

GERAKAN NPC PADA GAME PENGENALAN HURUF MENGGUNAKAN ALGORITMA DIJKSTRA DENGAN PERBEDAAN KETINGGIAN PADA OBSTACLE

Angga Junianto Sadewantoro

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Wijaya Kusuma Surabaya
 Jl. Dukuh Kupang XXXV/54 60225, Jawa Timur, Indonesia
angga.sadewantara@gmail.com

Abstrak

Game adalah permainan yang digunakan sebagai media hiburan ataupun untuk pembelajaran. Pada bagian *game* pastinya ada sebuah karakter yang tidak dapat dimainkan akan tetapi berperan untuk melanjutkan jalan cerita pada *game* tersebut yakni NPC (*Non Player Character*). Dari penelitian sebelumnya yang bertemakan pergerakan NPC menggunakan algoritma dijkstra pada permainan pengenalan huruf terdapat masalah apabila lingkungan yang didapatkan terdapat halang rintang atau biasa disebut *obstacle* NPC tersebut tidak dapat melakukan gerakan berikutnya jika terdapat *obstacle* dengan ketinggian tertentu. Dari permasalahan tersebut maka penulis bermaksud mengembangkan untuk menambahkan gerakan NPC dengan tindakan dapat melewati sebuah *obstacle* dengan ketentuan NPC dapat melompatnya jika tinggi *obstacle* $\frac{1}{4}$ dari tinggi NPC tersebut. Dalam mengatur gerakan NPC untuk menuju ke target biasanya sering mengalami kendala sebagaimana apabila terdapat berbagai macam *obstacle* yang ada disekitar lingkungannya terkadang NPC tersebut sulit untuk menentukan gerakan yang akan dilakukan. Pathfinding adalah metode yang sangat mendukung untuk mempermudah gerakan NPC dalam mencari target utama. Pathfinding yang disertai dengan algoritma dijkstra sangat cocok untuk memberi *Artificial Intelligence* pada NPC. Perhitungan algoritma dijkstra yang tepat dengan seluruh area untuk mencari jarak terdekat menuju target.

Kata Kunci: game, pathfinding, NPC, algoritma dijkstra.

Abstract

Game is a play used as a medium of entertainment or for learning. In the *game* certainly there is a character that can not be played but the role to continue the story in the *game* that is NPC (*Non Player Character*). From previous research with the theme of NPC movement using algorithm dijkstra in *game* introduction of letters there is a problem if the environment obtained there is *obstacle* or commonly called *obstacle* NPC can not do the next movement if there is *obstacle* with a certain height. From the problem then the author intends to develop to add NPC movements with actions can pass an *obstacle* with the provision that the NPC can jump if the *obstacle* height is $\frac{1}{4}$ of the height of the NPC. In managing the NPC movement to go to the target is usually often experienced constraints as if there are various kinds of *obstacles* that exist around the environment sometimes NPC is difficult to determine the movement to be done. Pathfinding is a very supportive method to ease the NPC movement in finding the main target. The pathfinding accompanied by the dijkstra algorithm is perfect for giving *Artificial Intelligence* to the NPC. Calculate the exact algorithm with the whole area to find the closest distance to the target.

Keywords: game, pathfinding, NPC, Algorithm dijkstra.

I. PENDAHULUAN

Dari penelitian sebelumnya (Nestia Febriyani Putri, 2015) yang bertemakan pergerakan NPC menggunakan algoritma dijkstra pada permainan pengenalan huruf terdapat masalah apabila lingkungan yang didapatkan terdapat halang rintang atau biasa disebut *obstacle* NPC tersebut tidak dapat melakukan gerakan berikutnya jika terdapat *obstacle* dengan ketinggian tertentu. Dari permasalahan tersebut maka penulis bermaksud mengembangkan untuk

menambahkan gerakan NPC dengan tindakan dapat melewati sebuah *obstacle* dengan ketentuan NPC dapat melompat jika tinggi *obstacle* $\frac{1}{4}$ dari tinggi NPC tersebut.

Non Player Characters (NPC) merupakan bagian penting di dalam sebuah permainan komputer. Peran komputer games (CRPGs) dan strategi permainan, menempatkan manusia pemain di dunia maya yang dihuni oleh komputer yang dikendalikan agen, yang disebut karakter non-pemain (NPC). Agen-agen ini dapat mengisi tiga peran: sebagai pendamping,

sebagai lawan, dan sebagai netral, latar belakang karakter. NPC tersebut juga dapat diberikan gerakan dengan menggunakan pathfinding untuk menyelesaikan masalah menangkap suatu target. (JinHyuk Hong and Sung-Bae Cho, 2005).

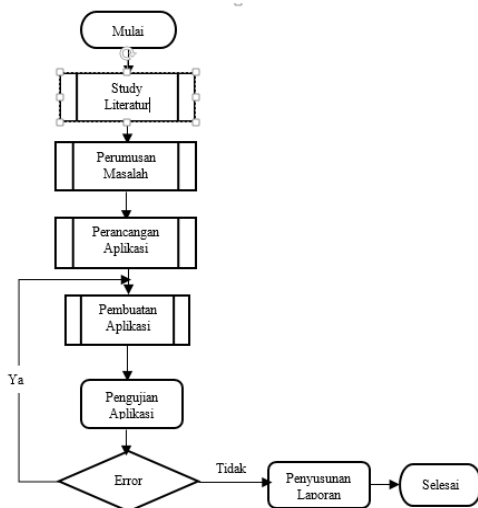
Salah satu algoritma yang dapat diterapkan untuk pathfinding adalah algoritma djikstra. Beberapa peneliti telah menggunakan algoritma tersebut untuk menentukan jarak terdekat. Algoritma ini sangat tepat karena perhitungannya dalam menentukan rute terpendek suatu NPC untuk mencapai target dengan gerakan yang bermacam-macam dan realistik secara *visual* namun karena komputasi tidak dilakukan secara *real-time*, *path planning* akan sulit diterapkan pada lingkungan yang bersifat dinamis. Pada game yang berjenis seperti *tower defense* pergerakan NPC untuk mencapai target sangat cocok untuk melakukan pathfinding namun dengan beberapa obstacle yang ada NPC tersebut diharapkan mampu melewatinya. *Tower defense* adalah *game* yang dalam permainannya mementingkan pertahanan pada player agar tidak mudah diserang oleh musuh. Dari uraian tersebut maka peneliti akan mengangkat judul “Gerakan NPC Pada *Game* Pengenalan Huruf Menggunakan Algoritma Dijkstra Dengan Perbedaan Ketinggian Pada *Obstacle*”.

II. METODE

Metode Penelitian berisi langkah langkah yang digunakan dalam penelitian ini agar terstruktur dengan baik.

2.1 Diagram Alir

Untuk membantu dalam penyusunan penelitian ini maka perlu adanya diagram alir yang jelas setiap tahap-tahapnya. Diagram alir ini merupakan langkah-langkah yang akan dilakukan dalam penyelesaian yang akan dibahas.



Gambar 1. Diagram Alir

2.2 Perumusan Masalah

Merumuskan tentang memberi variatif pergerakan NPC dalam mencari jarak untuk mengejar target dengan memperhatikan *obstacle* yang ada di rute yang akan ditempuhnya. Pada Permainan Pengenalan Huruf dengan perbedaan ketinggian *obstacle*.

2.3 Perancangan Aplikasi

Dalam proses perancangan aplikasi ini digunakan untuk merancang semua yang dibutuhkan untuk menyelesaikan aplikasi ini. Awalnya dengan membuat skenario, storyboard, gameplay, FSM, flowchart dan juga perancangan grafik seperti model huruf dan karakternya.

2.4 Pembuatan Aplikasi

Awalnya pada proses ini dibuatkan model 3d baik mengenai hurufnya ataupun model untuk karakter yang digunakan player, kemudian mengatur gerak npc yang menggunakan djikstra dengan ketentuan diberi obstacle pada lingkungannya, yang terakhir penulisan script untuk membuat npc huruf tersebut dapat melompati obstacle dengan tinggi $\frac{1}{4}$ dari tinggi npcnya untuk menuju ke targetnya tersebut.

III. HASIL DAN PEMBAHASAN

Pada bab ini membahas hasil dan pembahasan variatif dalam gerakan NPC dengan djikstra yang diberi ketinggian obstacle berbeda.

3,1 Main Menu



Gambar 2 tampilan main menu

Pada menu awal *game*, terdapat dua tombol yaitu “Play” dan “Quit”. Jika pemain ingin masuk kedalam *game* maka pemain perlu menekan tombol “Play” dan jika ingin keluar maka pemain harus menekan tombol “Quit”. Di menu awal terdapat *background* yang menyimbolkan sebagaimana *game* yang akan dimainkan.

3.2 Tampilan dalam Permainan



Gambar 3. Tampilan dalam *Game*

Tampilan permainan adalah tampilan utama dalam *game* pengenalan huruf dengan dijkstra yang diberi obstacle. Pada bagian kiri atas terdapat *health bar* yang berfungsi sebagai sisa darah yang dimiliki oleh *player*. Pada kanan atas terdapat batas waktu yang harus dilakukan agar dapat memenangkan *game* tersebut tanpa terbentur oleh huruf yang akan mengikutinya. Kanan bawah ada *ammo bar* yang berfungsi untuk menginformasikan sisa peluru yang dimiliki oleh *player*. Pada sekitar lingkungannya terdapat obstacle yang diberi ketinggian berbeda dapat dilompati oleh musuh yang akan mengejar *player* apabila memenuhi kriteria yang ditentukan.

3.3 Tampilan Berhasil



Gambar 4. Tampilan Berhasil

Jika pemain berhasil menembak semua huruf yang mengejarnya maka akan dipindahkan ke scene menang tersebut. Pada scene ini terdapat tombol main menu dan next level. Apabila menekan main menu akan kembali pada scene awal dan apabila tombol next level ditekan akan langsung menuju scene next level yang tersedia dengan musuh yang diberi kecepatan untuk menangkap target lebih cepat.

3.4 Tampilan Kalah



Gambar 5. Tampilan Kalah

Player akan memasuki scene ini apabila *health bar* yang ditunjukkan berada pada angka 0 dengan kata lain *player* tertabrak musuh sebanyak 5 kali yang membuat *health bar* akan habis. Terdapat tombol *Back* yang berguna untuk kembali ke main menu apabila ingin mengulangi *game* tersebut..

IV. PENUTUP

4.1. Kesimpulan

Kesimpulan yang dapat diambil dari *game* dengan dijkstra dan berbagai macam obstaclenya yakni:

1. Apabila musuh/huruf diberikan kecepatan yang tinggi dan juga turn speed tinggi maka kesempatan untuk menabrak *player* menjadi lebih berkurang akibat saat berbelok akan melebihi jalur yang telah ditentukan.
2. Dengan adanya timer dapat membuat *player* menjadi terpacu untuk segera menembak semua musuh yang akan datang kepadanya.

4.2. Saran

Saran yang diberikan untuk pengembangan Gerakan NPC Pada *Game* Pengenalan Huruf Menggunakan Algoritma Dijkstra Dengan Perbedaan Ketinggian Pada Obstacle :

1. Pengembang perlu memperindah grafik yang ditampilkan agar lebih menarik.
2. Menambahkan lebih banyak jenis obstacle agar dapat menjadi poin tersendiri.

DAFTAR PUSTAKA

- [1] JinHyuk Hong and Sung-Bae Cho. (2005). Evolving Reactive NPCs for the Real Time Simulation Game. 16.
- [2] Nestia Febriyani Putri. (2015). *Gerakan NPC Pada Permainan Pengenalan Huruf Menggunakan Algoritma Dijkstra*. Surabaya.

- [3] Novandi, R. A. (2007). Perbandingan Algoritma Dijkstra dan Algoritma Floyd-Warshall dalam. *MAKALAH IF2251 STRATEGI ALGORITMIK TAHUN 2007*, 2.
- [4] Yunifa Miftachul Arif, F. K. (2011). Desain Perubahan Perilaku pada NPC Game Menggunakan Logika Fuzzy. *Seminar On Electrical, Informatics, and ITS Education*, A2-107.
- [5] Zaidah, N. (2016). Penerapan Algoritma Dijkstra Untuk Game Hijaiyah. 26.