

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Kendaraan balap otonom telah menjadi domain utama dalam perkembangan teknologi otomotif dan telah menarik banyak perhatian dan minat akademis yang cukup besar dalam pengembangannya karena potensi dan prospeknya yang tinggi [1] [2] [3]. Dalam bidangnya kompetisi balap mobil otonom baik itu dalam *game* maupun dalam dunia nyata merupakan salah satu kompetisi yang sukses dan populer [4] [5] [6]. Pengembangan agen balap otonom penting untuk robotika bergerak karena menghadirkan masalah yang menantang yang memerlukan integrasi dari berbagai keterampilan, termasuk persepsi, kontrol, dan perencanaan jalur [7] juga salah satunya kemampuan kendaraan dalam menghindari penghalang [8]. Dalam pengaplikasian mobil robotik salah satunya kendaraan balap otonom, telah di implementasikan berbagai macam algoritma dan yang sering digunakan yaitu *Reinforcement learning* (RL) [2] [9] [10] untuk menyelesaikan beberapa masalah dan kekurangan atau inefisiensi dari pengaplikasian mobil robotik, salah satunya pada kendaraan balap otonom. Dalam beberapa studi terbaru yang meneliti tentang kendaraan balap otonom [2] [8], kendaraan mengalami kesulitan untuk bergerak secara otonom di lingkungan dinamis yang mana harus melakukan navigasi melalui lingkungan yang tidak terstruktur. Dalam kasus mobil otonom salah satu masalahnya yaitu untuk menghindari penghalang yang bersifat dinamis dimana transformasinya dapat berubah-ubah. Dalam beberapa penelitian sebelumnya, sebagian besar algoritma tradisional hanya efisien dalam lingkungan statis atau lingkungan yang tidak berubah-ubah transformasinya. Yang mana di lingkungan statis sebuah kendaraan balap otonom biasanya berfungsi berdasarkan informasi yang sudah dipetakan tentang kondisi lingkungan yang sudah ada. Artinya, menjadi masalah utama bagi kendaraan balap otonom untuk bergerak secara otonom di lingkungan dinamis seperti dalam kemampuannya menghindari penghalang dinamis secara terus-menerus [10]. Berdasarkan beberapa studi sebelumnya *Deep Reinforcement learning* (DRN) adalah metode yang sering

digunakan pada kasus mobil otonom dengan algoritmanya menggunakan *Deep Q-Network* (DQN) [1] [8]. DQN merupakan penggabungan dari algoritma *Q-learning* dan *Deep learning* dan merupakan salah satu algoritma DRN yang sering digunakan khususnya pada kasus mobil otonom [1].

Telah dilakukan beberapa penelitian tentang kendaraan otonom. Studi terbaru [2] membandingkan algoritma *Proximal Policy Optimization* (PPO) dan *Parallel Online Continuous Arcing* (POCA), penelitian ini berfokus untuk mengetahui performansi model dari kedua algoritma dengan mengukur *loss value* dan *qumulative reward*-nya. Penelitian yang serupa juga telah dilakukan [3] dengan membandingkan 3 algoritma termasuk PPO dan 2 lainnya yaitu DQN dan *Deep Deterministic Policy Gradient* (DDPG). Berbeda dengan penelitian sebelumnya, penelitian ini berfokus meningkatkan performansi dari kendaraan otonom. Pada penelitian lain, penulis [11] mengimplementasikan PPO dan DQN pada *drone* otonom virtual pada *Unreal Engine*, penelitian ini berfokus untuk mengukur peformansi dengan menghitung presentasi per-episode. Dilakukan pula penelitian dengan pengukuran yang sama oleh penulis [12] Membandingkan DDPG dan PPO pada kendaraan otonom.

Dasar dari Penelitian ini yaitu untuk memperluas area penelitian sebelumnya, Savid *et al* [2] menyarankan untuk dapat mengimplementasikan algoritma yang berbeda dikarenakan *Framework* yang digunakan yaitu *ML-Agent* hanya menyediakan sebagian kecil algoritma untuk dipilih. pekerjaan lainnya yaitu untuk dapat membuat agen dapat berkendara pada lingkungan yang dinamis seperti *track* balap yang berbeda dan penghalang dinamis. hal ini bertujuan agar dapat mencegah agen *overfitting* ke dalam lingkungan dimana agen dilatih.

Berdasarkan pada beberapa penelitian sebelumnya DQN dapat diperkirakan merupakan algoritma yang paling tepat untuk menyelesaikan permasalahan ini karena keunggulannya dalam hal performansi dan keakuratan dibandingkan dengan algoritma lain. dalam pengujianya dengan mengukur *Cummulative Reward* ataupun *Q-value* per-episodenya dan terbukti pada penelitian [3] Menunjukkan bahwa DQN unggul dibandingkan dengan 2 algoritma lainnya yaitu PPO dan DDPG, pada penelitian lain [11] dengan kasus yang berbeda yaitu *drone* otonom, DQN juga

mengungguli PPO. Dapat disimpulkan bahwa DQN sangat mungkin untuk diterapkan pada kemampuan kendaraan balap otonom dalam menghindari penghalang dinamis.

Oleh karena itu, tujuan utama dari penelitian ini yaitu untuk mengimplementasikan algoritma yang berbeda dari penelitian sebelumnya [2] dengan menggunakan DQN pada simulasi mobil balap yang akan menjadi keluaran dari penelitian ini. Penelitian ini akan berfokus untuk menguji performansi algoritma DQN yang di implementasikan pada mobil balap otonom untuk menghindari penghalang yang bersifat dinamis dan mencapai posisi *goal*.

1.2 Rumusan Masalah

Bagaimana menguji performansi algoritma DQN pada kendaraan balap otonom di lingkungan dinamis agar dapat menghindari penghalang dinamis.

1.3 Maksud dan Tujuan

Dari latar belakang dan rumusan masalah di atas, maka maksud dan tujuan penelitian yaitu sebagai berikut :

1.3.1 Maksud

Mengimplementasikan algoritma DQN pada kendaraan balap otonom di lingkungan dinamis.

1.3.2 Tujuan

Menguji performansi algoritma DQN pada kendaraan balap otonom di lingkungan dinamis dengan mengukur *cummulative reward*, *loss value* pada tahap *training* lalu mengukur jumlah capai posisi goal dan jumlah tabrakan pada tahap *testing*.

1.4 Ruang Lingkup Masalah

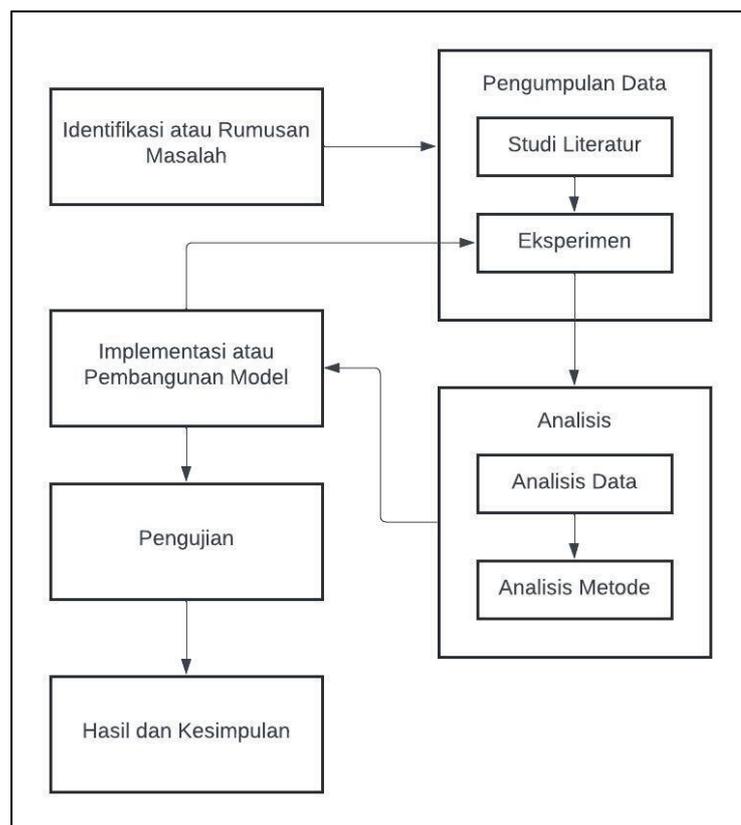
Penelitian ini dibatasi oleh hal-hal berikut :

1. Simulasi dilakukan secara virtual menggunakan *Unity Engine*.
2. Hanya satu kendaraan berupa mobil yang akan disimulasikan
3. Sifat dinamis dari lingkungan yaitu berupa perpindahan penghalang ban mobil yang bergerak ke kiri atau kanan dengan kecepatan yang konstan sebesar 2 frame/unit pada area *track* balap

4. Lingkungan hanya menggunakan 2 *track* balap, track lurus dan melingkar.
5. Kendaraan hanya dapat mendeteksi objek yang ada di bagian depan kendaraan sebesar 90 derajat.
6. Kendaraan mengobservasi lingkungan dengan menggunakan deteksi berbasis sensor.
7. Bahasa pemrograman yang digunakan yaitu C# dan Python.

1.5 Metodologi Penelitian

Metodologi penelitian yang akan digunakan dalam penelitian ini adalah kuantitatif eksperimental, untuk lebih jelasnya dapat dilihat pada gambar 1.1 Metodologi Penelitian



Gambar 1.1 Metodologi Penelitian

1.5.1 Identifikasi Atau Rumusan Masalah

Pada tahap awal penelitian akan dilakukan identifikasi atau rumusan masalah, yaitu tentang berbagai macam algoritma yang digunakan pada simulasi kendaraan otonom. Fokus dari penelitian ini adalah menguji performansi algoritma DQN yang di implementasikan pada mobil balap otonom untuk menghindari penghalang yang bersifat dinamis dan mencapai posisi *goal*.

1.5.2 Pengumpulan Data

Pengumpulan data pada penelitian ini akan dilakukan dengan dua tahap yaitu melalui hasil studi literatur dan juga eksperimen.

1. Studi Literatur

Pada tahap ini dilakukan pengumpulan data melalui *review* literatur seperti jurnal, artikel dan buk-buku yang berkaitan. Data yang akan di kumpulkan yaitu berupa data yang nantinya akan digunakan pada pemodelan agen mobil balap otonom, seperti data input, *reward*, terminal dan juga *output*. kemudia data-data lain seperti parameter yang digunakan pada algoritma DQN.

2. Eksperimen

Tahap ini mencakup proses pembuatan lingkungan simulasi sebagai sumber data yang digunakan untuk proses pelatihan model yang akan dibangun. *Environment* dibuat dalam bentuk model 3D yang kemudian akan dimasukkan ke dalam *Unity Engine*. *Environment* yang akan dibuat yaitu berupa *track* balap dan berbagai jenis objek penghalang. Tahap eksperimen juga merupakan proses merubah data yang dianggap kurang baik pada saat proses implementasi atau pembangunan model.

1.5.3 Analisis

Tahapan Analisis akan dibagi menjadi dua yaitu analisis data dan analisis metode.

1. Analisis Data

Pada tahapan ini dilakukan proses analisis dari data yang sudah diperoleh baik itu dari hasil studi literatur maupun hasil eksperimen. tahapan

ini akan mencakup proses penyesuaian data dengan model yang akan dibangun juga visualisasi data.

2. Analisis Metode

Tahap selanjutnya yaitu menganalisis metode untuk membangun model mobil balap otonom menggunakan DQN seperti menganalisis arsitektur jaringan, algoritma DQN, parameter yang digunakan, proses pelatihan dan juga metode pengujian.

1.5.4 Implementasi Atau Pembangunan Model

Pada tahap ini, peneliti membangun dan mengimplementasikan model DQN pada mobil balap otonom seperti mendesain arsitektur ANN yang akan digunakan dalam DQN. Ini termasuk menentukan jumlah *layer*, jumlah node per lapisan, fungsi aktivasi, dan struktur DQN. Kemudian mengimplementasikan algoritma DQN menggunakan bahasa pemrograman C# di *Unity Engine*. Setelah dilakukan proses implementasi model maka selanjutnya akan dilakukan proses pelatihan model (*training*). Kemudian dilakukan evaluasi kinerja model DQN dengan menguji kemampuan mobil dalam lingkungan simulasi. jika data yang digunakan pada saat proses *training* model tidak sesuai, maka akan dilakukan perulangan proses yaitu pada proses analisis data yang akan dilakukan eksperimen data sampai menemukan data yang sesuai untuk diimplementasikan pada model.

1.5.5 Pengujian

Tahap ini merupakan tahap pelatihan dan pengujian dari model yang sudah dibangun, proses *training* dilakukan sampai model dapat mencapai hasil yang diinginkan. Dalam proses *training* juga akan dilakukan pengujian, variabel yang akan digunakan pada penelitian adalah seberapa banyak penghalang yang berhasil dihindari dari setiap iterasi, *cumulative reward* dan juga *loss value*.

1.5.6 Hasil dan Kesimpulan

Pada tahap akhir dilakukan hasil dari pengujian model mengenai keakuratan model dalam menghindari penghalang dinamis dalam simulasi balap mobil otonom. Kesimpulan yang diambil dapat memberikan pemahaman mendalam mengenai keakuratan model dan saran-saran untuk pengembangan pada penelitian berikutnya.

1.6 Sistematika Penulisan

Untuk mempermudah pemahaman yang komprehensif terhadap seluruh isi skripsi, maka perlu dibuat sebuah sistematika penulisan yang berfungsi sebagai kerangka dan panduan dalam penulisan. Penelitian ini akan dibagi menjadi beberapa bagian diantaranya :

BAB 1 Pendahuluan

Bab ini terdiri dari latar belakang, rumusan masalah, tujuan, ruang lingkup masalah, metode penelitian dan sistematika penulisan skripsi.

BAB 2 Tinjauan Pustaka

Bab ini berisi Landasan teori yang berisi tentang pembahasan pengertian tentang mobil balap otonom, lingkungan dinamis, *Dynamic Obstacle avoidance* (DOA), *Artificial Intelligence* (AI), Pembelajaran mesin, *Reinforcement learning*, *Q learning*, *Deep learning*, *Deep Reinforcement learning*, *Deep Q Network*, *Unity Engine* dan bahasa Pemrograman C#

BAB 3 Analisis dan Perancangan

Pada Bab ini berisi beberapa rangkaian analisis dan perancangan sistem yang terdiri dari Analisis masalah, Analisis metode, analisis algoritma, analisis kebutuhan non fungsional, analisis kebutuhan fungsional dan perancangan sistem.

BAB 4 Implementasi dan Pengujian

Pada Bab ini berisi beberapa rangkaian implementasi dan pengujian yang terdiri dari implementasi perangkat keras implementasi perangkat lunak, implementasi antar muka. pengujian fungsional dan pengujian performansi.

BAB 5 Kesimpulan dan Saran

Bab ini terdiri dari penarikan kesimpulan dari hasil penelitian dan saran untuk penelitian selanjutnya.