi *obstacle* planar (bergerak) yang memiliki titik point yang direfrensikan di  $P_B$ . *Velocity obstacle*  $VO_B^A(V_B)$  dari *obstacle* atau halangan  $B$  menuju agen  $A$  merupakan kumpulan set yang terjadi dari seluruh *velocities* atau kelancaran  $V_A$  untuk  $A$  pada waktu yang sama akan menghasilkan hantaman dengan waktu *obstacle* atau halangan  $B$  yang bergerak pada kecepatan  $V_B$ . *Velocity obstacle* secara geometris dapat didefinisikan seperti Gambar 2.1



Gambar 2. 1 *Velocity obstacle*

$$A \oplus B = \{a + b | a \in B\}, -A = \{-a | a \in A\} \quad (2. 1)$$

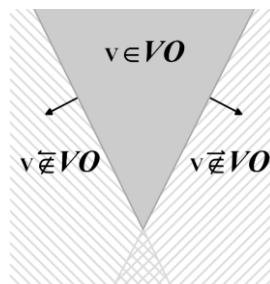
$A \oplus B$  menunjukkan jumlah Minkowski dari dua objek  $A$  dan  $B$ .  $-A$  menunjukkan objek  $A$  yang tercermin dalam titik refrensinya. Jika  $\lambda(p,v)$  menunjukkan *ray*  $A$  yang dimulai dari  $P$  dan menuju ke arah  $V$  :

$$\lambda (p, v) = \{p + tv | t \geq 0\} \quad (2. 2)$$

Jika *ray* yang dimulai pada  $P_A$  dan menuju ke arah *relative velocity* atau kecepatan *relative*  $A$  dan  $B$  (yaitu  $V_A - V_B$ ) memotong penjumlahan Minkowski yang berasal dari  $B$  dan  $-A$  yang terpusat di  $P_B$ , kecepatan  $V_A$  berlokasi pada  $VO$  atau hambatan kecepatan  $B$ . Oleh karena itu,  $VO$  dari  $B$  ke  $A$  dapat diinterpretasikan seperti berikut :

$$VO_B^A(V_B) = \{V_A | \lambda (P_A, V_A - V_B) \cap B \oplus -A \neq \emptyset\} \quad (2. 3)$$

Ini menunjukkan bahwa jika  $V_A \in VO_B^A(V_B)$ ,  $A$  dan  $B$  akan bertabrakan. Jika  $V_A$  berada di batas dari *velocity obstacle* atau rintangan kecepatan, maka  $A$  akan menyentuh  $B$  di waktu tertentu. Namun kedua objek dapat tidak akan pernah bertabrakan apabila  $V_A$  berada di luar *velocity obstacle* atau rintangan kecepatan  $B$ . *Velocity obstacle* adalah sebuah kerucut dengan puncaknya di  $V_B$ . Seperti Gambar 2.2



Gambar 2. 2 Wilayah velocity obstacle

Rancangan dari *velocity obstacle* mampu difungsikan guna mengarahkan di antara *obstacle* atau halangan yang bergerak. Pada setiap siklus perencanaan, agen memilih kelancaran yang bertepatan di luar dari *velocity obstacle* yang diinduksikan dari *obstacle* atau halangan yang bergerak. Agen akan aman menavigasikan tujuannya apabila *velocity* yang dipilih diarahkan ke posisi tujuan agen di antara *free velocity*

Rancangan *Velocity Obstacle* mampu dipakai guna mengarahkan multiagen dengan tiap agen mengasumsikan agen lain menjadi halangan bergerak serta memutuskan kelancaran bagi dirinya sendiri yang bertepatan di luar kecepatan hambatan yang ditimbulkan dari agen lain. Akan tetapi, penghampiran tersebut menciptakan sikap osilasi yang tidak diharapkan.

Kesimpulan sifat dasar dan notasi dari *velocity obstacle* sebagai berikut:

a. Lemma 1 (*Symmetry*)

$$V_A \in VO_B^A(V_B) \Leftrightarrow V_B \in VO_A^B(V_A) \quad (2.4)$$

b. Lemma 2 (*Translation invariance*)

$$V_A \in VO_B^A(V_B) \Leftrightarrow V_B \in VO_A^B(V_A) \quad (2.5)$$

Melalui peninjauan area di luar kecepatan hambatan, maka dapat membedakan area menuju sebelah kiri dan area menuju sebelah kanan dari kecepatan hambatan ditentukan dengan setengah bidang yang dibatasi oleh dua batas *velocity obstacle*.

$$V_A \overleftarrow{\notin} VO_B^A (V_B) \quad (2.6)$$

Apabila  $V_A$  bertepatan dalam setengah bidang di bagian kiri  $VO_B^A (V_B)$ . *Velocity* seperti itu memungkinkan  $A$  menyebrangi  $B$  di sisi kiri.

$$V_A \overrightarrow{\notin} VO_B^A (V_B) \quad (2.7)$$

Apabila  $V_A$  bertepatan dalam setengah bidang di bagian kanan  $VO_B^A (V_B)$ . *Velocity* seperti itu memungkinkan  $A$  menyebrangi  $B$  di sisi kanan. Dapat diperhatikan bahwa setengah bidang kiri dan kanan tumpang tindih sehingga *velocity* pada wilayah ini memungkinkan  $A$  dan  $B$  menyimpang. Lemma 2 dan 3 juga berlaku untuk bagian-bagian di luar *velocity obstacle* sehingga dapat disimpulkan bahwa ‘ $\in$ ’ pada lemma 2 dan 3 dapat dengan bebas digantikan oleh ‘ $\notin$ ’, ‘ $\overleftarrow{\notin}$ ’, atau ‘ $\overrightarrow{\notin}$ ’.

c. Lemma 3 (*Convexity*)

$$V_A \overrightarrow{\notin} VO_B^A (V_B) \wedge V'_A \overrightarrow{\notin} VO_B^A (V_B) \Rightarrow \quad (2.8)$$

$$(1 - \alpha)V_A + \alpha V'_A \overrightarrow{\notin} VO_B^A (V_B), \text{ for } 0 \leq \alpha \leq 1.$$

#### 2.1.4 Algoritma Reciprocal Velocity Obstacle (RVO)

Pada tahap ini, penulis akan menjelaskan mengenai konsep yang disebut dengan *reciprocal velocity obstacle* dengan memberikan pendekatan sederhana yang berguna untuk menvaigasikan *multiagen* dengan aman dan lancar tanpa komunikasi eksplisit antara agen.

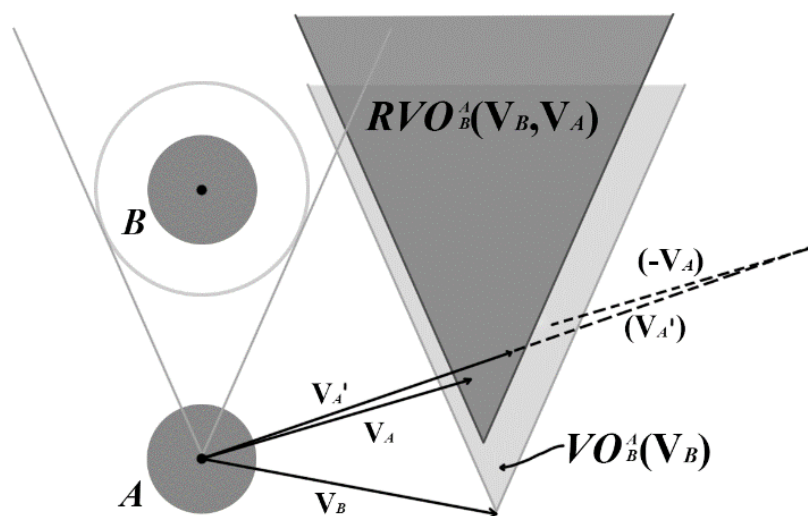
a. Pengertian

Menurut *Cambride Dictionary*, *reciprocal* adalah tindakan atau pengaturan timbal balik melibatkan dua orang atau kelompok orang yang berperilaku dengan

cara yang sama atau setuju untuk saling membantu dan saling memberi keuntungan (Cambridge University, 2022). *Reciprocal velocity obstacle* merupakan konsep dari *velocity obstacle* yang sudah dikembangkan sehingga mampu mengatasi hantaman reaktif lokal yang dipakai pada pengarahannya multiagen, dimana tiap-tiap agen mengarahkan dan menentukan rute secara mandiri tanpa komunikasi eksplisit antar agen. Algoritma RVO menggunakan pendekatan yang mengasumsikan sikap reaktif dari agen lain dengan anggapan bahwa agen lain melakukan tindakan pencegahan hantaman yang serupa dengan haluan mengakibatkan tindakan agen tanpa ada hantaman, baik itu antar agen lain atau halangan atau *obstacle*.

Memilih kelancaran baru berupa pertengahan kecepatan arus serta kelancaran yang bertepatan di luar *velocity obstacle* agen lainnya adalah bagian mendasar dari RVO. RVO didefinisikan sebagai berikut:

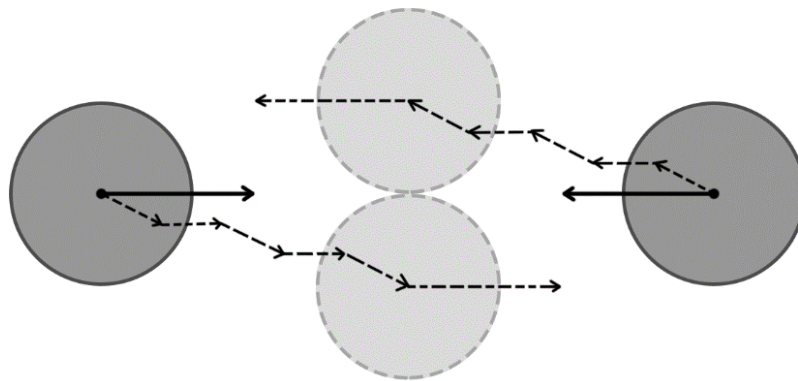
$$RVO_B^A(V_B, V_A) = \{V'_A | 2V'_A - V_A \in VO_B^A(V_B)\} \quad (2.9)$$



Gambar 2. 3 RVO dari agen B menuju agen A

Pada Gambar 2.3 memperlihatkan *reciprocal velocity obstacle*  $RVO_B^A$  ( $V_B, V_A$ ) berasal objek B menuju objek A yang berisi semua *velocity* objek A merupakan rata-rata *velocity* arus  $V_A$  dan juga *velocity* yang terdapat pada  $VO_B^A$  ( $V_B$ ) dari objek B. Menurut geometris  $RVO_B^A$  mampu diinterpretasikan menjadi *velocity obstacle*  $VO_B^A$  kemudian ditranslasikan agar nilai puncak berada pada  $\frac{V_A+V_B}{2}$ .

b. Osilasi



Gambar 2. 4 Osilasi

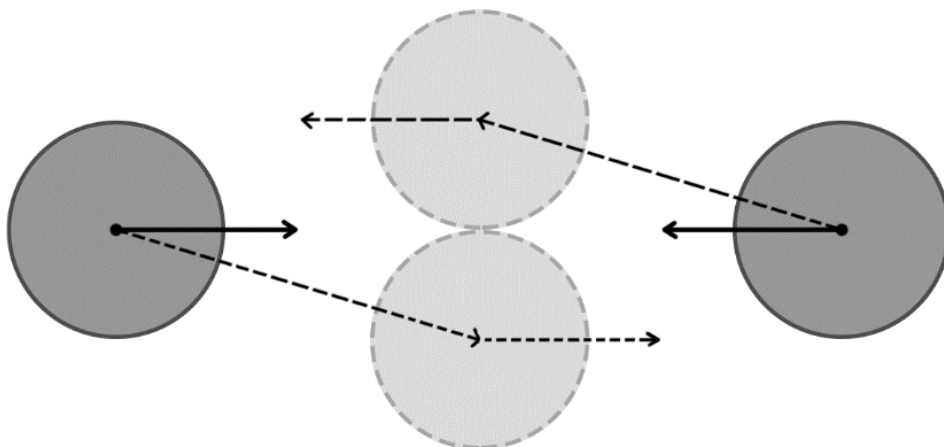
Pada Gambar 2.4 menunjukkan dua objek A dan B sedang berpindah tempat dengan *velocity* masing-masing yaitu  $V_A$  dan  $V_B$  secara bersamaan maka dari itu  $V_A \in VO_B^A$  ( $V_B$ ) dan  $V_B \in VO_A^B$  ( $V_A$ ) yang jika dilanjutkan maka akan terjadi sebuah benturan atau tabrakan. Tabrakan atau benturan tersebut dapat dihindari dengan cara agen A mengubah kecepatannya menjadi  $V_A$  yang berada di luar *velocity obstacle* B ( $V_A \notin VO_B^A$  ( $V_B$ )) dan diikuti oleh agen B yang juga merubah kecepatannya menjadi  $V_B$  yang berada di luar *velocity obstacle* A B ( $V_B \notin VO_A^B$  ( $V_A$ )).



Masalah terlihat ketika dua agen dalam kecepatan yang lama yaitu  $V_A$  dan  $V_B$  yang sudah berada di luar *velocity obstacle* A dan B. Dengan kecepatannya yang lama, kedua agen memiliki jarak menuju titik tujuan terpendek. Oleh sebab itu, kedua agen akan kembali menggunakan kecepatannya yang lama. Kedua agen akan bertabrakan dengan menggunakan kecepatannya yang lama pada siklus berikutnya yang menyebabkan kecepatan dari dua agen tersebut akan berubah lagi menjadi  $V_A$  dan  $V_B$ . Siklus ini akan terus terjadi hingga kedua agen dapat berpapasan. Perubahan kecepatan yang terus-menerus inilah yang menimbulkan gerak osilasi seperti Gambar 2.4.

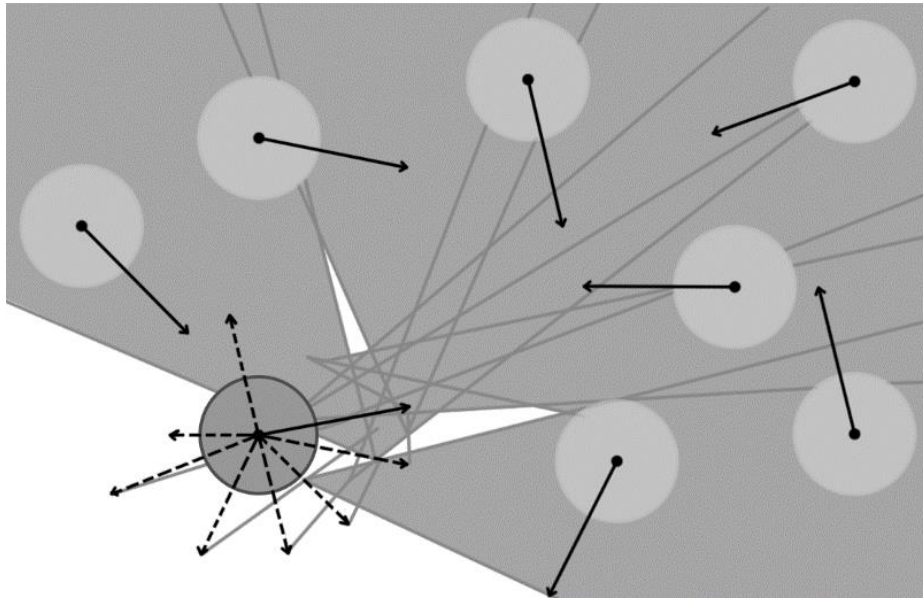
c. Multiagen navigasi Reciprocal Velocity Obstacle (RVO)

*Velocity* yang terdapat di luar kumpulan seluruh wilayah *reciprocal obstacle* dan paling mendekati dengan *velocity* yang diharapkan adalah pilihan yang akan dipilih oleh agen berdasarkan *valid velocity* dari RVO. Dapat dilihat pada Gambar 2.5



Gambar 2. 5 Agen bebas osialasi

Dalam Gambar 2.6 menunjukkan area gerak pada arsiran gelap memperlihatkan *invalid velocity* bagi agen berwarna gelap yang merupakan *union* dari RVO dari agen berwarna cerah.



Gambar 2. 6 Navigasi multiagen dengan RVO

## 2.2 Software Pendukung

Observasi ini memerlukan perangkat lunak yang dipakai dalam mengembangkan *system*, diantaranya adalah sebagai berikut.

### 2.2.1 Blender

Blender merupakan perangkat lunak yang digunakan untuk mengkreasikan sebuah karya 3 dimensi yang bersifat gratis dan *open source*. Aplikasi ini mendukung semua proses pembuatan 3D seperti animasi, *modelling*, simulasi, *rendering*, *motion tracking* dan *compositing*. Selain itu, aplikasi blender juga mendukung pengeditan video dan pengembangan *game*. Para pengguna blender

professional menggunakan aplikasi ini sebagai aplikasi yang dapat digunakan untuk mensimulasikan visualisasi 3D, serta tayangan yang memiliki kualitas baik. Sedangkan penggabungan 3D langsung dapat menciptakan konten 3D yang interaktif. Selain beberapa hal tersebut, blender memiliki berbagai macam kegunaan termasuk pemodelan, pemolesan yang menjiwai, pembentukan tekstur, *rigging* atau menggerakkan, dan masih banyak lagi. (Zebua et al., 2020)

### 2.2.2 Unity

Unity adalah *software* yang dipakai untuk pengembangan *game*, arsitektur bangunan dan pensimulasian. *Unity* merupakan *game engine* yang dibuat oleh perusahaan Unity Technologies Inc. Aplikasi ini dapat digunakan pada untuk membuat game berbasis *desktop* dan *games* berbasis *online*. Unity menyediakan sebuah plugin, seperti *Unity Web Player* yang dapat digunakan untuk mengembangkan *game* berbasis *online*. Pada pengembangannya aplikasi ini mendukung beberapa pemrograman yaitu; *JavaScript*, *C#*, dan *Boo*. (Mongi et al., 2018)

### 2.3 Penelitian Terkait

(Furqon, 2019) dalam sebuah makalah yang membahas mengenai penerapan Algoritma RVO untuk simulasi kerumunan pada alun – alun kota batu menyajikan metode *reciprocal velocity obstacle* yang disimulasikan menggunakan *crowd simulation*. *Reciprocal velocity obstacle* digunakan untuk memberikan kontribusi dengan deteksi tabrakan antar objek pada simulasi yang dilakukan. Hasil dari simulasi menampilkan bahwa akan timbul tabrakan dalam jumlah keramaian

tertentu yang dipengaruhi oleh kepadatannya sehingga memperoleh perhitungan yang kompleks untuk agen. Sedangkan satu durasi simulasi akan terus meningkat diiringi dengan banyaknya jumlah agen dan juga luas area simulasi serta jumlah rintangan berpengaruh terhadap waktu yang diperlukan untuk simulasi.

(Juniastuti et al., 2017) pada observasi yang membahas tentang simulasi keramaian menggunakan algoritma *leader-follower* berbasis RVO, menghasilkan sikap bebas osilasi serta hantaman dalam pengarahan multi-agen. Melalui kombinasi RVO dan *leader following*, tiap-tiap agen bernavigasi mengikuti leadernya guna mencapai tujuannya. Hasil studi ini menampilkan bahwa pengarahan melalui ratusan agen pada lingkungan padat masyarakat 18% lebih cepat daripada pengarahan tanpa koneksi untuk mencapai haluan massal.

(Syawab, 2020) pada makalahnya yang membahas penggunaan algoritma *artificial bee colony* pada simulasi pergerakan pengunjung di pulau Pari. *Crowd simulation* digunakan sebagai simulasi keadaan pulau Pari guna memprogramkan skenario marketing penjualan kedepannya melalui media game yang atraktif dan menarik. Menggunakan teknologi *Artificial intelligence* dengan algoritma *artificial bee colony* guna mengontrol *Non-Playable Character* (NPC) karena algoritma tersebut mampu menangani pergerakan yang kompleks.

(Muhammad et al., 2018) melaksanakan obeservasi *crowd simulation* dalam entitas penyelamatan bencana dengan pembahasan simulasi evakuasi keramaian pada agen yang berbeda-beda menggunakan RVO. Tujuan dari riset yang telah dilakukan adalah memiliki fungsi menyelidiki kesanggupan manusia melarikan diri ketika terjebak pada suatu bencana memakai cara hambatan RVO berbasis agen

yang digunakan. Hasil simulasi memperlihatkan bahwa mensimulasikan ratusan agen pada 3 *scene* penyelamatan memakai relasi *leader-follower* mengarah pada probabilitas tabrakan tiap-tiap agen yang lebih tinggi.

(Douthwaite et al., 2019) pada penelitiannya yang berjudul “*Velocity Obstacle Approaches for Multi-Agent Collision Avoidance*” menyajikan analisis kritis dari beberapa pendekatan yang paling menjanjikan untuk penghindaran tabrakan geometris dalam sistem multi-agen, yaitu, *velocity obstacle* (VO), *reciprocal velocity obstacle* (RVO), *hybrid-reciprocal velocity obstacle* (HRVO) dan *optimal reciprocal collision avoidance* (ORCA). Setiap pendekatan dievaluasi sehubungan dengan peningkatan populasi agen dan asumsi penginderaan variabel. Dalam menerapkan masalah penghindaran lokal, penulis mencatat masalah simetri yang tidak dipertimbangkan dalam literatur. Ketidakpastian dalam lintasan rintangan terbukti meningkatkan waktu komputasi rata-rata dari semua pendekatan yang diusulkan tanpa langkah-langkah kompensasi. Metode HRVO dan ORCA terbukti lebih efektif dalam menegosiasikan lingkungan yang berantakan hambatan sambil menahan ketidakpastian dalam lintasan rintangan. Metode ORCA juga terbukti menghasilkan lintasan resolusi yang lebih halus daripada metode lain yang disajikan meskipun toleransi rendah terhadap ketidakpastian hambatan. HRVO terbukti kompetitif secara statistik dengan ORCA dalam kemungkinan tabrakan, dengan toleransi yang lebih tinggi terhadap ketidakpastian hambatan. Manfaat dari pendekatan ORCA bagaimanapun, dapat dengan jelas dilihat dalam skalabilitasnya namun, menghasilkan waktu komputasi jelas lebih rendah daripada metode lainnya.

(Liu et al., 2018) pada penelitiannya yang berjudul “*Avoidance of High-speed Obstacles Based on Velocity Obstacle*” mengusulkan algoritma *two-period velocity obstacle* yang layak, di mana satu periode memprediksi potensi tabrakan dalam bagian waktu terbatas dan periode kedua meramalkan tabrakan di luar bagian itu. Periode kedua diaktifkan hanya ketika kecepatan bergerak rintangan lebih besar dari kecepatan maksimum robot. *Velocity obstacle* baru dibangun di setiap langkah yang dapat memprediksi tabrakan di luar bagian waktu metode berbasis *velocity obstacle* yang ada. Karena *velocity obstacle* baru ini diaktifkan hanya ketika kecepatan rintangan lebih besar dari kecepatan maksimum robot, algoritmanya dapat diterima secara komputasi, lebih sensitif terhadap kecepatan rintangan dibandingkan dengan versi aslinya dan lebih kuat dalam menangani rintangan dinamis daripada versi halus dari metode VO seperti ORCA.

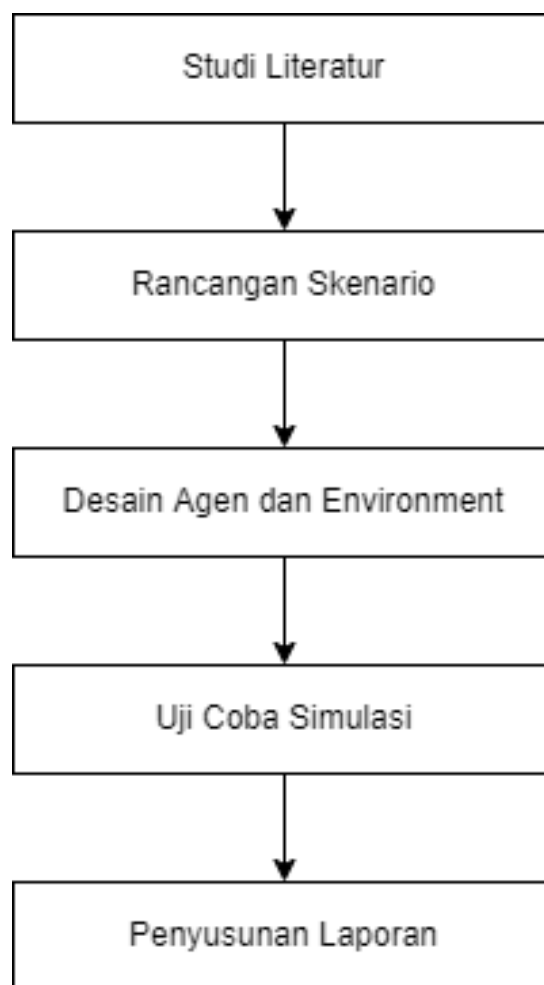
(Fadila & Arif, 2020) pada penelitiannya yang berjudul “Implementasi Algoritma RVO sebagai Sistem Kendali Gerombolan NPC pada Permainan *Action RPG*” mengimplementasikan algoritma RVO pada NPC di dalam *game action* agar NPC dapat menyerang pemain tanpa saling bertabrakan satu dengan yang lainnya. Ketika pemain mendekat, agent-agent atau NPC akan berkumpul dan berkoordinasi untuk menyerang pemain. Pada simulasi yang dilakukan terlihat bahwa gerombolan NPC dengan jumlah yang cukup banyak mampu bergerak dengan baik menuju target. Namun kekurangan dari RVO sendiri yaitu kurang mampu untuk menangani NPC dengan jumlah ribuan.

## BAB III

### DESAIN DAN IMPLEMENTASI

#### 3.1 Tahapan Penelitian

Adapun tahapan penelitian dalam penelitian ini digambarkan serasi seperti bagan alur riset sebagai berikut.



Gambar 3. 1 Tahapan penelitian

### 3.1.1 Studi Literatur

Dalam langkah ini penulis melakukan beragam himpunan simulasi terkait dengan sejumlah perkara, adalah:

- a. Pencarian keterangan mengenai universitas
- b. Pencarian keterangan mengenai produksi 3D *simulation* dengan mengaplikasikan Unity3D
- c. Pencarian keterangan mengenai cara *reciprocal velocity obstacle*
- d. Pencarian keterangan mengenai penelitian terkait

### 3.1.2 Rancangan Skenario

Dalam tahap ini, penulis merancang bagaimana tiruan akan berjalan memakai cara *reciprocal velocity obstacle*.

### 3.1.3 Desain Agen dan *Environment*

Dalam langkah ini, penulis merancang bagaimana desain *interface* untuk simulasi dan desain untuk tiap agen.

### 3.1.4 Uji Coba Simulasi

Uji coba pada riset ini akan dilaksanakan dalam simulasi serta karakter agen sudah mengimplementasikan algoritma *reciprocal velocity obstacle*. Penulis membangun program memakai *Unity engine* dengan bahasa C#. Penulisan *script* dibuat dengan memakai *text editor*. Sesudah itu dilakukan proses pengecekan untuk *output* yang dihasilkan apakah sudah sesuai dengan yang diinginkan

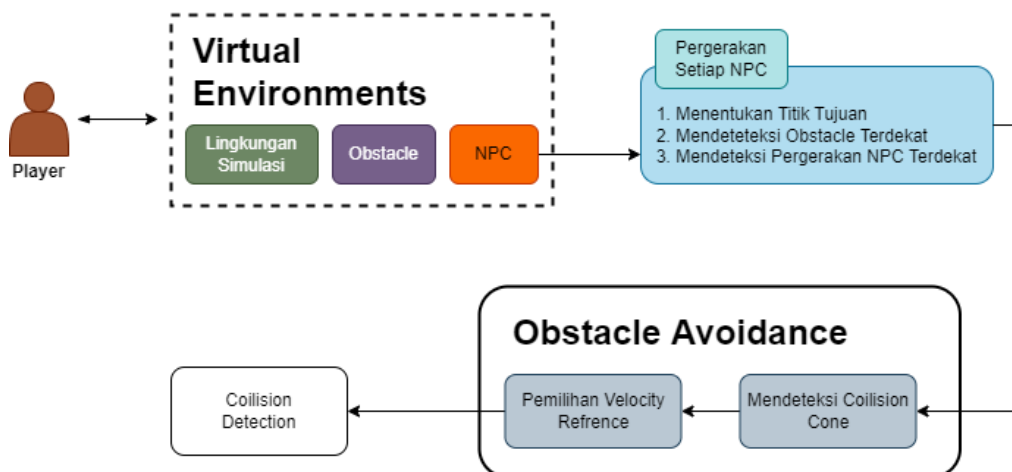


### 3.1.5 Penyusunan Laporan

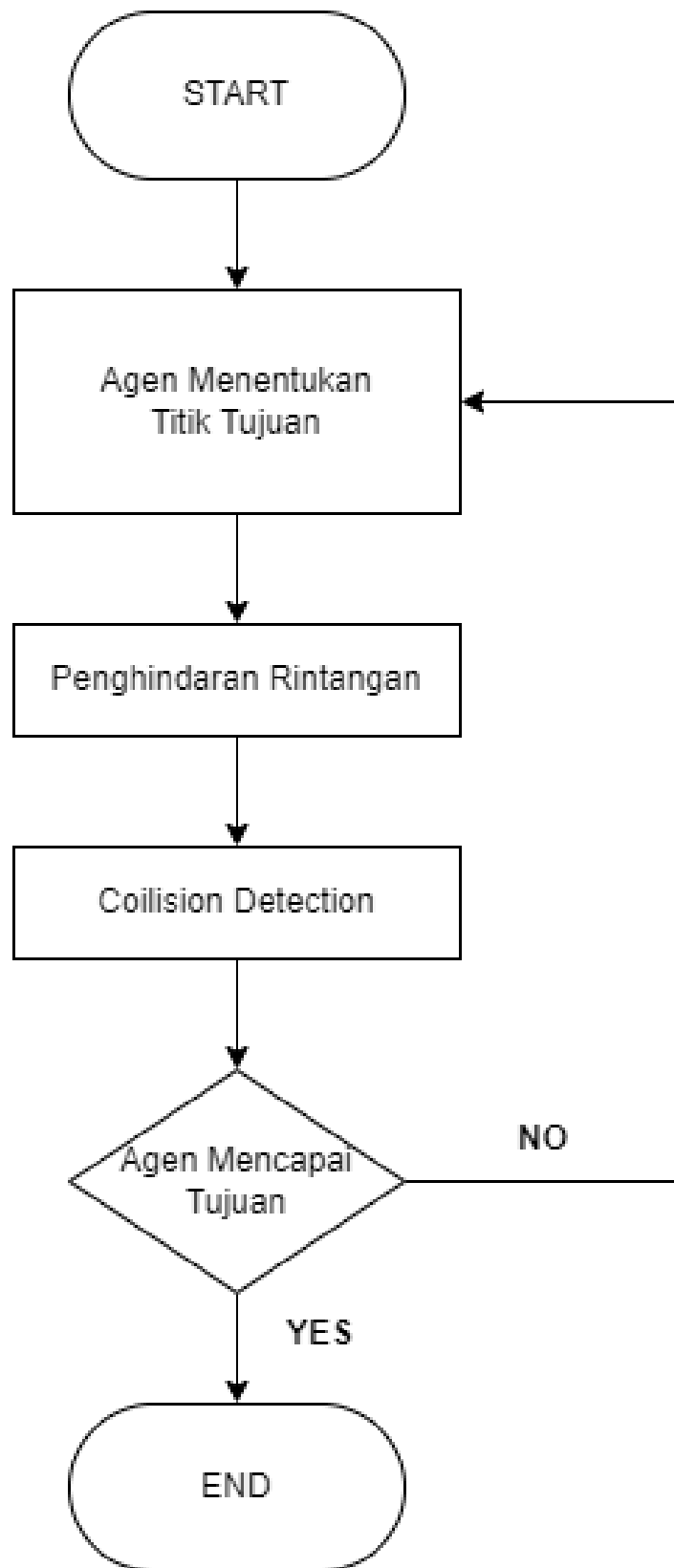
Untuk mendokumentasikan dari keseluruhan pelaksanaan pada penelitian ini, maka penulis melakukan penyusunan laporan supaya bermanfaat untuk penelitian selanjutnya.

### 3.2 Desain Game dan Simulasi

Untuk menerapkan algoritma *reciprocal velocity obstacle* maka dilakukan *crowd simulation* di area kampus universitas. Pada simulasi terdapat sebuah player dan karakter NPC berupa agen dimana agen tersebut adalah civitas kampus yang sedang berkeliling di area kampus dan sudah menggunakan algoritma *reciprocal velocity obstacle*. Penggunaan cara ini berfungsi agar sikap pada agen ketika berada di dalam keramaian dapat menjauhi atau tidak saling bertabrakan saat bergerak menuju titik tujuannya. Gambar 3.2 adalah tahapan desain sistem-nya



Gambar 3. 2 Desain Sistem



Gambar 3. 3 Flowchart pergerakan agen

a Agen menentukan titik tujuan

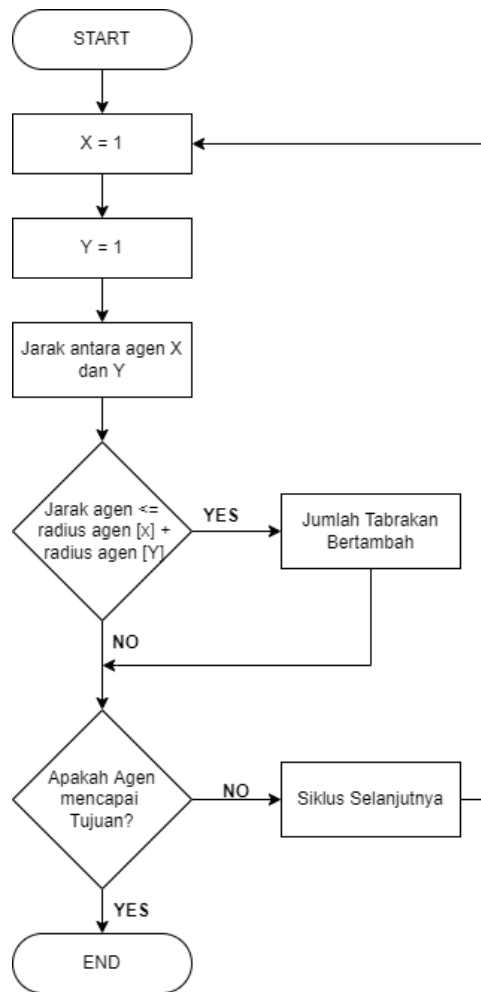
Penentuan destinasi atau titik tujuan yang diinginkan oleh setiap setiap agen setelah itu mendapatkan kecepatan referensi untuk bergerak menuju titik tujuan.

b Penghindaran Rintangan (*Obstacle avoidance*)

Pada tahap penghindaran rintangan (*obstacle avoidance*) yaitu dengan mencegah terjadinya benturan atau tabrakan pada hambatan tidak bergerak. Dengan menentukan pergerakan agen yang benar berdasarkan kelancaran yang valid dengan algoritma RVO maka penghindaran dapat dilakukan. *Velocity* referensi dapat dikatakan valid jika *velocity* referensi tersebut tidak berada di *collision cone* dari RVO dan tidak menyebabkan *collision*. *Velocity* referensi dapat dikatakan tidak valid jika *velocity* tersebut akan menyebabkan benturan atau hantaman serta *velocity* agen akan ditentukan berdasarkan *velocity* yang berada di luar himpunan *velocity* RVO. Namun jika *velocity* referensi sudah resmi menjadi kecepatan agen pada periode tersebut, kecepatan referensi dapat dinyatakan berlaku. Hasil dari tahap ini yaitu untuk memilih kelancaran agen yang akan diaplikasikan pada periode waktu itu.

c Deteksi Tabrakan (*Collision Detection*)

Bagian final yaitu *collision detection* atau mendeteksi benturan atau tabrakan. Deteksi lingkaran-ke-lingkaran adalah algoritma pendeteksi tabrakan yang akan digunakan. Apabila jangka antara dua agen adalah tidak lebih dari total jari-jari dua agen, maka agen tersebut dianggap bertabrakan.



Gambar 3. 4 Flowchart deteksi tabrakan

### 3.3 Implementasi Algoritma RVO

#### 3.3.1 Parameter

Penggunaan parameter berfungsi untuk menganalisis efektivitas dari navigasi multi agen menggunakan algoritma RVO pada *crowd simulation*. Berikut merupakan parameter yang digunakan pada simulasi penelitian ini:

a. Jumlah agen

Jumlah agen adalah data akumulasi agen yang digunakan pada saat pengujian simulasi. Penggunaan total agen pada setiap simulasi yang diujikan berbeda-beda

b. Waktu simulasi

Waktu simulasi adalah data yang mampu mendemonstrasikan waktu yang dibutuhkan saat melaksanakan simulasi. Penentuan waktu simulasi dilakukan sebelum dilakukannya simulasi dan simulasi dinyatakan selesai jika agen telah mencapai titik tujuan selaras dengan waktu yang sudah diatur. *Stuck* dapat terjadi apabila agen tidak mampu menuju tujuannya sehingga mengakibatkan agen tidak bisa meneruskan perjalanan. Keterangan waktu simulasi diperoleh melalui cara mencatat keterangan waktu yang terdapat dalam *unity* ketika simulasi itu berlangsung.

c. Jumlah *frame*

Jumlah *frame* adalah data yang menampilkan akumulasi *frame* yang terlihat saat simulasi dilakukan. Pada tiap *frame* memperlihatkan *cycle* perhitungan pergerakan agen. Jumlah *frame* pada saat simulasi berlangsung dapat diperoleh dengan menggunakan fitur *update per-frame* pada aplikasi *unity* pada saat simulasi dilakukan.

d. Rata-rata *framerate*

Rata-rata *framerate* adalah keterangan dari rata-rata akumulasi gambar per-detik. Data tersebut diperoleh dengan cara membagi akumulasi *frame* saat simulasi berlangsung dengan waktu simulasi yang dilakukan.

$$\Sigma \text{FPS} = \frac{\Sigma \text{Frame}}{\Sigma \text{Waktu Simulasi}} \quad (3.1)$$

e. Jumlah *collision*

Jumlah *collision* adalah data dari akumulasi benturan atau tabrakan yang terjadi saat simulasi dilakukan. Data jumlah *collision* diperoleh berdasarkan akumulasi total tabrakan yang berlangsung ketika simulasi berjalan.

### 3.3.2 Perhitungan RVO

Setelah data terkumpulkan berdasarkan parameter yang digunakan, langkah berikutnya yaitu melakukan perhitungan algoritma RVO. Dalam tahap ini akan dilakukan pembuktian dari algoritma RVO yang berfungsi untuk membuat pergerakan bebas hambatan serta bebas osilasi.

a. Pengarahan bebas tabrakan

Kita anggap bahwa  $V_A$  adalah *velocity* sekarang dari agen A dan  $V_B$  adalah *velocity* sekarang dari agen B. Kemudian tiap agen A dan agen B menetapkan *velocity* barunya yaitu  $V_A'$  dan  $V_B'$  yang berada di luar masing-masing RVO. Dua agen tersebut tidak akan berbenturan dan memilih bagian yang serupa untuk saling melintas dengan menggunakan lemma berikut.

$$V_A \in VO_B^A(V_B) \Leftrightarrow V_B \in VO_A^B(V_A) \quad (3.2)$$

$$V_A \in VO_B^A(V_B) \Leftrightarrow V_A + U \in VO_B^A(V_B + U) \quad (3.3)$$

$$V_A \notin VO_B^A(V_B) \wedge V_A' \notin VO_B^A(V_B) \quad (3.4)$$

$$\Rightarrow (1 - a)V_A + aV_A' \notin VO_B^A(V_B) \text{ for } 0 \leq a \leq 1$$

$$V'_A \vec{\notin} RVO_B^A(V_B, V_A) \wedge V'_B \vec{\notin} RVO_A^B(V_A, V_B) \quad (3.5)$$

$$V'_A \vec{\notin} VO_B^A(V'_B) \wedge V'_B \vec{\notin} VO_A^B(V'_A) \quad (3.6)$$

$$V'_A \notin RVO_B^A(V_B, V_A) \wedge V'_B \notin RVO_A^B(V_A, V_B) \quad (3.7)$$

$$RVO_A^B(V_B, V_A) = \{V'_A | 2V'_A - V_A \in VO_B^A(V_B)\} \quad (3.8)$$

Apabila kita mengimplementasikannya pada rumus RVO akan seperti berikut:

$$2V'_A - V_A \notin VO_B^A(V_B) \wedge 2V'_B - V_B \notin VO_A^B(V_A) \quad (3.9)$$

$$2V'_A - V_A \notin VO_B^A(V_B) \wedge V_A \notin VO_B^A(2V'_B - V_B) \quad (3.10)$$

$$2V'_A - V_A - V_B \notin VO_B^A(0) \wedge V_A = V_B - 2V'_B \notin VO_B^A(0) \quad (3.11)$$

$$V'_A - V_B \notin VO_B^A(0) \quad (3.12)$$

$$V'_A \notin VO_B^A(V'_B) \quad (3.13)$$

$$V'_A \notin VO_B^A(V'_B) \wedge V'_B \notin VO_A^B(V'_A) \quad (3.14)$$

#### b. Pengambilan sisi sama

Jika kedua agen tersebut menentukan kecepatan yang berada di luar hambatan kecepatan timbal balik agen lain yang paling dekat dengan *velocity* sekarang, maka RVO mampu membuat kedua agen tersebut menentukan bagian yang sama secara otomatis untuk saling melewati. Apabila untuk agen A adalah  $V_A + u$  merupakan kecepatan yang paling dekat dengan  $V_A$  yang berada di luar

hambatan kecepatan timbal balik B, hingga untuk agen B adalah  $V_B + u$  merupakan *velocity* paling dekat  $V_B$  yang berada di luar hambatan kecepatan timbal balik A.

$$V_A + U \notin RVO_B^A(V_B, V_A) \Leftrightarrow V_B - U \notin RVO_A^B(V_A, V_B) \quad (3.15)$$

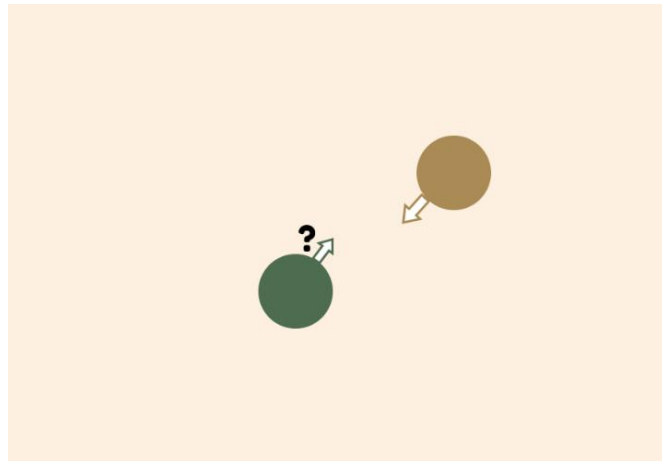
Hal tersebut dapat ditunjukkan dengan  $V_A + U \notin RVO_B^A(V_B, V_A)$

$$\Leftrightarrow 2(V_A + U) - V_A \notin VO_B^A(V_B) \quad (3.16)$$

$$\Leftrightarrow 2(V_B - U) - V_B \notin VO_A^B(V_A) \quad (3.17)$$

$$\Leftrightarrow V_B - U \notin VO_A^B(V_A) \quad (3.18)$$

Pada simulasi yang dilakukan mengharuskan agen-agen untuk mampu menghindari benturan atau tabrakan dengan agen lainnya. Berikut adalah tahap-tahap dalam melakukan simulasi.

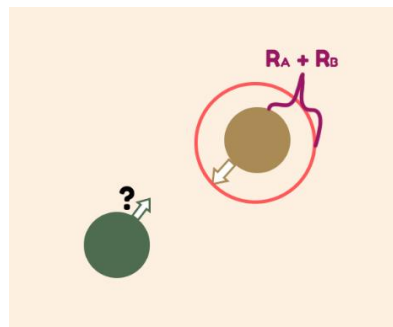


Gambar 3. 5 Agen saling berhadapan

Dari Gambar 3.5 menampilkan dua agen dengan warna coklat dan hijau. Agen berwarna hijau memandang agen kuning sebagai sebuah *obstacle* karena keberadaan agen kuning merupakan salah satu agen yang paling dekat dengan agen

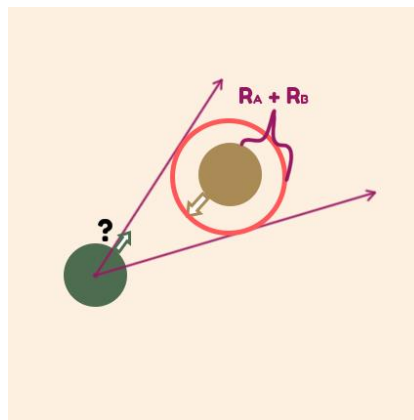


hijau. *Velocity obstacle* agen hijau terhadap agen kuning sehingga secara geometri diartikan dengan  $A \oplus B = \{a + b | a \in A, b \in B\}$ ,  $-A = \{-a | a \in A\}$ . Objek A dan B dijumlahkan menggunakan penjumlahan Minkowski yang dilambangkan dengan  $A \oplus B$ . Setelah itu agen hijau melakukan perhitungan himpunan dari seluruh *velocity* yang akan bertabrakan melalui agen kuning apabila agen tetap tidak bergerak.



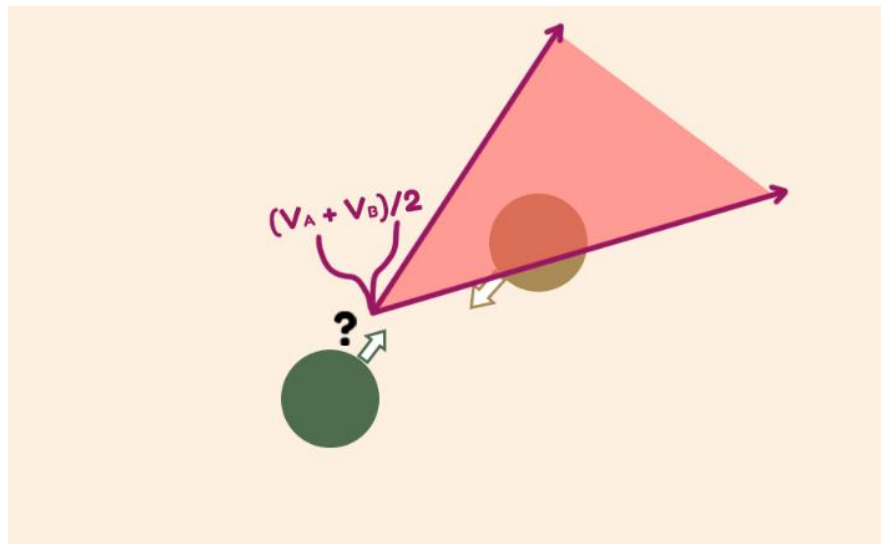
Gambar 3. 6 Agen hijau menghitung area collider agen kuning

Pada Gambar 3.6 menunjukkan bahwa agen hijau melakukan perhitungan wilayah terluar dari benturan atau tabrakan dengan memikirkan area di luar dari kecepatan hambatan yaitu dua kali jari-jari agen. Merupakan salah satu sifat *velocity obstacle translasi* yaitu  $V_A \in VO_B^A(V_B) \Leftrightarrow V_B \in VO_A^B(V_A)$



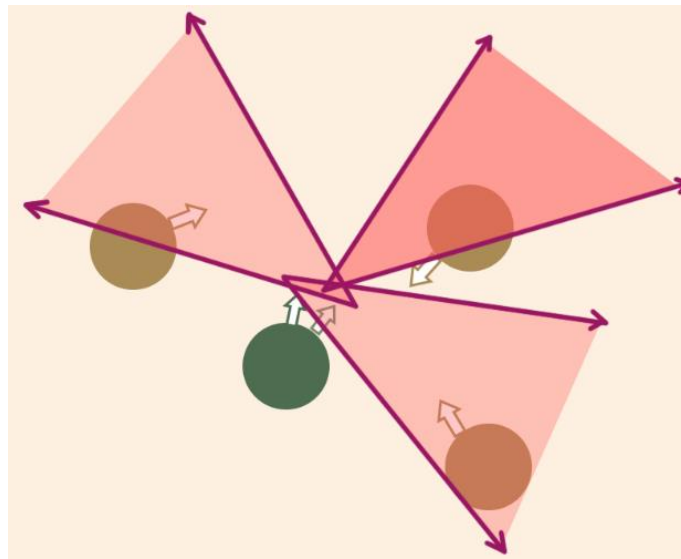
Gambar 3. 7 Velocity obstacle agen hijau pada agen kuning

Sudut kerucut yang berpusat pada posisi agen hijau yang terlihat pada Gambar 3.7 merupakan wilayah *velocity obstacle* yang didapat setelah melakukan perhitungan himpunan berdasarkan seluruh *velocity* agen dan area yang memungkinkannya terjadi benturan. *Velocity obstacle* agen hijau berada di dalam *velocity obstacle* agen kuning sehingga didefinisikan dengan  $VO_B^A(V_B) = \{V_A \mid \lambda(P_A, V_A - V_B) \cap B \oplus -A \neq \emptyset\}$



Gambar 3. 8 RVO agen hijau pada agen kuning

Gambar 3.8 menunjukkan adanya perubahan pusat kerucut dengan menggunakan algoritma RVO yaitu  $RVO_B^A(V_B, V_A) = \{V'_A \mid 2V'_A - V_A \in VO_B^A(V_B)\}$ . Pusat kerucut berpindah memiliki fungsi untuk menghindari osilasi serta menjelaskan bahwa kedua agen akan bergerak saling menghindar



Gambar 3. 9 RVO agen hijau bersama agen-agen kuning

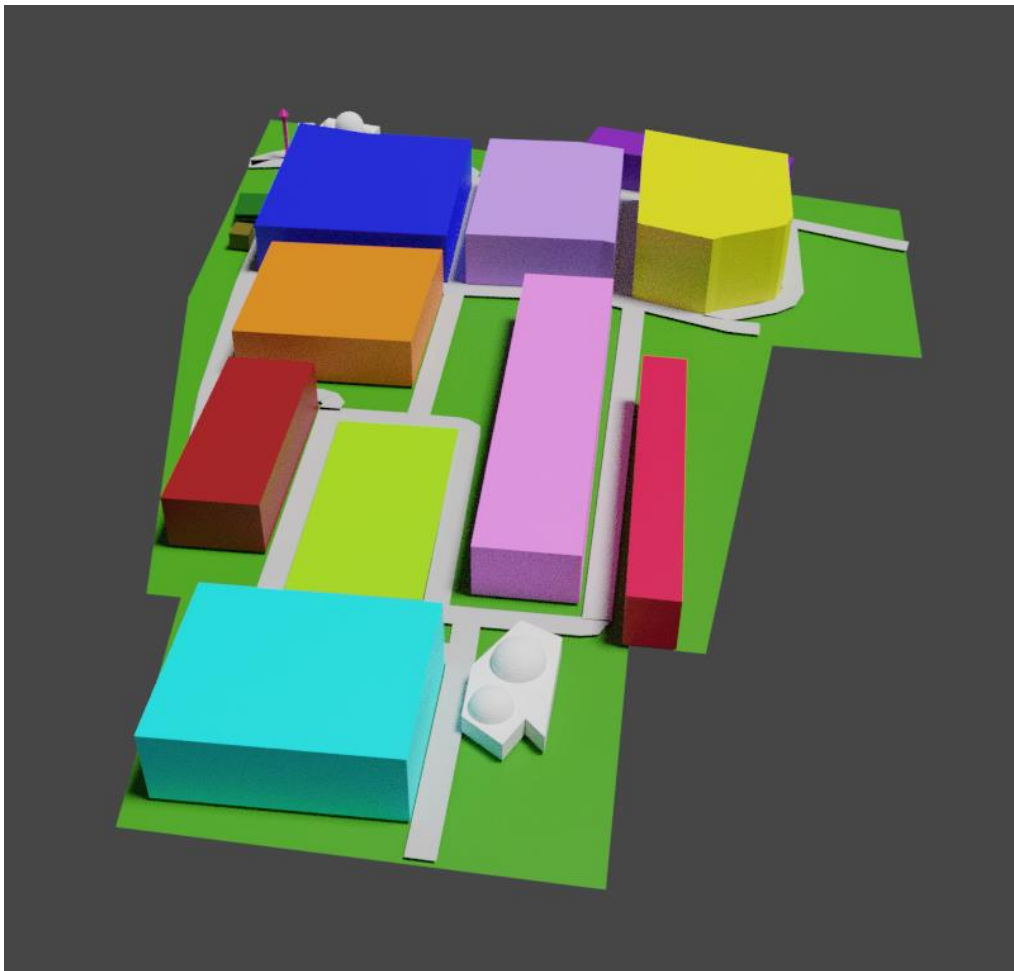
Tahap tersebut diulang untuk tiap agen dan menyatukan semua kerucut yang dihasilkan dari *velocity obstacle*. Daerah batas wilayah yang digambarkan secara efisien memprediksi *velocity* agen yang diharapkan menuju titik terdekat dengan tujuannya adalah *velocity* aman yang dapat digunakan agen hijau untuk mewujudkan serangkaian kecepatan bebas benturan.

### 3.4 Pembuatan Lingkungan Simulasi

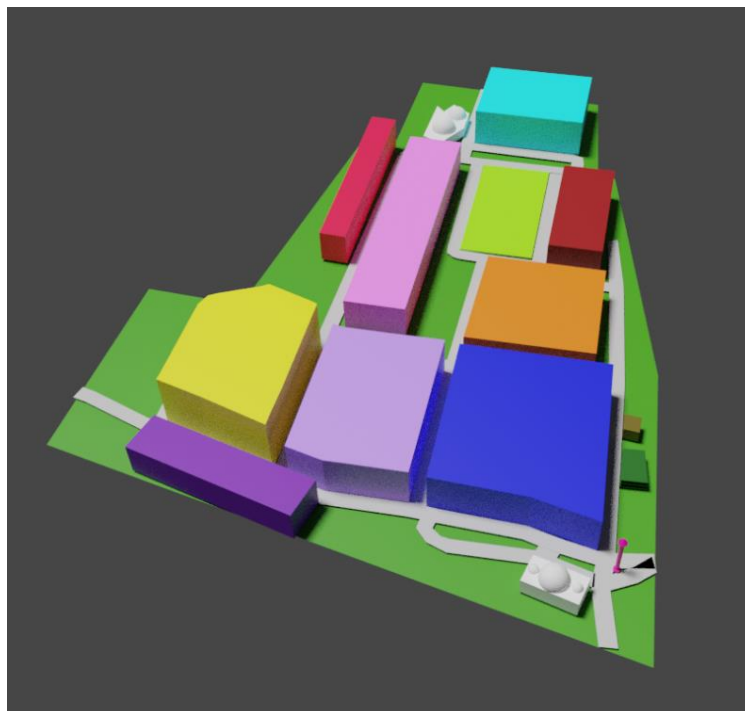
Objek yang berada di simulasi dibagi dua jenis berlandaskan kesanggupan agen untuk melakukan navigasi yakni *walkable* dan *unwalkable*. Pembuatan simulasi kerumunan atau *crowd simulation* menyesuaikan dengan skenario simulasi. Oleh karena itu, pentingnya mendefinisikan lingkungan simulasi yang mampu dilewati oleh agen dan yang tidak mampu dilewati oleh agen pada setiap skenario.

#### 1. Skenario Kampus UIN Malang

Pembuatan lingkungan simulasi pada tahap universitas skenario ini berada pada kawasan UIN Malang. Pada skenario kampus UIN Malang mempunyai empat titik awal yaitu gerbang masuk yang berada di beberapa titik merupakan tempat *spawn* agen dan titik akhir untuk skenario kampus UIN Malang adalah gedung-gedung yang tersebar dalam wilayah UIN Malang. Pada skenario kampus, titik akhir dari setiap kerumunan dibuat acak dengan membatasi agen sebanyak 200 agen dalam waktu 200 detik.



Gambar 3. 10 Wilayah skenario kampus UIN Malang tampak dari selatan



Gambar 3. 11 Wilayah skenario kampus UIN Malang tampak dari utara

Dalam Gambar 3.10 serta Gambar 3.11 menampilkan denah UIN Malang secara sederhana yang akan dilakukan pengujian. Warna ungu adalah gedung fakultas humaniora, warna kuning adalah gedung pusat UIN Malang, warna ungu muda adalah gedung perpustakaan, warna biru adalah gedung fakultas sains dan teknologi, putih adalah masjid, pink tua adalah menara UIN Malang, coklat adalah ruangan radio, oranye adalah gedung fakultas psikologi, *pink* adalah gedung B, merah muda adalah gedung A, merah tua adalah gedung C, hijau muda adalah lapangan, dan biru muda adalah gedung *sport center*. Pada universitas skenario terdapat beberapa titik awal dari setiap keramaian yang berada pada gerbang utama, gerbang belakang, gerbang asrama putra, gerbang asrama putri dan gerbang keluar. Gerbang adalah area masuk dari tiap agen untuk menuju wilayah simulasi yang dapat dilihat pada tabel berikut.

Tabel 3.1 Keterangan titik awal agen

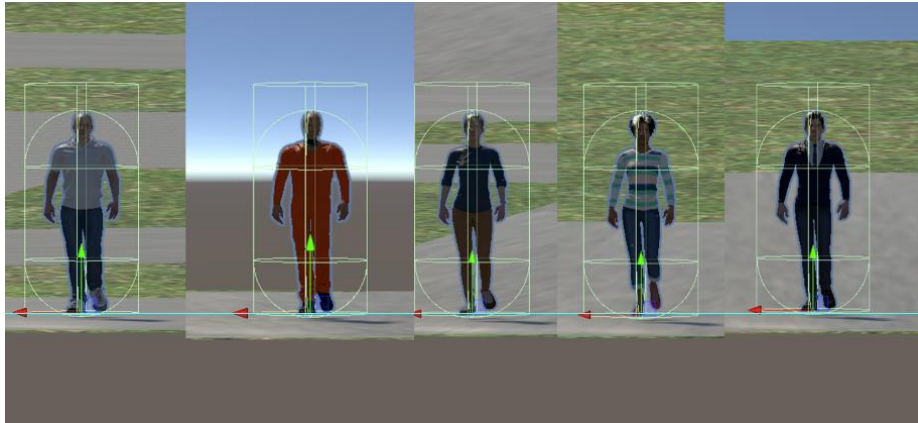
Keterangan Titik Awal	Koordinat
Gerbang Utama	(10,5 ; 0,5 ; -5)
Gerbang Mahad Pria	(124,5 ; 0,5 ; -48,13)
Gerbang Belakang	(137 ; 0,5 ; -26,3)
Gerbang Mahad Wanita	(73,3 ; 0,5 ; 118,9)
Gerbang Samping	(28,3 ; 0,5 ; 27,9)

Tabel 3.2 Keterangan titik tujuan agen

Keterangan Tujuan	Koordinat
<i>Sport Center</i>	(79 ; 0 ; 114,2)
Gedung Rektorat	(32,5 ; 0 ; -8,5)
Perpustakaan	(59 ; 4 ; 1,8)
Gedung B	(76,8 ; 0,01 ; 30,15)
Gedung C	(104,8 ; 0 ; 94,45)
Fakultas Psikologi	(128,3 ; 0 ; 0,6)
Fakultas Saintek	(130 ; 0 ; -47,7)
Fakultas Humaniora	(86,3 ; 0 ; -25,6)
Masjid Tarbiyah	(95 ; 0 ; -44,3)

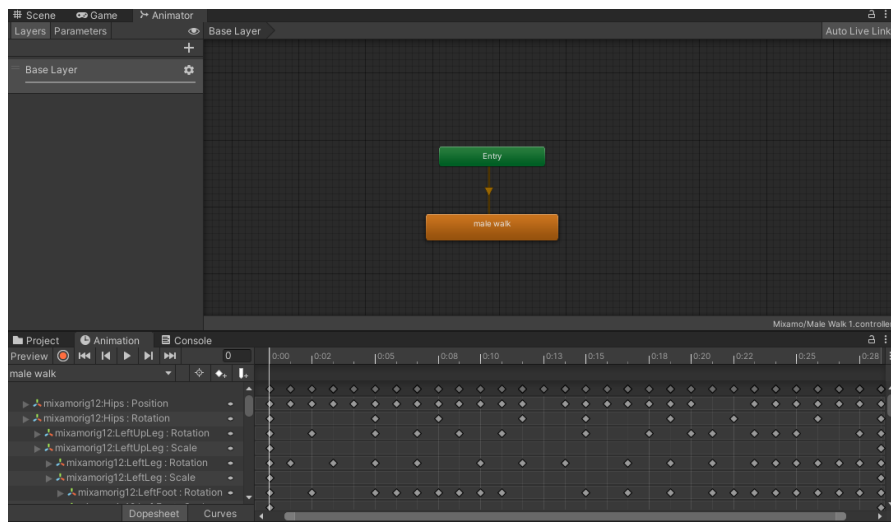
### 3.4.1 Deskripsi Agen

Agan pada simulasi kerumuman digambarkan sebagai 5 bentuk orang berbeda yang bergerak pada bidang datar. Desain agen yang diterapkan dapat dilihat pada Gambar 3.12 dengan ukuran dan diameter yang sama tiap agen. Setiap agen dideklarasikan sebagai *navmesh agent* dalam *Unity* yang berfungsi untuk menavigasikan agent pada bidang yang sudah dideklarasikan sebagai *navigation mesh* dalam filter *walkable*. *Navmesh agent* akan bergerak pada bidang *navigation mesh* yang *walkable*. Salah satu property RVO yaitu *navmesh agent* untuk mendeskripsikan *valid velocity* berlandaskan dasar lebar *navmesh agent*.



Gambar 3. 12 Bentuk agen

FSM atau *Finite State Machine* adalah sebuah metode perancangan sistem yang bisa menggambarkan sebuah tingkah laku. FSM diberikan kepada tiap-tiap agen yang berada saat simulasi berlangsung. Setiap agen memiliki desain FSM yang sama yaitu diam dan berjalan dapat dilihat pada Gambar 3.13



Gambar 3. 13 FSM pada tiap agen

Pada Gambar 3.13 dapat terlihat dua bagian yaitu bagian atas merupakan desain FSM NPC yang terdiri dari dua *state* yaitu *entry* dan *male walk* serta bagian bawah merupakan *keyframes* pergerakan tubuh agen. Ketika NPC mulai

menjalankan simulasi, maka setiap NPC akan mulai dari *state entry* seperti pada Gambar 3.13 yang berwarna hijau kemudian meneruskannya ke *state* yang sudah dipersiapkan yaitu *state male walk* berwarna kuning. Pada *state male walk* akan menjalankan animasi yang sudah diatur melalui *keyframes* di bagian bawah yang terhubung dengan *rigging* NPC sehingga menghasilkan gerakan jalan

### 3.4.2 Deskripsi Kampus

#### 1. Walkable area

*Walkable area* adalah area yang memungkinkan untuk dilalui atau ditempuh oleh agen yang sudah terdapat komponen *navmesh agent*. *Walkable area* pada Skenario Kampus UIN Malang dapat dilihat pada gambar 3.14 yang berwarna biru.

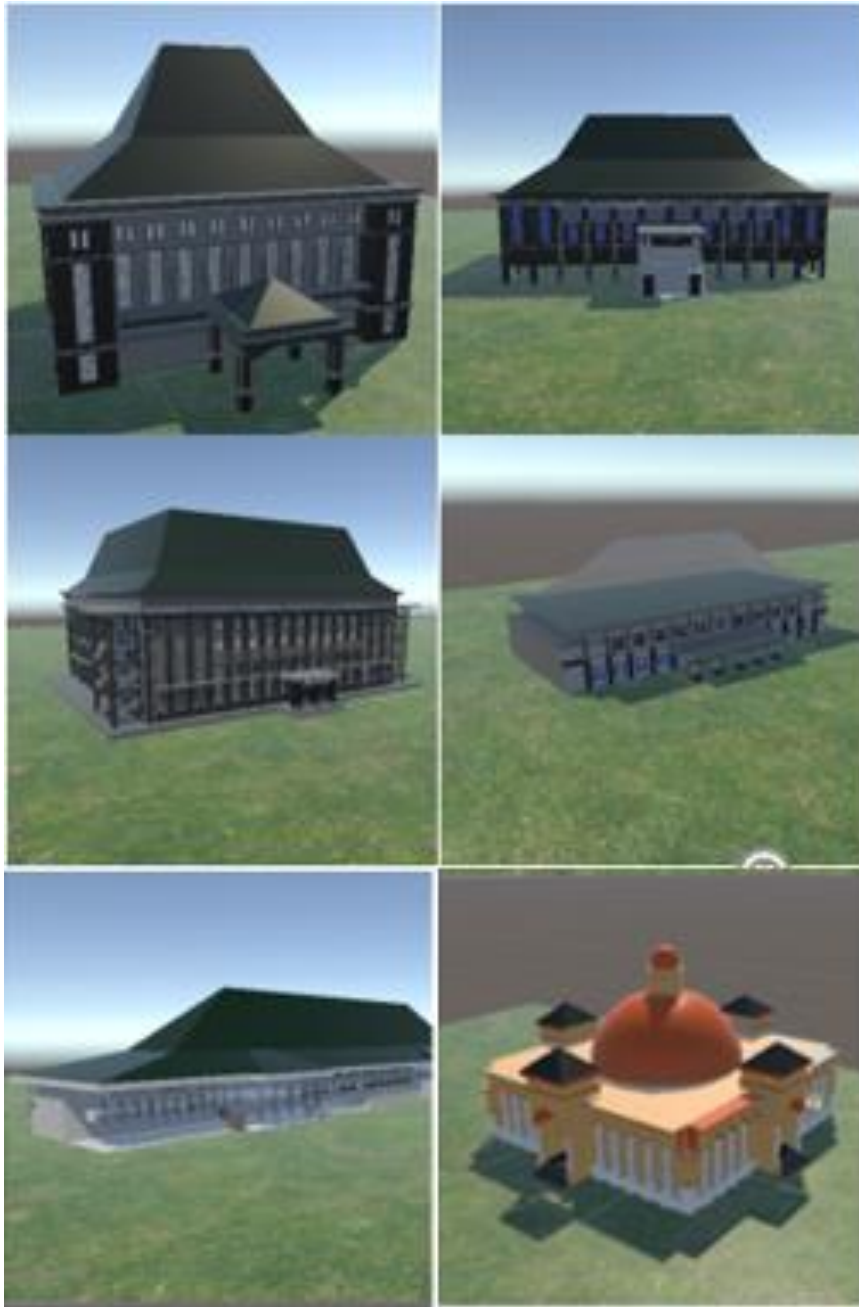


Gambar 3. 14 Walkable area skenario kampus UIN Malang



## 2. *Destination*

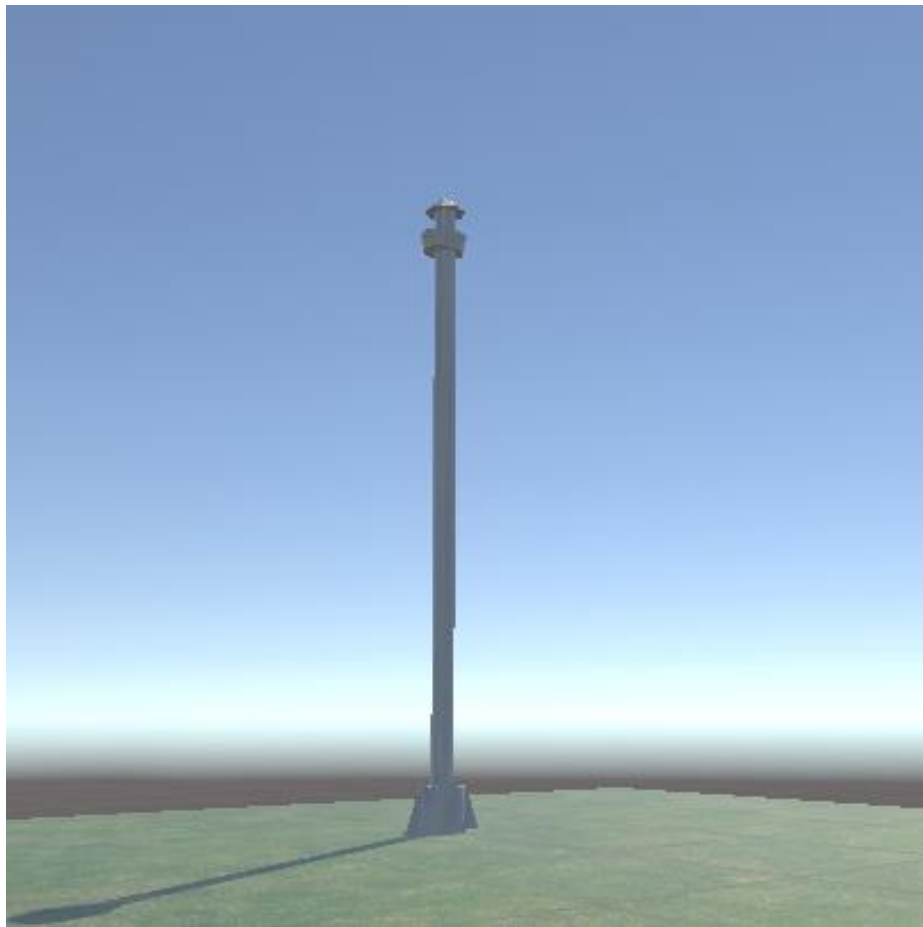
*Destination* adalah tujuan dari setiap agent. Pada skenario kampus UIN Malang terdapat 9 *destination* yang berupa bangunan pada wilayah kampus UIN Malang.



Gambar 3. 15 Destination skenario kampus UIN Malang

### 3. *Obstacle*

*Obstacle* adalah objek sebagai hambatan yang ada saat simulasi berlangsung. Setiap agen perlu menghindari *obstacle* supaya mencapai *destination*. Pada skenario kampus UIN Malang terdapat *obstacle* statis dan dinamis. *Obstacle* statis adalah objek penghalang yang tidak bergerak atau diam sedangkan *obstacle* dinamis adalah objek penghalang yang bergerak saat simulasi dijalankan. *Obstacle* statis pada skenario kampus UIN Malang berupa bangunan yaitu Menara UIN Malang dan *obstacle* dinamisnya adalah agen keramaian yang berlokasi pada simulasi.



Gambar 3. 16 *Obstacle* skenario kampus UIN Malang

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Tahapan Penelitian**

Desain sistem serta penerapan yang dijelaskan di bab sebelumnya, akan dijelaskan hasil pengujiannya pada bagian ini. Adapun pengujian dilakukan dengan tujuan membuktikan *crowd simulation* mampu terlaksana dengan baik menggunakan RVO dan terlaksana secara ideal.

##### **4.1.1 Perangkat Keras**

Pada proses pengujian ini, perangkat keras yang penulis gunakan adalah sebagai berikut

- Laptop MSI Katana G66 11UD
- Processor Intel I5 Gen 11-1140H
- NVIDIA GeForce RTX 3050 Ti dan RAM 8GB.

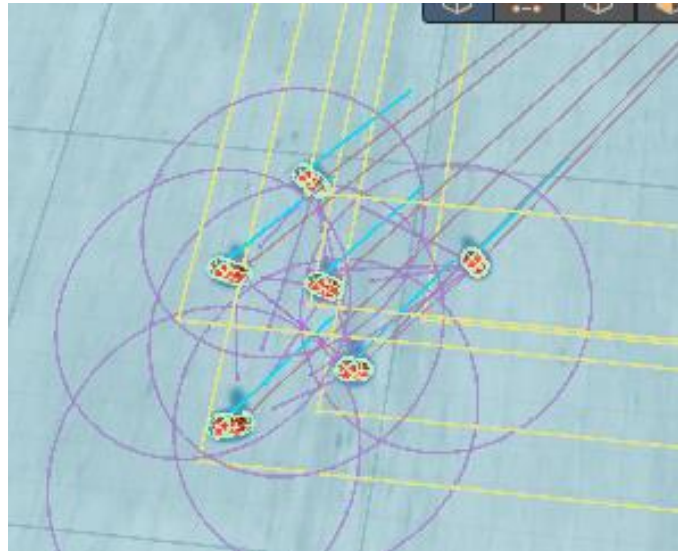
##### **4.1.2 Perangkat Lunak**

Adapun perangkat lunak yang digunakan untuk melaksanakan simulasi adalah seperti berikut :

- *Operating System* : Windows 11 Home 64-bit
- *Tools* : Blender, Unity3D
- *Desain* : Adobe Photoshop CC 2019
- *Text Editor* : Visual Studio Code

## 4.2 Pengujian RVO

### 4.2.1 Seeking

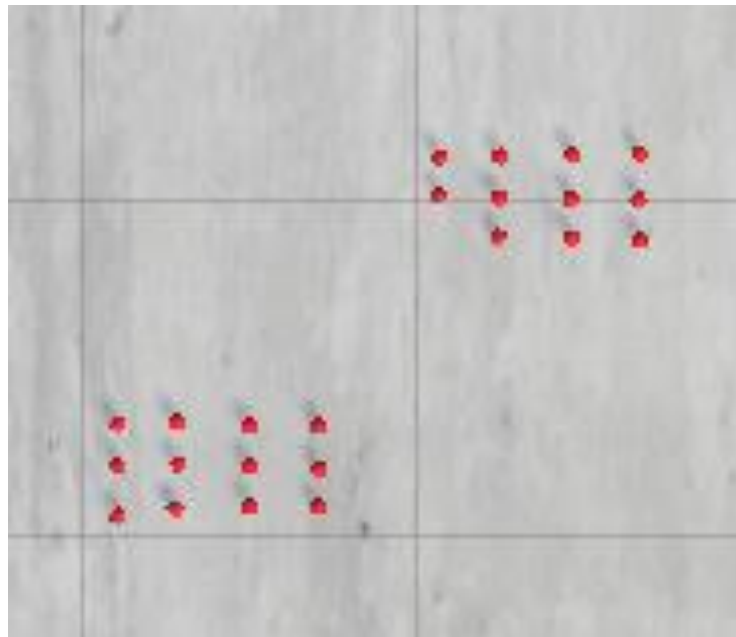


Gambar 4.1 *Seeking*

Simulasi yang menentukan pergerakan agen pertama kali adalah proses eksplorasi tujuan atau *seeking*. Dalam tahapan ini akan nampak kemampuan setiap agen untuk mencari titik tujuan mereka. Gambar 4.1 memperlihatkan produk dari mekanisme *seeking* yang dilakukan oleh tiap-tiap agen. Proses ini akan tersimpan berupa *velocity preference* yang memiliki informasi mengenai kecepatan dan proses dari awal sampai ke titik tujuan.

### 4.2.2 Steering

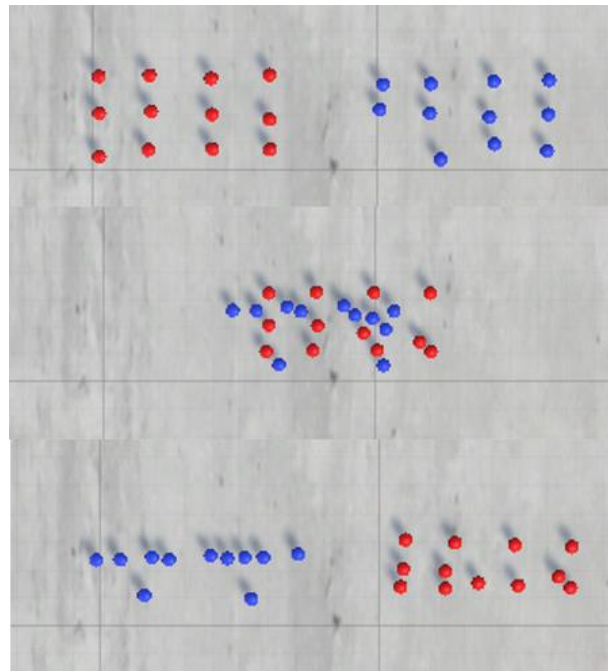
Proses selanjutnya yang akan dilakukan setelah menentukan tujuan dan mendapatkan informasi *velocity* adalah proses pengarahan atau disebut *steering*. Pada gambar 4.2 halaman selanjutnya memperlihatkan proses pengarahan agen menuju titik tujuan.



Gambar 4.2 *Steering*

### 4.2.3 Collision Avoidance

Tahap ketiga adalah tahap pengujian penghindaran tabrakan atau disebut *collision avoidance*. Algoritma RVO digunakan oleh agen untuk menguji keberhasilan *velocity referensi*. Pengujian tersebut dilakukan dengan cara memberikan gerakan kepada kerumunan agen dan memiliki tujuan berbeda atau berlawanan. Hal tersebut akan menyebabkan peristiwa benturan apabila masing-masing agen tetap berpindah memakai *velocity* referensinya. Gambar 4.3 menunjukkan bahwa mekanisme pengarahannya penghindaran tabrakan menggunakan RVO akan mengubah kecepatan agen diluar kecepatan hambatan yang memungkinkan terjadinya benturan.

Gambar 4.3 *Collision Avoidance*

#### 4.2.4 Ekstraksi Data

Pengujian selanjutnya adalah menguji keseluruhan manfaat dalam ekstraksi keterangan serta memastikan fungsi dapat digunakan dengan baik. Tabel 4.1 menunjukkan hasil dari pengujian fungsi.

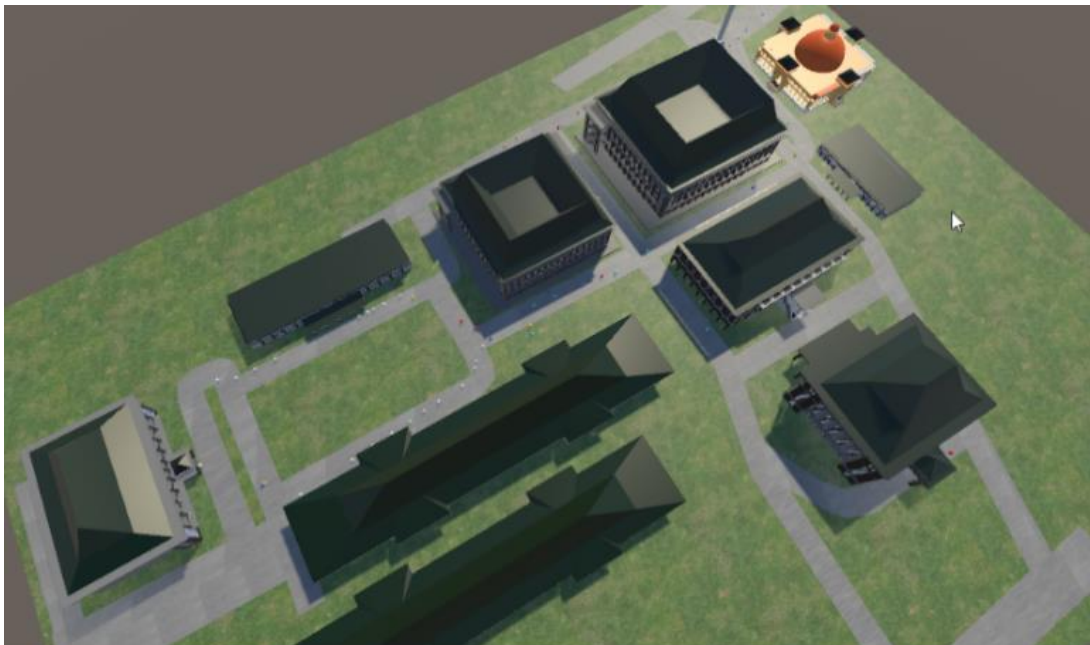
Tabel 4.1 Pengujian data dan fungsi

No	Keterangan	Hasil fungsi berjalan
1.	Jumlah tabrakan	$\sqrt{(Ya)}$
2.	Jumlah agen	$\sqrt{(Ya)}$
3.	Rata-rata <i>framerate</i>	$\sqrt{(Ya)}$

#### 4.3 Pengujian Skenario Kampus UIN Malik Ibrahim Malang

Pada pengujian ini, terdapat 5 titik awal diantaranya Gerbang Masuk Utama, Gerbang Ma'had Pria, Gerbang Belakang, Gerbang Samping dan Gerbang Ma'had Wanita. Sementara itu, tujuan akhir yang dimiliki adalah 9 titik tujuan diantaranya

Masjid Tarbiyah, Fakultas Humaniora, Fakultas Saintek, Rektorat, Perpustakaan, Fakultas Psikologi, Gedung B, Gedung C dan Gedung *Sport Center*. Sehingga didapatkan 45 skenario *velocity* dari agen. Setiap pengujian dilakukan dengan memberikan konstanta pada skenario, yaitu agen yang masuk tiap detiknya. Simulasi dilakukan dengan menambahkan 1 agen tiap detiknya baik yang memiliki tujuan sama ataupun tidak. Adapun *collision* yang terjadi akan dijelaskan lebih lanjut pada masing masing sub bab.



Gambar 4.4 Preview Skenario Kampus UIN Malang

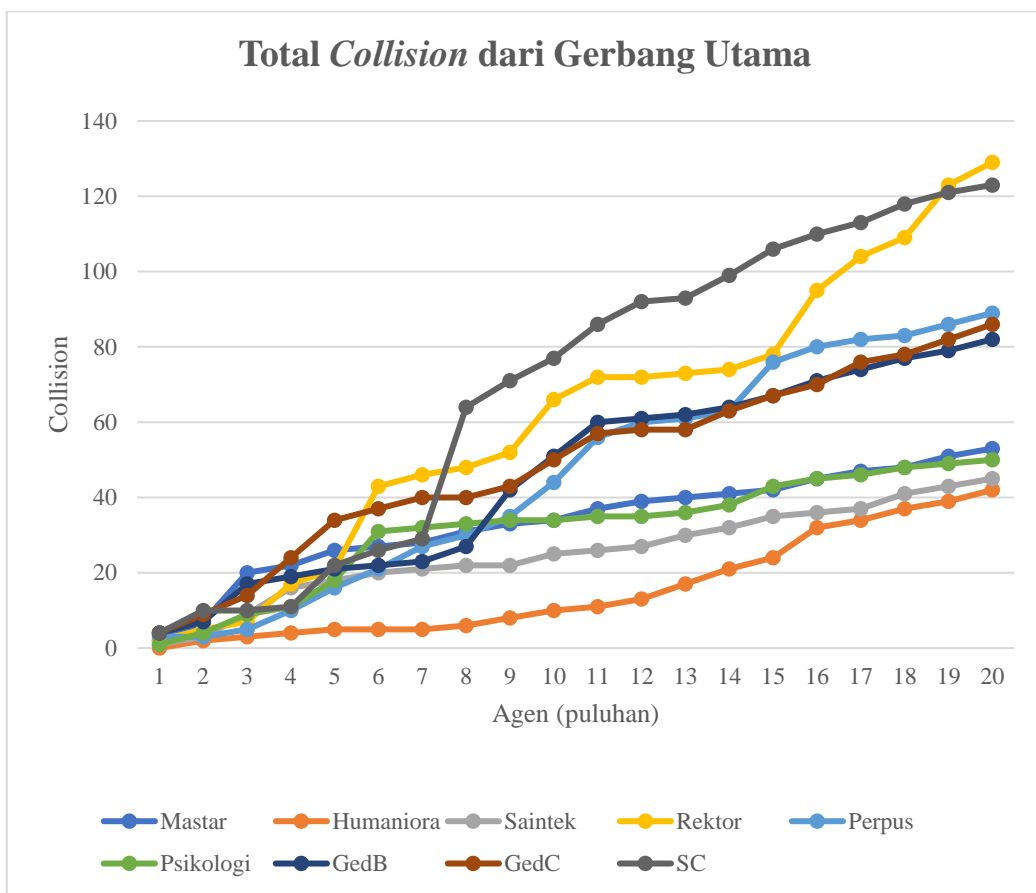
#### 4.3.1 Arah dari Gerbang Utama

Pada skenario kampus UIN Malang titik awal dimulai dari Gerbang Utama menuju 9 titik yang telah ditentukan. Titik-titik yang dimaksud dapat dilihat pada Gambar 4.6



Gambar 4.5 Titik awal dan tujuan dari Gerbang Utama

Hasil pengujian total *Collision* dapat dilihat pada Gambar 4.7



Gambar 4.6 Total Collision dari Gerbang Utama



Berdasarkan Gambar 4.7, total tabrakan yang dihasilkan untuk skenario ini cenderung meningkat seiring bertambahnya agen. Total tertinggi dari *collision* atau tabrakan terdapat pada rute tujuan akhir Gedung Rektor dengan angka 129 tabrakan dari 200 agen. Sementara terendah ada pada tujuan Gedung Humaniora dengan angka 42 tabrakan dari total agen yang sama yaitu 200.

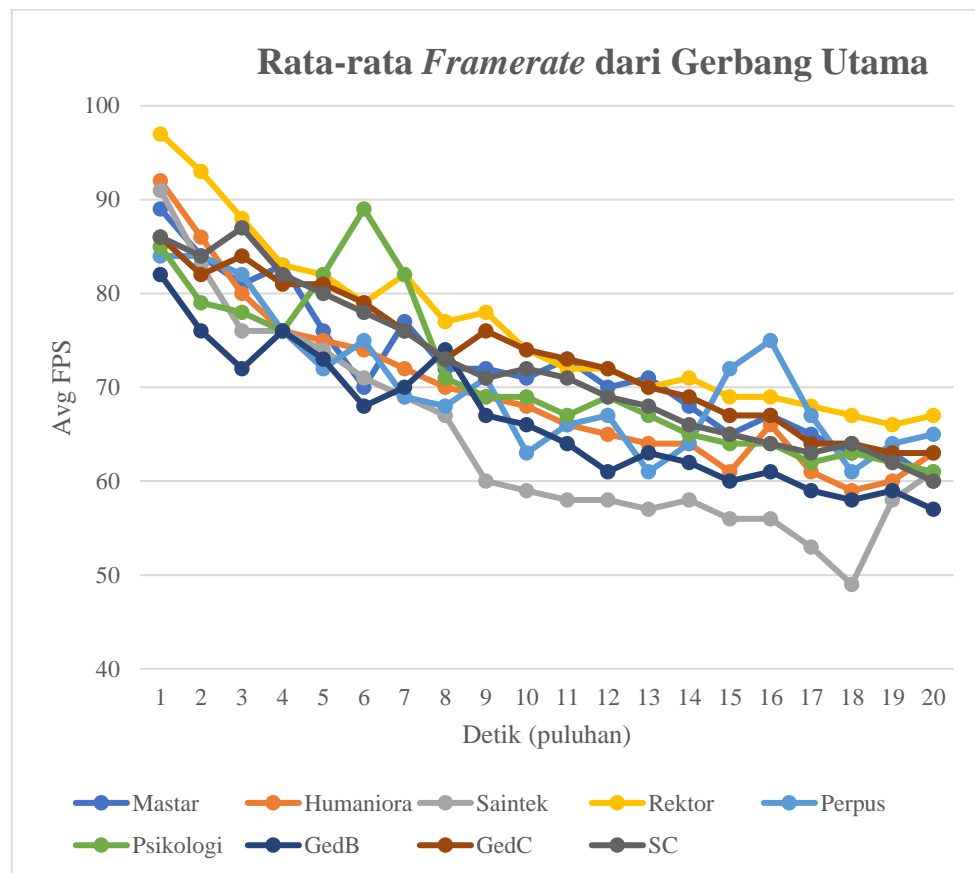
Selanjutnya adalah pengujian terhadap jumlah agen yang dapat sampai tujuan dalam range waktu tertentu yaitu 200 detik. Hasil pengujian jumlah agen dapat dilihat pada Tabel 4.2 berikut.

Tabel 4.2 Jumlah Agen dari Gerbang Utama

Mastar	Humani	Saintek	Rektor	Perpus	Psikologi	GedB	GedC	SC
191	181	179	195	181	167	165	157	144

Tabel berwarna hijau merupakan indikasi jumlah agen terbanyak yang sampai ke titik tujuan, sedangkan tabel berwarna kuning merupakan indikasi jumlah agen tersedikit yang sampai ke titik tujuan. Berdasarkan Tabel 4.2 dengan pengujian penghitungan agen yang sampai dalam waktu 200 detik, Gedung SC memiliki jumlah agen paling sedikit. Hal tersebut disebabkan oleh jarak antara titik awal dan tujuan yang lebih jauh daripada rute lainnya. Sementara Gedung Rektor memiliki jumlah agen paling banyak karena memiliki lokasi terdekat dari Gerbang Utama.

Selain total tabrakan dan jumlah agen yang tiba, pengujian juga dilakukan dalam data *framerate*. Gambar 4.8 menunjukkan rata-rata *framerate* saat simulasi berjalan.

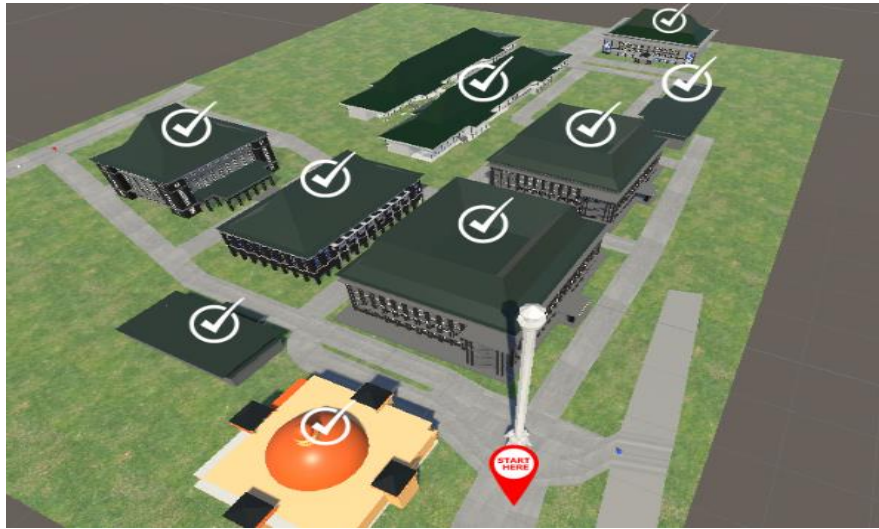


Gambar 4.7 Rata-rata *Framerate* dari Gerbang Utama

Berdasarkan Gambar 4.8, rata-rata *framerate* dari 9 titik tujuan yang telah ditentukan dapat dikatakan menurun seiring bertambahnya agen. Namun penurunan tersebut tidak ditunjukkan secara signifikan melainkan dengan kurva naik turun. Rata-rata *framerate* tertinggi adalah titik tujuan Gedung Rektor mencapai 97 FPS dan terendah adalah Gedung B dengan angka 49 FPS.

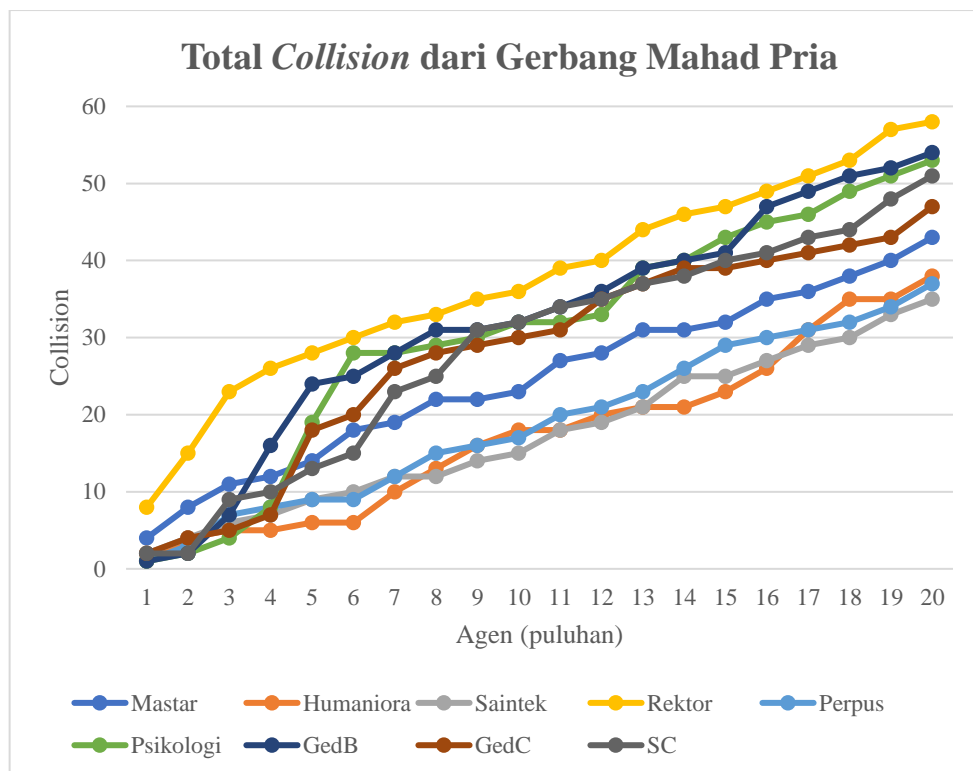
#### 4.3.2 Arah dari Gerbang Mahad Pria

Pada skenario ini titik awal dimulai dari Gerbang Mahad Pria menuju 9 titik yang telah ditentukan. Titik-titik yang dimaksud dapat dilihat pada Gambar 4.9



Gambar 4.8 Titik awal dan tujuan dari Gerbang Mahad Pria

Hasil pengujian total *Collision* pada dapat dilihat pada Gambar 4.9



Gambar 4.9 Total *Collision* dari Gerbang Mahad Pria

Berdasarkan Gambar 4.10, total tabrakan yang dihasilkan untuk skenario ini cenderung meningkat seiring bertambahnya agen. Total tertinggi dari *collision* atau tabrakan terdapat pada rute tujuan akhir Gedung Rektor dengan angka 58 tabrakan dari 200 agen. Sementara terendah ada pada tujuan Gedung Saintek dengan angka 35 tabrakan dari total agen yang sama yaitu 200.

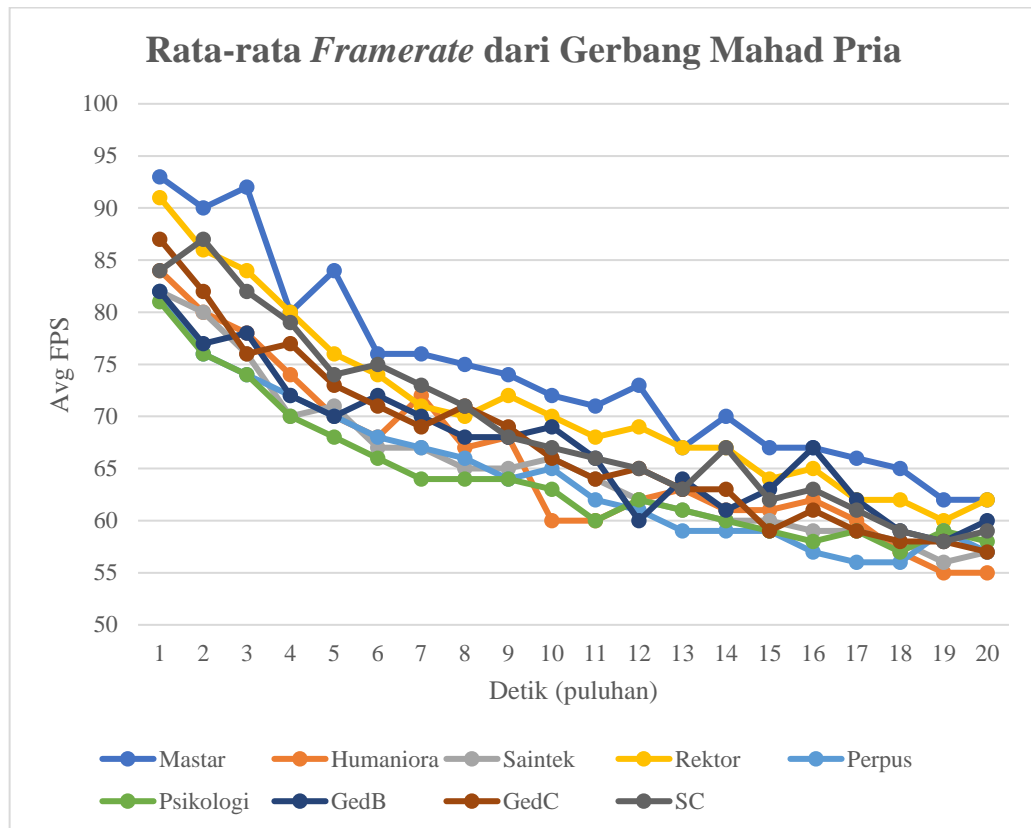
Selanjutnya adalah pengujian terhadap jumlah agen yang dapat sampai tujuan dalam range waktu tertentu yaitu 200 detik. Hasil pengujian jumlah agen dapat dilihat pada Tabel 4.3 berikut.

Tabel 4.3 Jumlah Agen dari Gerbang Mahad Pria

Mastar	Humani	Saintek	Rektor	Perpus	Psikologi	GedB	GedC	SC
201	201	193	187	222	201	162	184	187

Tabel berwarna hijau merupakan indikasi jumlah agen terbanyak yang sampai ke titik tujuan, sedangkan tabel berwarna kuning merupakan indikasi jumlah agen tersedikit yang sampai ke titik tujuan. Berdasarkan Tabel 4.3 dengan pengujian penghitungan agen yang sampai dalam waktu 200 detik, Gedung B memiliki jumlah agen paling sedikit. Hal tersebut disebabkan oleh jarak antara titik awal dan tujuan yang lebih jauh daripada rute lainnya. Sementara Gedung Mastar dan Humaniora memiliki jumlah agen sama-sama paling banyak karena memiliki lokasi terdekat dari Gerbang Mahad Pria.

Selain total tabrakan dan jumlah agen yang tiba, pengujian juga dilakukan dalam data *framerate*. Gambar 4.11 menunjukkan rata-rata *framerate* saat simulasi berjalan.

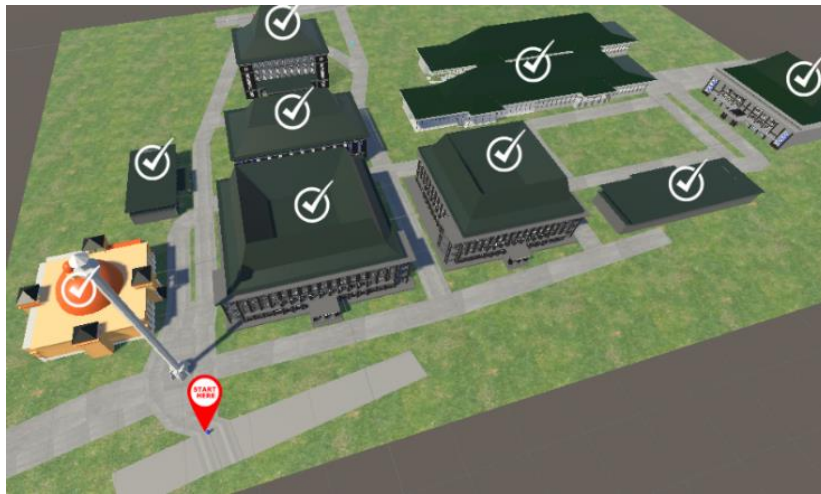


Gambar 4.10 Rata-rata Framerate dari Gerbang Mahad Pria

Berdasarkan Gambar 4.11, rata-rata *framerate* dari 9 titik tujuan yang telah ditentukan dapat dikatakan menurun seiring bertambahnya agen. Namun penurunan tersebut tidak ditunjukkan secara signifikan melainkan dengan kurva naik turun. Rata-rata frame rate tertinggi adalah titik tujuan Masjid Tarbiyah mencapai 92 FPS dan terendah adalah Gedung Humaniora dengan angka 55 FPS.

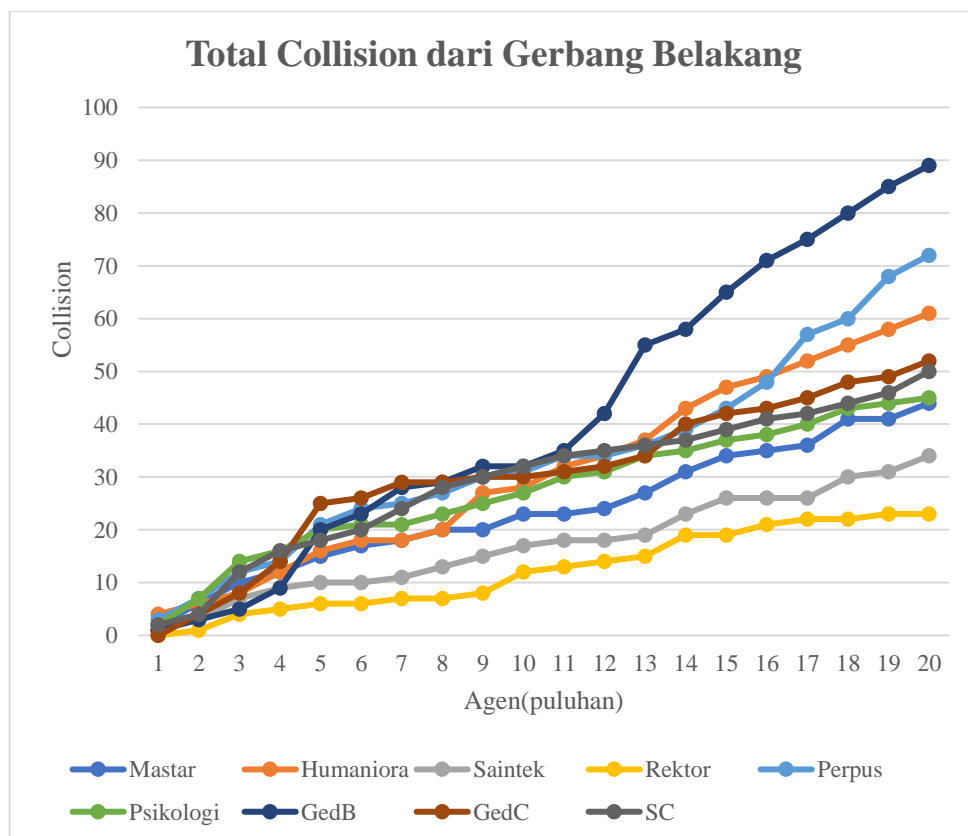
### 4.3.3 Arah dari Gerbang Belakang

Pada skenario ini titik awal dimulai dari Gerbang Belakang menuju 9 titik yang telah ditentukan. Titik-titik yang dimaksud dapat dilihat pada Gambar 4.12



Gambar 4.11 Titik awal dan tujuan dari Gerbang Belakang

Hasil pengujian total *Collision* dapat dilihat pada Gambar 4.12



Gambar 4.12 Total Collision dari Gerbang Belakang

Berdasarkan Gambar 4.13, total tabrakan yang dihasilkan untuk skenario ini cenderung meningkat seiring bertambahnya agen. Total tertinggi dari *collision* atau tabrakan terdapat pada rute tujuan akhir Gedung B dengan angka 89 tabrakan dari 200 agen. Sementara terendah ada pada tujuan Gedung Rektor dengan angka 23 tabrakan dari total agen yang sama yaitu 200.

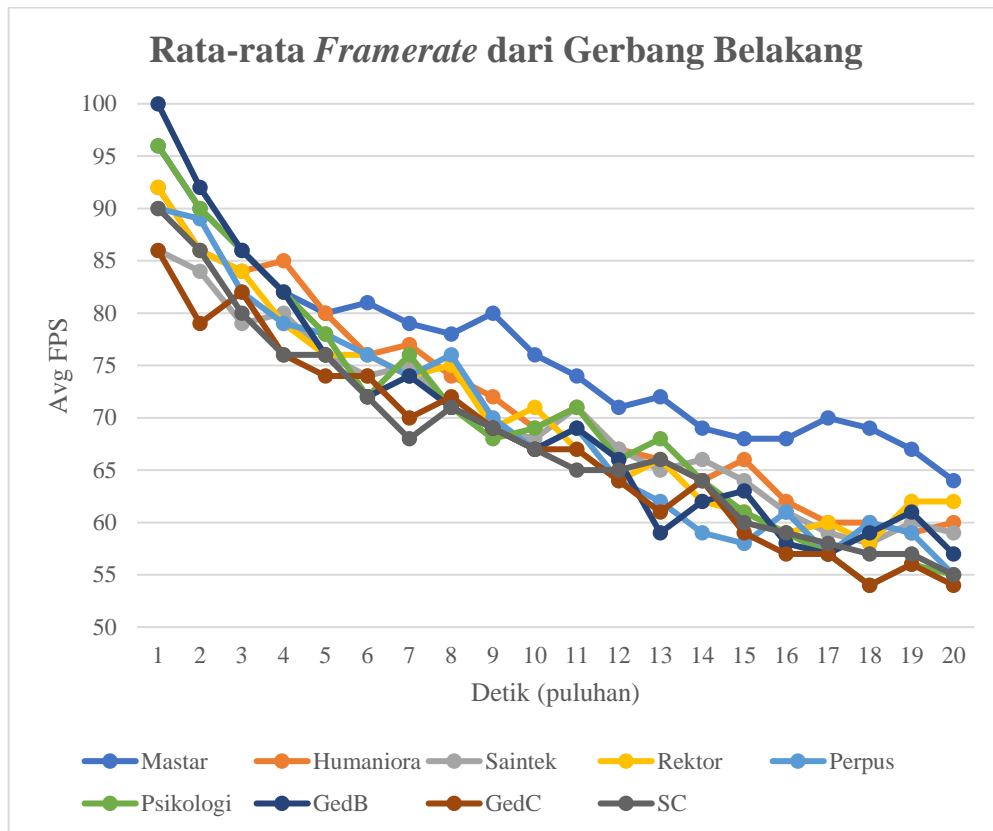
Selanjutnya adalah pengujian terhadap jumlah agen yang dapat sampai tujuan dalam range waktu tertentu yaitu 200 detik. Hasil pengujian jumlah agen dapat dilihat pada Tabel 4.4 berikut.

Tabel 4.4 Jumlah Agen dari Gerbang Belakang

Mastar	Humani	Saintek	Rektor	Perpus	Psikologi	GedB	GedC	SC
197	183	189	178	174	182	166	172	179

Tabel berwarna hijau merupakan indikasi jumlah agen terbanyak yang sampai ke titik tujuan, sedangkan tabel berwarna kuning merupakan indikasi jumlah agen tersedikit yang sampai ke titik tujuan. Berdasarkan Tabel 4.4 dengan pengujian penghitungan agen yang sampai dalam waktu 200 detik, Gedung B memiliki jumlah agen paling sedikit. Hal tersebut disebabkan oleh jarak antara titik awal dan tujuan yang lebih jauh daripada rute lainnya. Sementara Masjid Tarbiyah memiliki jumlah agen paling banyak karena memiliki lokasi terdekat dari Gerbang Utama.

Selain total tabrakan dan jumlah agen yang tiba, pengujian juga dilakukan dalam data *framerate*. Gambar 4.14 menunjukkan rata-rata *framerate* saat simulasi berjalan.



Gambar 4.13 Rata-rata *Framerate* dari Gerbang Belakang

Berdasarkan Gambar 4.14, rata-rata *framerate* dari 9 titik tujuan yang telah ditentukan dapat dikatakan menurun seiring bertambahnya agen. Namun penurunan tersebut tidak ditunjukkan secara signifikan melainkan dengan kurva naik turun. Rata-rata frame rate tertinggi adalah titik tujuan Gedung B mencapai 100 FPS dan terendah adalah Gedung C dengan angka 54.

#### 4.3.4 Arah dari Gerbang Samping

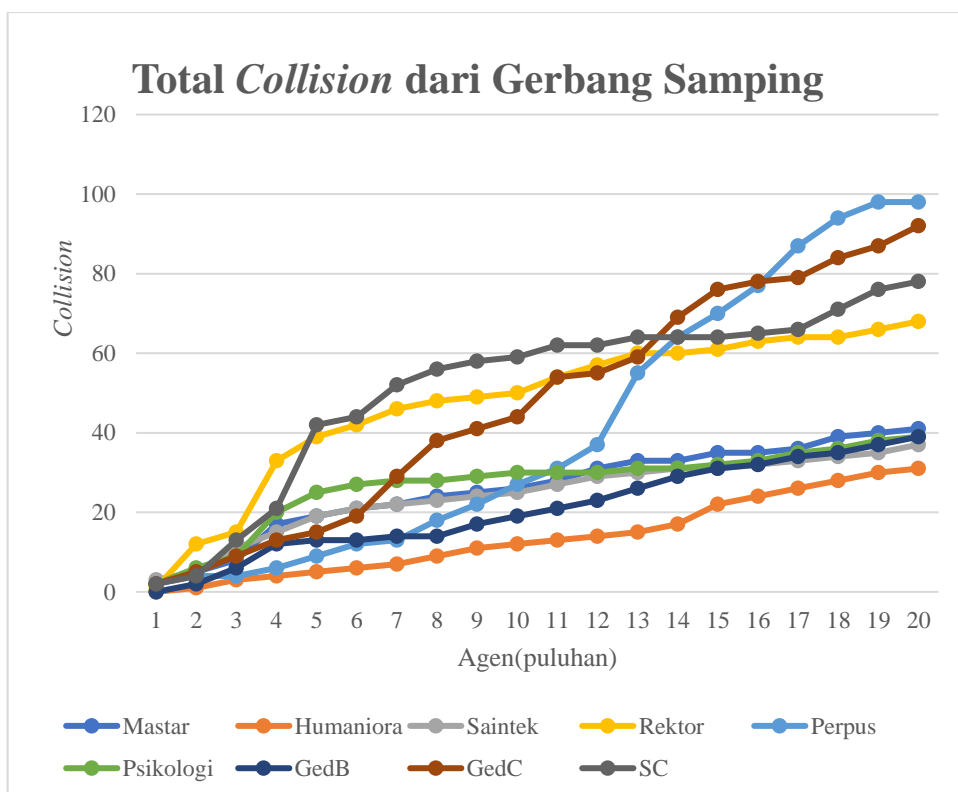
Pada skenario ini titik awal dimulai dari Gerbang Samping menuju 9 titik yang telah ditentukan. Titik-titik yang dimaksud dapat dilihat pada Gambar 4.15





Gambar 4.14 Titik awal dan tujuan dari Gerbang Samping

Hasil pengujian total *Collision* dapat dilihat pada Gambar 4.16.



Gambar 4.15 Total Collision dari Gerbang Samping

Berdasarkan Gambar 4.16, total tabrakan yang dihasilkan untuk skenario ini cenderung meningkat seiring bertambahnya agen. Total tertinggi dari *collision* atau tabrakan terdapat pada rute tujuan akhir Gedung Perpustakaan dengan angka 98 tabrakan dari 200 agen. Sementara terendah ada pada tujuan Gedung Humaniora dengan angka 31 tabrakan dari total agen yang sama yaitu 200.

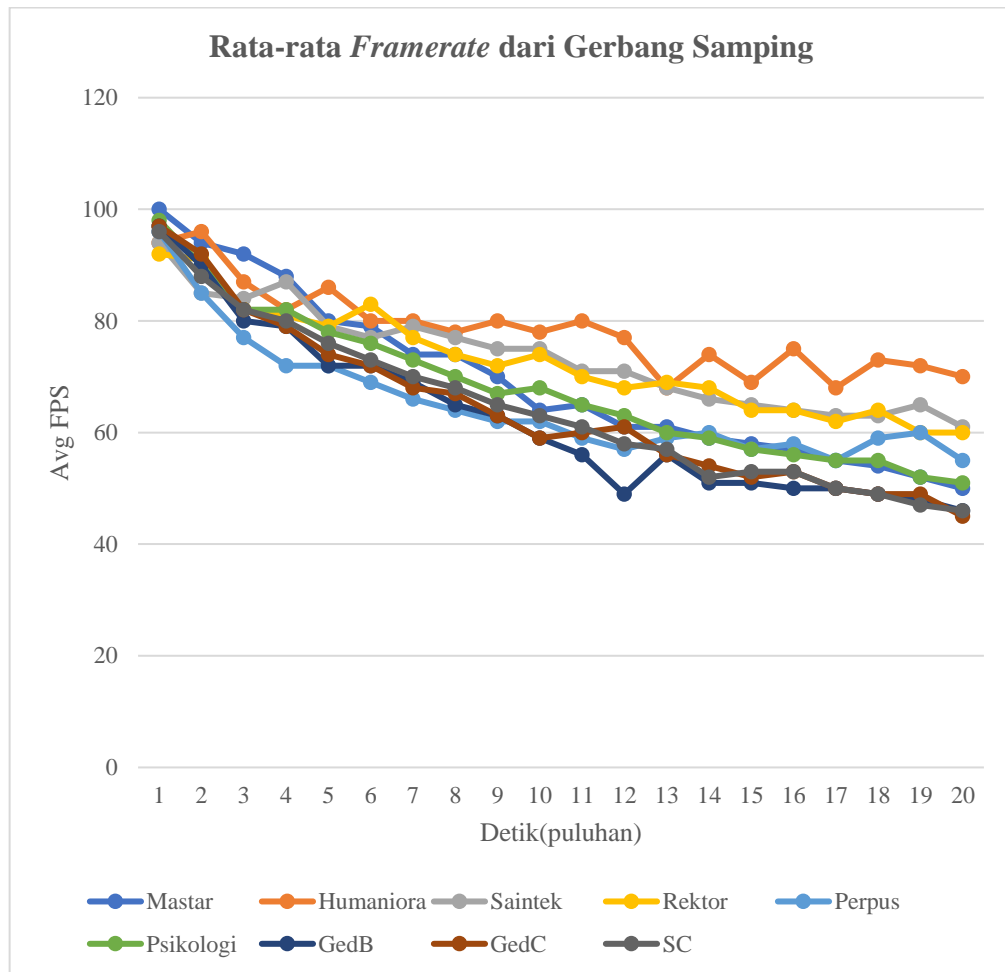
Selanjutnya adalah pengujian terhadap jumlah agen yang dapat sampai tujuan dalam range waktu tertentu yaitu 200 detik. Hasil pengujian jumlah agen dapat dilihat pada Tabel 4.5 berikut.

Tabel 4.5 Jumlah Agen dari Gerbang Samping

Mastar	Humani	Saintek	Rektor	Perpus	Psikologi	GedB	GedC	SC
192	190	202	200	252	205	200	182	208

Tabel berwarna hijau merupakan indikasi jumlah agen terbanyak yang sampai ke titik tujuan, sedangkan tabel berwarna kuning merupakan indikasi jumlah agen tersedikit yang sampai ke titik tujuan. Berdasarkan Tabel 4.5 dengan pengujian penghitungan agen yang sampai dalam waktu 200 detik, Gedung Perpustakaan memiliki jumlah agen paling banyak karena memiliki lokasi paling dekat. Sementara agen sampai di gedung yang lain rata-rata hampir di angka yang sama karena Gerbang samping berada di tengah tengah kawasan kampus UIN Malik Ibrahim Kota Malang. Jumlah agen terendah yang sampai titik tujuan pada rute ini adalah dengan tujuan Gedung C karena berada di ujung dan melewati titik kerumunan yang kompleks.

Selain total tabrakan dan waktu yang dibutuhkan agen, pengujian juga dilakukan dalam data *framerate*. Gambar 4.17 menunjukkan rata-rata *framerate* saat simulasi berjalan.



Gambar 4.16 Rata-rata *Framerate* dari Gerbang Samping

Berdasarkan Gambar 4.17, rata-rata *framerate* dari 9 titik tujuan yang telah ditentukan dapat dikatakan menurun seiring bertambahnya agen. Namun penurunan tersebut tidak ditunjukkan secara signifikan melainkan dengan kurva naik turun. Rata-rata frame rate tertinggi adalah titik tujuan Masjid Tarbiyah mencapai 100 FPS dan terendah adalah Gedung SC dengan angka 46 FPS.

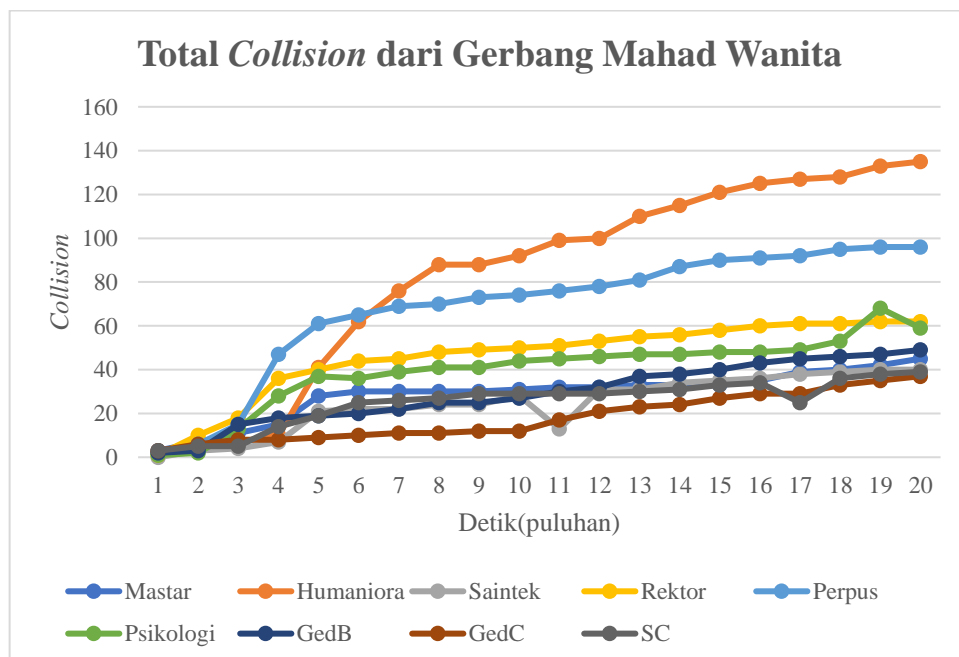
### 4.3.5 Arah dari Gerbang Mahad Wanita

Pada skenario ini titik awal dimulai dari Gerbang Mahad Wanita menuju 9 titik yang telah ditentukan. Titik-titik yang tersebut dapat dilihat pada Gambar 4.18



Gambar 4.17 Titik awal dan tujuan dari Gerbang Mahad Wanita

Hasil pengujian total *Collision* dapat dilihat pada Gambar 4.18



Gambar 4.18 Total Collision dari Gerbang Mahad Wanita

Berdasarkan Gambar 4.19, total tabrakan yang dihasilkan untuk skenario ini cenderung meningkat seiring bertambahnya agen. Total tertinggi dari *collision* atau tabrakan terdapat pada rute tujuan akhir Gedung Humaniora dengan angka 135 tabrakan dari 200 agen. Sementara terendah ada pada tujuan Gedung SC dengan angka 37 tabrakan dari total agen yang sama yaitu 200.

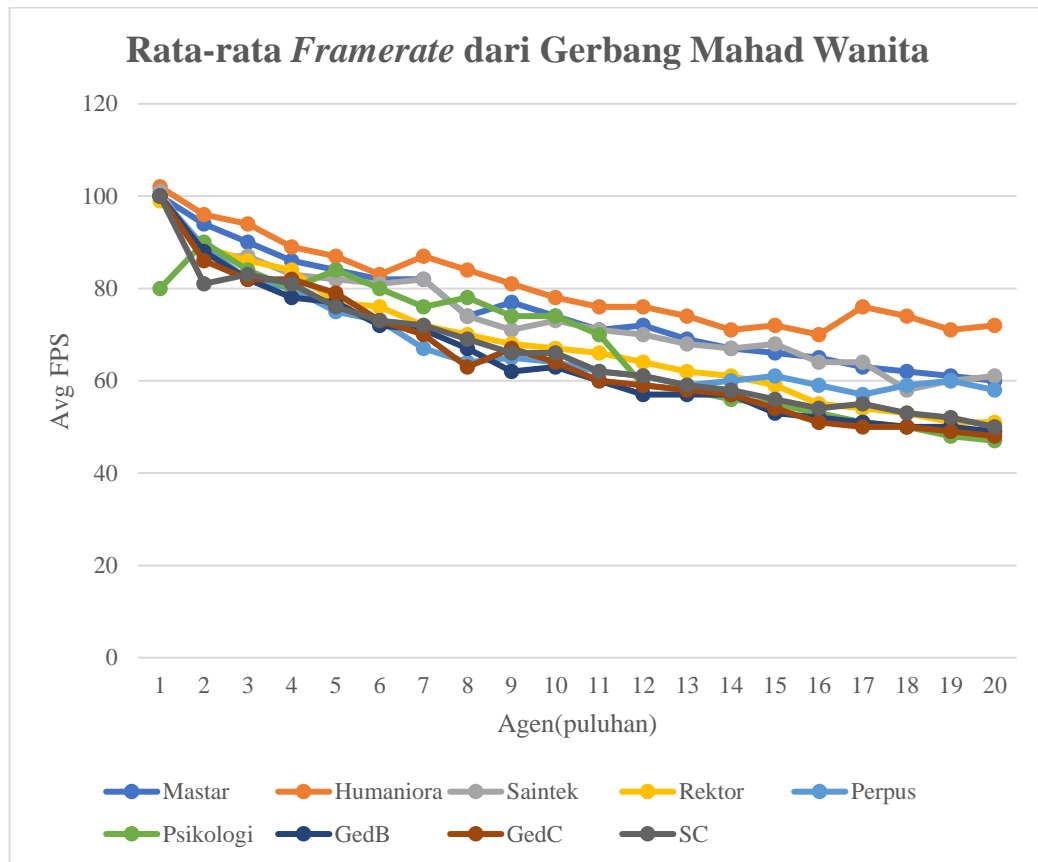
Selanjutnya adalah pengujian terhadap jumlah agen yang dapat sampai tujuan dalam range waktu tertentu. Hasil pengujian jumlah agen dapat dilihat pada Tabel 4.6 berikut.

Tabel 4.6 Jumlah Agen dari Gerbang Mahad Wanita

Mastar	Humani	Saintek	Rektor	Perpus	Psikologi	GedB	GedC	SC
188	161	174	186	194	200	190	190	203

Tabel berwarna hijau merupakan indikasi jumlah agen terbanyak yang sampai ke titik tujuan, sedangkan tabel berwarna kuning merupakan indikasi jumlah agen tersedikit yang sampai ke titik tujuan. Berdasarkan Tabel 4.6 dengan pengujian penghitungan agen yang sampai dalam waktu 200 detik, *Sport Center* (SC) memiliki jumlah agen paling banyak karena memiliki lokasi paling dekat. Gedung Humaniora memiliki jumlah agen paling sedikit karena memiliki jarak yang jauh.

Selain total tabrakan dan waktu yang dibutuhkan agen, pengujian juga dilakukan dalam data *framerate*. Gambar 4.19 menunjukkan rata-rata *framerate* saat simulasi berjalan.



Gambar 4.19 Rata-rata Framerate dari Gerbang Mahad Wanita

Rata-rata *framerate* dari 9 titik tujuan yang telah ditentukan dapat dikatakan menurun seiring bertambahnya agen. Namun penurunan tersebut tidak ditunjukkan secara signifikan melainkan dengan kurva naik turun. Rata-rata frame rate tertinggi adalah titik tujuan Gedung Humaniora mencapai 102 FPS dan terendah adalah Gedung C dengan angka 47 FPS.

#### 4.4 Integrasi Islam

Berdasarkan pengujian yang telah dilaksanakan, didapatkan hasil bahwa dari semua skenario yang disimulasikan memiliki hasil yang berbeda-beda. Hal

tersebut dapat terjadi dikarenakan perbedaan skenario yaitu titik awal, titik tujuan, jumlah agen, *obstacle* dan jarak tempuh yang dibutuhkan oleh agen tersebut.

Demikian pula dengan iman yang dimiliki setiap individu. Iman kita dapat berubah-ubah disebabkan oleh beberapa faktor diantaranya lingkungan sekitar, ketekunan ibadah, budaya, ilmu yang dimiliki dan lain-lain. Dengan adanya beberapa faktor, iman seseorang seringkali naik turun berbentuk kurva. Naik turunnya iman telah dijelaskan oleh Allah SWT pada surat Al-Anfal ayat 2 yang berbunyi:

إِنَّمَا الْمُؤْمِنُونَ الَّذِينَ إِذَا ذُكِرَ اللَّهُ وَجِلَّتْ قُلُوبُهُمْ وَإِذَا تُلِيَتْ عَلَيْهِمْ آيَاتُهُ زَادَتْهُمْ إِيمَانًا وَعَلَىٰ رَبِّهِمْ  
يَتَوَكَّلُونَ

*“Sesungguhnya orang-orang yang beriman ialah mereka yang bila disebut nama Allah gemetarlah hati mereka, dan apabila dibacakan ayat-ayat-Nya bertambahlah iman mereka (karenanya), dan hanya kepada Tuhanlah mereka bertawakkal”*. (QS. Al-Anfal:2)

Berdasarkan ayat tersebut, Al-Muyassar atau Kementerian Agama Saudi Arabi menafsirkan bahwa sesungguhnya orang-orang yang beriman adalah mereka yang ketika mendengar Nama Allah, hati mereka merasa takut, dan ketika dibacakan ayat-ayat suci Al-Qur’an bertambahlah keimanan mereka bersama keimanan yang sudah ada. Hal tersebut menerangkan bahwa iman seseorang pasti juga akan bertambah ketika semakin mendekatkan diri kepada Allah, dan sebaliknya ketika diri jauh dari ilmu agama dan perintahNya, maka akan berkurang pula iman dari pada diri seseorang.

Rasulullah SAW senantiasa menjelaskan tentang naik-turunnya iman dalam hadis sebagai berikut

عَنْ أَبِي هُرَيْرَةَ قَالَ قَالَ رَسُولُ اللَّهِ -صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ- «الإِيمَانُ بِضْعٌ وَسَبْعُونَ بَابًا فَأَدْنَاهَا إِيمَانُ  
«الأَدَى عَنِ الطَّرِيقِ وَأَرْفَعُهَا قَوْلُ لَا إِلَهَ إِلَّا اللَّهُ»

“Dari Abu Hurairah radhiyallahu ‘anhu, ia berkata bahwa Rasulullah shallallahu ‘alaihi wa sallam bersabda, “Iman itu ada tujuh puluh sekian pintu. Yang paling rendah dari iman adalah menyingkirkan gangguan dari jalanan. Yang paling tinggi adalah kalimat laa ilaha illallah.” (HR. Muslim, no. 35 dan Tirmidzi, no. 2614)

Berdasarkan hadis tersebut, Syaikh `Abdurahman as-Sa’ di *raimahullah* menjelaskan bahwa secara jelas hadis ini menunjukkan kepada kita iman dapat bertambah serta berkurang. Ia menyampaikan “Ini jelas sekali menunjukkan bahwa iman itu bertambah dan berkurang sesuai dengan bertambahnya aturan syariat dan cabang-cabang iman, serta amalan hamba tersebut atau tidak mengamalkannya. Sudah dimaklumi bersama bahwa manusia bertingkat-tingkat dalam hal ini. Siapa yang berpendapat bahwa iman itu tidak bertambah dan berkurang, maka telah menyelisihi realita yang nyata di samping menyelisihi syariat sebagaimana telah diketahui”

Dalam penelitian ini terdapat 2 konsep muamalah, muamalah yang dimaksud adalah *muamalah ma’a Allah* serta *muamalah ma’a an-nas*. *Muamalah ma’a Allah* adalah hubungan manusia sebagai seorang hamba kepada Allah sedangkan *muamallah ma’a an-nas* adalah hubungan antar manusia.

#### 4.4.1 *Muamalah ma’a Allah*

Kita harus memahami apabila iman seseorang sanggup bertambah dan juga mampu berkurang, dan hal ini sudah menjadi *ijma’* (kesepakatan) para ulama *Ahlussunnah Wal Jamaah* bahkan ulama terdahulu, termasuk para sahabat



*radhiallahu Anhu*. Dan *ijma'* (keepakatan) ini diungkapkan di antaranya dari kalangan para sahabat *radhiallahu Anhu* misal sahabat Abu Darda *radhiyallahu Anhu*. Abu Darda pernah mengatakan “*al imanu yazidu wa yanqush.*” iman itu bertambah dan berkurang, naik dan turun, menguat dan melemah. Iman bertambah kekuatannya dengan ketaatan-ketaatan yang kita lakukan seperti menjalankan perintah-perintah Allah dan amal sholeh. Iman berkurang kekuatannya dengan kemaksiatan yang dilakukan. Pasang surut iman seseorang tersebut dapat dipengaruhi dari godaan duniawi seperti lingkungan, pertemanan, budaya, ilmu dan lain-lainnya. Allah sudah memberikan petunjuk untuk menjaga iman kita melalui surat At-Taubah ayat 119 yang berbunyi:

يَا أَيُّهَا الَّذِينَ آمَنُوا اتَّقُوا اللَّهَ وَكُونُوا مَعَ الصَّادِقِينَ

“Wahai orang-orang yang beriman! Bertakwalah kepada Allah, dan bersamalah kamu dengan orang-orang yang benar.” (QS. At-Taubah: 119).

Berdasarkan ayat tersebut menjelaskan cara mempertahankan naik turunnya kekuatan iman kita agar tidak melampaui batas-batas yang membahayakan. Jika iman melemah atau turun yaitu bertakwalah kepada Allah, menjalankan perintah Allah dan menjauhi laranganNya. Kemudian dilanjutkan dengan bersamalah dengan orang-orang yang bisa membawa kita ke kebenaran, menjaga kita tetap berada dalam kebenaran dan bersama dengan orang-orang yang mampu membuat kita menjadi bersemangat dalam amal sholeh. Jadi dua kuncinya yaitu bertakwa kepada Allah dan carilah orang-orang yang benar dan sholeh.

Salah satu cara memperbarui iman juga ditunjukkan oleh Rasulullah yaitu dengan memperbanyak *tahlil*. Sesuai dalam hadis yang berbunyi:

عَنْ أَبِي هُرَيْرَةَ رَضِيَ اللَّهُ عَنْهُ قَالَ: قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ جَدِّدُوا إِيمَانَكُمْ قِيلَ  
يَارَسُولَ اللَّهِ وَكَيْفَ نُجَدِّدُوا إِيمَانَنَا قَالَ أَكْثَرُوا مِنْ قَوْلِ لَا إِلَهَ إِلَّا اللَّهُ

*Dari Abu Hurairah r.a. berkata bahwa Rasulullah shallallahu alaihi wa shallam bersabda, “Perbaharuilah iman kalian.” Para sahabat bertanya, “Bagaimana cara memperbaharui iman kami, ya Rasulullah?” Rasulullah saw bersabda, “Perbanyaklah ucapan ‘laa ilaaha illallaah’”. (HR Al-Bukhari)*

#### 4.4.2 *Muamalah ma’a an-nas*

Penelitian yang telah dilakukan memiliki tujuan yaitu untuk menolong sesama manusia. Hal tersebut antara lain membantu orang-orang yang memiliki hambatan untuk menuju atau merasakan suasana kampus UIN Malang secara langsung. Dengan cara meminimalisir hambatan tersebut, penggunaan *virtual reality* digunakan sebagai media untuk merasakan atmosfer kampus UIN Malang secara *virtual* dengan *visualisasi* suasana lingkungan yang mendekati nyata. Sehingga untuk terasa berada di lingkungan UIN Malang bisa dilakukan secara lebih efektif dan efisien. Allah berfirman pada surat Al-Maidah ayat 2 yang berbunyi:

... وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ وَاتَّقُوا اللَّهَ إِنَّ اللَّهَ شَدِيدُ الْعِقَابِ

*Dan tolong-menolonglah kamu dalam (mengerjakan) kebajikan dan takwa, dan jangan tolong-menolong dalam berbuat dosa dan permusuhan. Bertakwalah kepada Allah, sungguh, Allah sangat berat siksaan-Nya. (QS.Al-Maidah : 2)*

Pada ayat tersebut, ditegaskan bahwa sikap saling membantu mampu membangun kerukunan dalam bermasyarakat. Pada tafsir lengkap kemenag

dijelaskan bahwa bahagian terakhir ayat mewajibkan orang-orang mukmin tolong-menolong sesama mereka dalam berbuat kebaikan dan bertakwa, untuk kepentingan dan kebahagiaan mereka. Dilarang tolong-menolong dalam berbuat dosa dan pelanggaran serta memerintahkan supaya tetap bertakwa kepada Allah agar terhindar dari siksaan-Nya yang sangat berat.

Berdasarkan ayat Al-Quran dan tafsir tersebut dapat disimpulkan bahwa kita sebagai manusia harus saling tolong-menolong, begitu juga keterkaitan penelitian ini terhadap ayat Al-Quran dan tafsir tersebut. Penelitian ini bertujuan untuk membantu sesama manusia untuk penyebaran informasi mengenai simulasi keramaian yang berada di UIN Malang menggunakan *virtual reality*. Dengan adanya penelitian ini, diharapkan untuk orang-orang mampu merasakan suasana UIN Malang dimana saja dan kapan saja tanpa ada hambatan jarak dan waktu.

## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dikerjakan dengan simulasi keramaian di Kampus UIN Malik Ibrahim Kota Malang, maka dapat ditarik kesimpulan bahwa metode *Reciprocal Velocity Obstacle* dapat diterapkan dalam navigasi multi agen. Hal tersebut dikarenakan agen mampu melakukan pemilihan rute, menavigasi mandiri sehingga mampu menghindari tabrakan antar agen tanpa perlunya komunikasi.

Pada setiap simulasi yang dilakukan, tabrakan akan terjadi pada jumlah sekumpulan agen dikarenakan tingkat kepadatan yang tinggi. Hal tersebut terjadi karena kepadatan tinggi menyebabkan perhitungan yang kompleks untuk navigasi yang dilakukan oleh agen.

Berikut merupakan kesimpulan dari hasil pengujian RVO pada beberapa skenario di Kampus UIN Malik Ibrahim Kota Malang:

1. Hasil simulasi pada kampus skenario menggunakan RVO pada setiap rute menghasilkan *collision* terendah 23 terdapat pada jalur gerbang belakang menuju gedung rektorat sedangkan *collision* tertinggi yaitu 35 terdapat pada jalur gerbang mahad wanita menuju fakultas humaniora. Rata-rata *collision* yang terjadi pada tiap rute sebanyak 34
2. Hasil simulasi pada kampus skenario menggunakan RVO menghasilkan jumlah agen yang mencapai *destination* terendah dalam jangka waktu 200 detik adalah dari gerbang utama menuju gedung *sport center* dengan total agen 144

sedangkan jumlah agen yang mencapai *destination* terbanyak yaitu dari gerbang samping menuju perpustakaan dengan total agen 252. Rata-rata jumlah agen yang sampai titik tujuan pada setiap rute yaitu 94

3. Hasil simulasi pada kampus skenario menggunakan RVO menghasilkan rata-rata FPS tertinggi mencapai 102 pada rute gerbang mahad wanita menuju fakultas humaniora sedangkan rata-rata FPS terendah mencapai 49 pada rute gerbang utama menuju gedung B. Rata-rata *framerate* yang terjadi pada setiap rute yaitu 79

## 5.2 Saran

Dalam proses pembuatan hingga pengujian yang telah dilakukan, penulis menyadari bahwa masih banyak kekurangan yang dapat dikembangkan di kemudian hari. Beberapa diantaranya adalah sebagai berikut:

1. Penggunaan perangkat keras dengan spesifikasi lebih tinggi akan sangat disarankan agar dapat melakukan uji simulasi dengan jumlah agen yang lebih banyak.
2. *Obstacle* yang lebih bervariasi dapat ditambahkan agar simulasi dapat dengan mudah dipahami dan lebih mendekati realita
3. Simulasi dapat dikembangkan dengan menggunakan metode desain dan pengembangan yang lain sehingga memungkinkan simulasi berjalan dengan optimal

## DAFTAR PUSTAKA

- Douthwaite, J. A., Zhao, S., & Mihaylova, L. S. (2019). Velocity Obstacle Approaches for Multi-Agent Collision Avoidance. *Unmanned Systems*, 7(1), 55–64. <https://doi.org/10.1142/S2301385019400065>
- Fachri, M. (2016). *Simulasi Kerumunan Menggunakan Navigasi Multiagen Berbasis Reciprocal Velocity Obstacles dengan Relasi Leader Follower*. 99. <http://repository.its.ac.id/51417/>
- Fadila, J. N., & Arif, Y. M. (2020). Implementasi Algoritma RVO sebagai Sistem Kendali Gerombolan NPC pada Permainan Action RPG. *Matics*, 12(1), 87. <https://doi.org/10.18860/mat.v12i1.8959>
- Furqon, M. (2019). *Implementasi Algoritma Reciprocal Velocity Obstacle Untuk Crowd Simulation Wisata Alun-Alun Kota Batu*.
- Juniastuti, S., Fachri, M., Nugroho, S. M. S., & Hariadi, M. (2017). Crowd navigation using leader-follower algorithm based Reciprocal Velocity Obstacles. *2016 International Symposium on Electronics and Smart Devices, ISESD 2016*, 148–152. <https://doi.org/10.1109/ISESD.2016.7886709>
- Liu, Z., Jiang, Z., Xu, T., Cheng, H., Xie, Z., & Lin, L. (2018). Avoidance of High-Speed Obstacles Based on Velocity Obstacles. *Proceedings - IEEE International Conference on Robotics and Automation*, 7624–7630. <https://doi.org/10.1109/ICRA.2018.8463200>
- Macal, C. M., & North, M. J. (2008). Agent-based modeling and simulation: ABMS examples. *Proceedings - Winter Simulation Conference*, 101–112. <https://doi.org/10.1109/WSC.2008.4736060>
- Mongi, L. S., Lumenta, A. S. M., & Sambul, A. M. (2018). Rancang Bangun Game Adventure of Unsrat Menggunakan Game Engine Unity. *Jurnal Teknik Informatika*, 13(1). <https://doi.org/10.35793/jti.13.1.2018.20191>
- Muhammad, F., Juniastuti, S., Nugroho, S. M. S., & Hariadi, M. (2018). Crowds Evacuation Simulation on Heterogeneous Agent Using Agent-Based Reciprocal Velocity Obstacle. *Proceeding - 2018 International Seminar on Intelligent Technology and Its Application, ISITIA 2018*, 275–280. <https://doi.org/10.1109/ISITIA.2018.8711117>
- Saputri, T., Nugraha, C., & Amila, K. (2014). Model Simulasi Untuk Pergerakan Kendaraan Pada Ruang Dua Dimensi Kontinu Dengan Pendekatan Pemodelan Berbasis Agen Tari Saputri, Cahyadi Nugraha, Khuria Amila. *Jurnal Online Institut Teknologi Nasional Oktober*, 02(04), 2338–5081.

- Seniati, L. (2006). Pengaruh Masa Kerja, Trait Kepribadian, Kepuasan Kerja, Dan Iklim Psikologis Terhadap Komitmen Dosen Pada Universitas Indonesia (Vol. 10, Issue 2).
- Snape, J., Berg, J. Van Den, Guy, S. J., & Manocha, D. (2011). The hybrid reciprocal velocity obstacle. *IEEE Transactions on Robotics*, 27(4), 696–706. <https://doi.org/10.1109/TRO.2011.2120810>
- Syawab, M. H. (2020). Simulasi Pergerakan Agen Pengunjung Pada Wisata Pulau Pari Menggunakan Algoritma Artificial Bee Colony. 1–9.
- Zebua, T., Nadeak, B., & Sinaga, S. B. (2020). Pengenalan Dasar Aplikasi Blender 3D dalam Pembuatan Animasi 3D. *Jurnal ABDIMAS Budi Darma*, 1(1), 18–21.
- Zheng, J. M., Chan, K. W., & Gibson, I. (1998). Virtual reality. *IEEE Potentials*, 17(2), 20–23. <https://doi.org/10.1109/45.666641>