

APLIKASI UNTUK MENAMPILKAN 3D-MESH DENGAN SKELETAL ANIMATION PADA BOARD GAME

Kiat Stanley¹, Liliana², Gregorius Satia Budhi³

Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra

Siwalankerto 121-131, Surabaya, Indonesia

Email: somebody.weknow@gmail.com¹, lilian@petra.ac.id², greg@petra.ac.id³

ABSTRAK

Animasi adalah bagian yang tak terpisahkan dari video game. Animasi dalam video game bisa menghasilkan berbagai gerakan pada objek-objek dalam game. Game yang menggunakan animasi yang sangat minim akan menghasilkan kesan yang monoton. Sistem animasi pada pengembangan game tidak bisa dhiraukan.

Untuk dapat menghasilkan game dengan animasi yang baik, diperlukan sebuah metode untuk menganimasikan objek-objek dalam game. Salah satu metode yang digunakan adalah metode Skeletal Animation. Skeletal Animation adalah metode dimana pada sebuah objek diberikan kerangka sehingga objek bisa bergerak mengikuti kerangka dari objek.

Dengan menggunakan metode Skeletal Animation, bisa dihasilkan sebuah game dengan animasi yang menarik dan terkesan tidak monoton sehingga pengguna dapat menikmati permainan.

Kata Kunci

Game, Animasi, Skeletal Animation.

ABSTRACT

Animation is an inseparable part of video games. Animation in video games is capable of producing various movements on objects in the game. Games that use animation minimally can result a monotone feel. Animation system in game development cannot be overlooked.

To produce a game with an acceptable animation, a method to animate game objects is required. One of the methods is Skeletal animation. Skeletal animation is an animation method wherein a skeleton is given to an object so the object can move in accordance with the skeleton.

By using Skeletal Animation method, a game with captivating animation and less monotone feel can be produced.

Keywords

Game, Animation, Skeletal Animation.

1. PENDAHULUAN

Game yang berkembang dewasa ini adalah game 3D. Menampilkan grafik dari game 3D lebih kompleks dibandingkan game 2D. Terdapat berbagai tantangan seperti cara menampilkan bayangan dari sebuah objek, cara mensimulasikan dua atau lebih objek yang bertabrakan, cara memanipulasi kamera sehingga tepat untuk game,

dan masih terdapat berbagai contoh lainnya. Dengan berbagai detail yang harus lebih diperhatikan dibanding game 2D, game 3D cenderung lebih berat untuk diproses oleh komputer, dan game harus diproses dengan cepat sehingga tidak terjadi lag pada permainan yang menyebabkan game kurang bisa dinikmati oleh pemain.

Grafik 3D memiliki peran yang cukup signifikan dalam game, baik secara gameplay maupun secara estetika. Grafik 3D memberikan kesan yang lebih realistis. Jika karakter pada game 2D digambarkan dengan bentuk sprite, karakter pada game 3D bisa digambarkan dalam bentuk objek 3D. Jika pada game 2D karakter hanya bisa bergerak ke kiri, kanan, atas, dan bawah, karakter pada game 3D bisa melakukan gerakan tersebut ditambah dengan gerakan ke depan dan belakang, sehingga ruang gerak pada game 3D lebih luas dibanding 2D.

2. DASAR TEORI

2.1 Pengertian Game

Definisi dari game adalah sebuah aktivitas interaktif sukarela, dimana satu atau lebih pemain mengikuti peraturan yang membatasi perilaku pemain-pemain tersebut, memberlakukan sebuah konflik yang bisa menghasilkan akhir yang jelas dan bisa dihitung [1], sedangkan definisi dari video game adalah game yang bisa dimainkan dengan bantuan perangkat audiovisual dan bisa berdasarkan sebuah cerita [2].

Salah satu genre dari game adalah strategy game. Strategy game adalah genre dari game yang menekankan kemampuan berpikir dan berencana dari pemain untuk mencapai kemenangan. Hal yang membedakan strategy game dengan genre lainnya adalah adanya penekanan utama pada kemampuan pemain dalam mengalahkan strategi pemain lawan [3].

2.2 Pengertian Mesh

Mesh adalah representasi dari sebuah model 3D yang terdiri dari kumpulan poligon yang terdiri dari titik-titik 3D dan bagian tepi yang menghubungkannya [4].

2.3 Pengertian Animasi

Animasi adalah sebuah proses yang menghasilkan ilusi gerakan. Hal tersebut dilakukan dengan menampilkan lebih dari satu potongan gambar secara bergantian dengan sangat cepat sehingga manusia menangkap gambar tersebut sebagai gerakan [5].

2.4 Pengertian *Skeletal Animation*

Salah satu metode untuk menganimasikan objek 3D adalah dengan metode *skeletal animation*. *Skeletal animation* adalah teknik animasi komputer dimana sebuah kerangka yang terdiri dari tulang-tulang yang saling terhubung dibuat dan dipasang pada sebuah mesh. Permukaan dari mesh akan mengikuti gerakan dari tulang yang telah dipasang. Teknik ini digunakan pada objek-objek yang tidak amorf, seperti pada model manusia dan binatang. Dewasa ini terdapat berbagai jenis video game yang menggunakan metode *skeletal animation* untuk menganimasikan objek [6].

2.5 A* Search Algorithm

Adalah algoritma yang digunakan untuk menentukan rute gerakan dari sebuah peta. Algoritma ini digunakan secara luas karena akurat dan cepat dalam menemukan rute [7].

2.6 Bresenham's Line Algorithm

Adalah algoritma yang bisa membentuk aproksimasi garis antara dua titik dan digunakan untuk menggambar garis pada komputer [8]. Perhitungan Bresenham dapat dilihat pada rumus (1).

$$y = ((y1 - y0) / (x1 - x0)) * (x - x0) + y0 \quad (1)$$

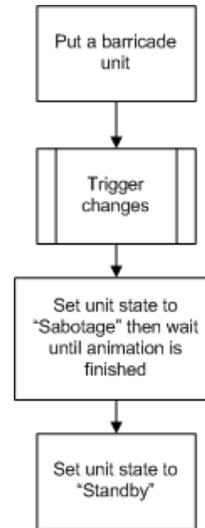
3. DESAIN SISTEM

Aplikasi yang dibuat adalah *video game* bergenre *turn-based strategy* (TBS) dengan tema perang gerilya Jenderal Sudirman. Akan dibahas mengenai desain grafik dan manipulasi objek 3D dalam game.

Yang dibuat dalam aplikasi adalah antara lain pembuatan dari data unit dan dunia dari game, manajemen *Hit Points* dan *Action Points* dari tiap unit, segala bentuk penempatan objek 2D dan 3D dalam aplikasi, dan animasi dari setiap objek 3D dalam aplikasi.

3.1 Desain Animasi Barricade

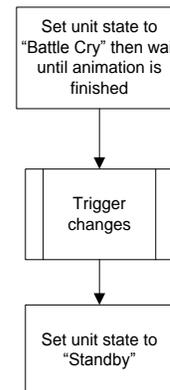
Jika sebuah unit diperintahkan oleh pemain untuk membangun barikade, unit akan dihadapkan ke arah dimana barikade akan dibangun. Animasi membangun dari unit tersebut dipanggil dan barikade diletakkan dalam game. Unit tersebut lalu terlindung oleh barikade yang dibangun dan akan memanggil animasi jongkok. Gambar 1 menunjukkan skema dari animasi ini.



Gambar 1. Skema dari animasi Baricade

3.2 Desain Animasi Battle Cry

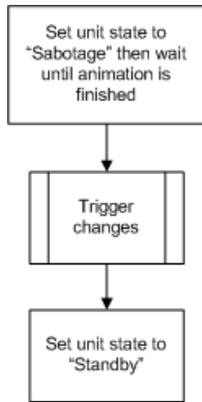
Jika sebuah unit diperintahkan untuk melakukan *action* ini, unit akan memanggil animasi untuk menyuarakan semangat. Semua unit lain yang terkena efek dari perintah ini jika bisa juga akan memanggil animasi tersebut. Gambar 2 menunjukkan skema dari animasi ini.



Gambar 2. Skema dari animasi Battle Cry

3.3 Desain Animasi Sabotage

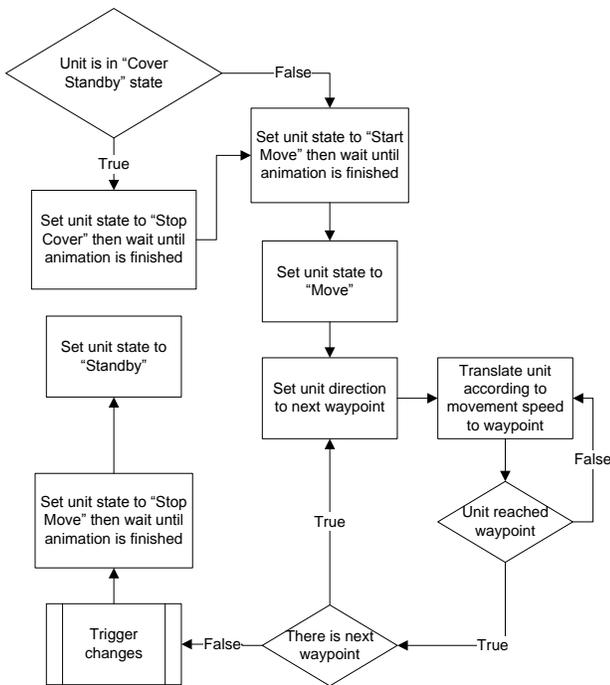
Jika sebuah unit diperintahkan untuk melakukan *action* ini, unit akan memanggil animasi menyabotase. Unit lalu kembali ke animasi semula. Gambar 3 menunjukkan skema dari animasi ini.



Gambar 3. Skema dari animasi Sabotage

3.4 Desain Animasi Move

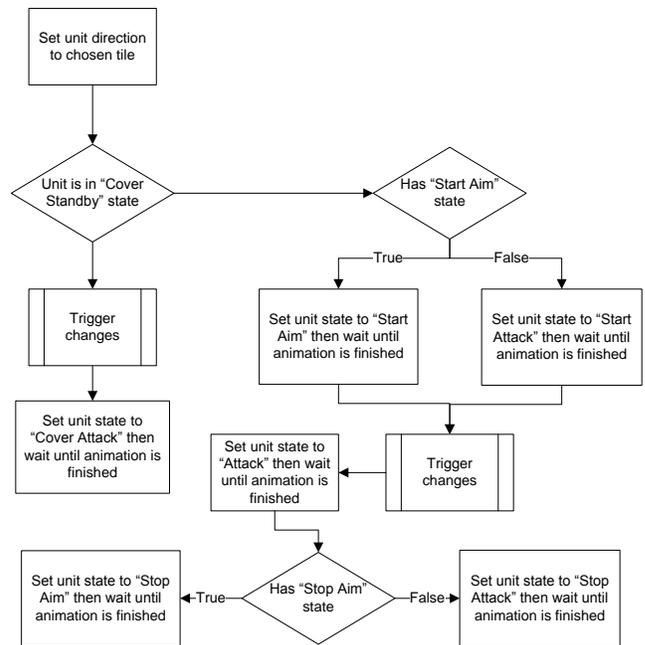
Jika unit diperintahkan untuk bergerak ke sebuah tile, unit tersebut akan bergerak mengikuti rute yang dibuat oleh aplikasi. Unit tersebut tidak dapat bergerak menembus penghalang atau unit musuh, tetapi bisa bergerak menembus unit teman. Unit akan berhenti bergerak ketika unit mencapai tile yang dituju. Gambar 4 menunjukkan skema dari animasi ini.



Gambar 4. Skema dari animasi Move

3.5 Desain Animasi Attack

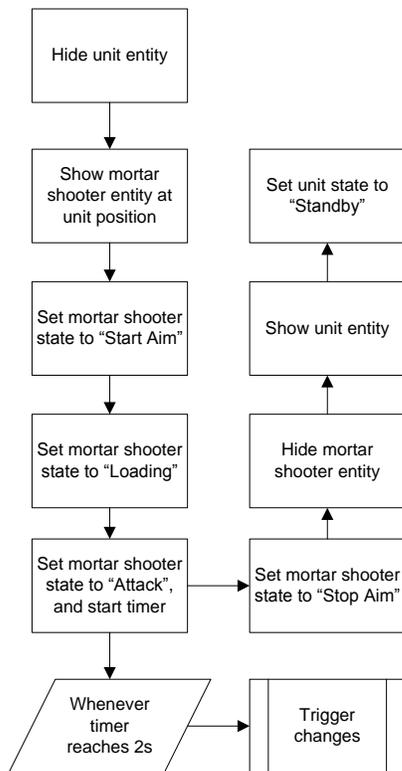
Jika unit diperintahkan untuk menyerang unit lain, unit tersebut dihadapkan ke unit yang akan diserang. Unit lalu memanggil animasi mempersiapkan serangan dan unit memanggil animasi melakukan serangan. Pada awal dari jalannya animasi serangan, unit yang terkena serangan akan memainkan animasi terkena serangan. Unit yang diperintahkan lalu kembali ke animasi semula, Gambar 5 menunjukkan skema dari animasi ini.



Gambar 5. Skema dari animasi Attack

3.6 Desain Animasi Mortar

Jika unit diperintahkan untuk menyerang menggunakan mortir, unit tersebut ditukar sementara dengan unit lain yang bersenjata mortar. Unit baru tersebut dihadapkan ke arah serang dan memainkan animasi mempersiapkan serangan. Unit tersebut lalu memanggil animasi serangan, dan beberapa waktu kemudian unit-unit lain yang terkena efek dari serangan memainkan animasi terkena efek serangan. Setelah animasi berakhir, unit yang asli dikembalikan ke posisinya dan unit yang membawa mortir dihilangkan dari permainan. Gambar 6 menunjukkan skema dari animasi ini.



Gambar 6. Skema dari animasi Attack

4. IMPLEMENTASI SISTEM

Animasi pada aplikasi dilakukan dengan bantuan OGRE. OGRE dapat memanggil animasi yang telah tersimpan dalam file model 3D. Jika misalnya animasi “Walk” pada model dipanggil, model akan melakukan animasi yang bernama “Walk”.

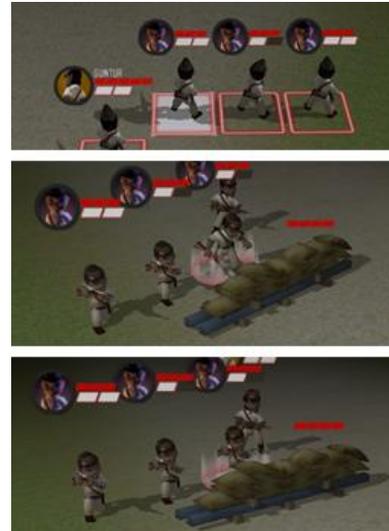
Ketika unit dalam game diperintahkan untuk melakukan sebuah aksi, unit tersebut akan menjalankan animasi-animasi sesuai dengan desain sistem yang telah dibuat sambil digerakkan dan diarahkan. Unit-unit yang terkena efek dari aksi sebuah unit juga akan memainkan animasi-animasi yang telah ditentukan. Unit yang mati dalam permainan juga akan memainkan animasi matinya unit tersebut sebelum dihilangkan dari permainan.

5. PENGUJIAN SISTEM

Pengujian dilakukan dengan memerintahkan unit melakukan sebuah *action*. Hasil pengujian adalah animasi yang dihasilkan dengan *action* tersebut. Pengujian dilakukan menggunakan laptop ASUS N46V. *Operating system* yang digunakan adalah *Microsoft Windows 7*.

5.1 Pengujian Animasi Barricade

Berikut ini adalah hasil pengujian pada animasi untuk action Barricade. Pada Gambar 7, unit yang terlindung oleh barikade memanggil animasi jongkok. Unit yang membangun barikade juga memanggil animasi jongkok karena unit tersebut terlindung oleh barikade.



Gambar 7. Pengujian Animasi Barikade

5.2 Pengujian Animasi Battle Cry

Berikut adalah hasil dari pengujian pada animasi untuk action Battle Cry. Pada Gambar 8, semua unit yang terkena efek Battle Cry yang memiliki animasi “Battle Cry” memanggil animasi tersebut.



Gambar 8. Pengujian Animasi Battle Cry

5.3 Pengujian Animasi Sabotage

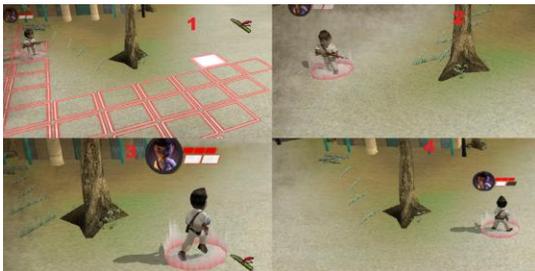
Berikut adalah hasil dari pengujian pada animasi untuk action Sabotage. Unit yang melakukan action ini akan melakukan sabotase pada pos milik musuh. Walaupun efek pada permainan belum diimplementasikan, animasi sudah bisa dilakukan seperti yang terlihat pada Gambar 9.



Gambar 9. Pengujian Animasi Sabotage

5.4 Pengujian Animasi Move

Berikut adalah hasil dari pengujian pada animasi untuk action Move. Ketika unit diperintahkan untuk bergerak pada sebuah *tile*, aplikasi akan membuat sebuah rute menuju *tile* tersebut. Unit akan bergerak mengikuti rute tersebut tanpa menembus halangan seperti unit musuh atau tembok. Unit bisa bergerak menembus unit teman. Ketika unit mencapai *tile* yang dituju, unit berhenti bergerak. Hasil pengujian bisa dilihat pada Gambar 10.



Gambar 10. Pengujian Animasi Move

5.5 Pengujian Animasi Attack

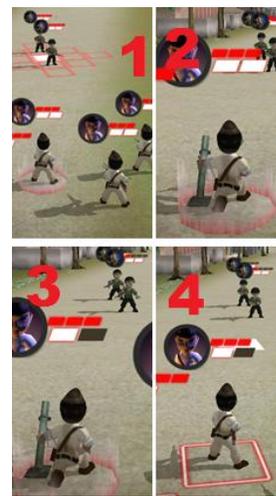
Berikut adalah hasil dari pengujian pada animasi untuk action Attack. Seperti yang didemonstrasikan pada Gambar 11, unit yang menyerang akan dirotasikan ke arah unit yang diserang. Unit lalu mempersiapkan serangan ke lawan. Ketika unit mulai menyerang, unit yang diserang akan memanggil animasi terserang. Unit lalu kembali ke animasi semula.



Gambar 11. Pengujian Animasi Attack

5.6 Pengujian Animasi Mortar

Berikut adalah hasil dari pengujian pada animasi untuk action Mortar. Unit yang memanggil *action* mortar akan dihilangkan dari game dan diganti dengan model unit lain yang membawa senjata mortar. Seperti pada *action* attack, unit tersebut diarahkan ke *tile* yang dituju. Unit lalu memanggil animasi mempersiapkan serangan. Unit lalu memanggil animasi serangan, dan beberapa detik kemudian unit-unit yang terkena efek dari serangan memainkan animasi terserang. Unit tersebut lalu dihilangkan dari game dan ditukar dengan unit yang sesungguhnya. Hal ini dilakukan karena model unit yang membawa senjata standard dan model unit yang membawa senjata mortar adalah model yang berbeda. Hasilnya bisa dilihat pada Gambar 12.



Gambar 12. Pengujian Animasi Mortar

6. KESIMPULAN

Dari proses perancangan sistem hingga pengujian aplikasi yang telah dilakukan, dapat ditarik kesimpulan yaitu: dengan metode translasi, rotasi, dan skeletal animation, animasi pada objek 3D bisa dilakukan.

Jika dibandingkan dengan game 3D komersil sejenis yang sudah beredar di masyarakat, animasi dari proyek ini lebih kaku. Salah satunya pada animasi bergerak. Perpindahan unit pada animasi gerak unit pada beberapa game komersil tidak dilakukan dengan konstan, tetapi dengan memperhitungkan percepatan pula. Selain itu, game-game komersil juga memasukkan efek-efek pada aplikasinya, sedangkan pada aplikasi ini tidak. Bisa disimpulkan bahwa masih banyak ruang perkembangan pada aplikasi ini.

7. REFERENSI

- [1] Zimmerman, E. (2004). Narrative, Interactivity, Play, and Games. In N. Wardrip-Fruin, & P. Harrigan, First Person: New Media as Story, Performance and Game (pp. 154-164). Massachusetts: MIT Press.

- [2] Esposito, N. (2005). A Short and Simple Definition of What a Videogame Is.
- [3] Walker, M. H. (2002). *Strategy Gaming*. Retrieved Desember 28, 2008, from *GameSpy*:
<http://archive.gamespy.com/articles/february02/strategy1/index.shtm>
- [4] Saphiro, L., & Stockman, G. (2000). *Computer Vision*. Prentice Hall.
- [5] Feldman, A. (2000). *Designing Arcade Computer Game Graphics*. Wordware Publishing Inc.
- [6] Soriano, M. (2011). Skeletal Animation. Retrieved Januari 18, 2013, from
http://alumni.cs.ucr.edu/~sorianom/cs134_09win/lab5.htm
- [7] Goldberg, A. V., Harrelson, C., Haim, K., & Werneck, R. F. (n.d.). Efficient Point-to-Point Shortest Path Algorithm. Retrieved 5 25, 2013, from Computer Science Department at Princeton University:
<http://www.cs.princeton.edu/courses/archive/spr06/cos423/Handouts/EPP%20shortest%20path%20algorithms.pdf>
- [8] Bresenham, J. E. (1965). Algorithm for computer control of a digital plotter. *IBM Systems Journal* 4.