

PEMBUATAN SIMULASI ALL TERRAIN VEHICLE PADA DUNIA VIRTUAL

Andreas Sutantra

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra
Jl. Siwalankerto 121 – 131 Surabaya 60236

E-mail: andreas0502@yahoo.com

ABSTRAK

Pembuatan simulasi ini bertujuan untuk memudahkan seseorang belajar tahap awal bagaimana mengendalikan kendaraan khusus tersebut tanpa perlu menghadapi resiko kecelakaan yang besar, maka dibuatlah simulator *All Terrain Vehicle* dalam dunia virtual ini. Simulasi kendaraan darat / mobil *offroad* (*All Terrain Vehicle*) yang dibuat memiliki beberapa fitur sederhana seperti berjalan pada medan berbatu dan bergunung – gunung (naik – turun). Kontrol pada *software* simulasi kendaraan darat ini berupa sebuah *game steering wheel* dan / atau *keyboard*. Game Simulator ini dibuat dengan menggunakan PhysX untuk memudahkan memasukkan kondisi fisik yang ada pada mobil di dunia nyata ke dalam dunia virtual. Untuk membuat inputan dari Logitech steering wheel G27, dibutuhkan Logitech SDK yang bisa didapatkan dari Logitech Gaming Software. Hasil dari skripsi ini adalah data berupa posisi mobil dengan koordinat x,y,z, dan kecepatan mobil yang nantinya akan digunakan untuk menggerakkan platform MA³I. Output ini dipengaruhi oleh beberapa input dari user yang berupa gas, rem, dan *steering wheel*.

Kata Kunci: Game Simulator, Logitech SDK, PhysX

ABSTRACT

The making of this simulation is to facilitate someone studying the early stages of how to control the special vehicles without need to face the risk of an accident. The simulation of land vehicles / offroad car (All Terrain Vehicle) made, having some simple features like walks in rocky and bumpy road. The software use Logitech Steering Wheel or Keyboard as the controller. This simulator game is made using PhysX to make it easier when incorporating physics in the real world into the virtual world. Logitech SDK, derived from Logitech Gaming Software, is needed for data input from Logitech steering wheel. The result of this thesis is the data such as car position with coordinates x, y, z, and the speed of the car that will be used to control the MA³I platform. This output is affected by some input from the user in the form of gas, brake, and steering wheel.

Keywords: Game Simulator, Logitech SDK, PhysX

1. PENDAHULUAN

Mengendarai kendaraan di medan yang tidak rata, berbatu – batu, dan berlumpur (*offroad*) tidaklah mudah. Dibutuhkan kendaraan khusus untuk bisa melewati medan yang sulit dan berbatu – batu. Selain dibutuhkan kendaraan khusus, dibutuhkan juga keterampilan khusus bagi pengemudi untuk dapat menjalankan kendaraan tersebut di medan *offroad*. Untuk memudahkan seseorang belajar tahap awal bagaimana mengendalikan kendaraan khusus tersebut, maka dibuatlah simulator *All Terrain Vehicle* dalam dunia virtual.

Saat ini sedang berjalan sebuah penelitian tentang implementasi *software Neuro-Fuzzy* sebagai sistem pengambilan keputusan pada platform MA³I (*Massively Actuator Array using Artificial Intelligence*)^[1]. *Software* tersebut akan di desain dan dikoneksikan dengan input *Keyboard* atau *Steer* yang berfungsi sebagai inputan dari algoritma *Neuro-Fuzzy*. Sebelum alat tersebut dapat dijalankan, maka dibutuhkan simulasi kendaraan darat / mobil *offroad* (*All Terrain Vehicle*) yang dapat menghasilkan data *training* berupa posisi koordinat x, y, z, dan kecepatan mobil untuk *Neuro-Fuzzy* yang akan digunakan untuk mengontrol MA³I. Hasil utama dari penelitian ini adalah desain *software* yang sesuai dengan inputan *real* (*Keyboard/Steer*) dan dapat dijadikan acuan untuk melakukan penelitian tahap ketiga, yaitu implementasi *Motion Simulator* berbasis *Artificial Intelligence*. Adapun *input* dari *user* berupa gas, rem, dan *steering wheel*

2. LANDASAN TEORI

2.1. SIMULASI

Menurut *Oxford American Dictionary* (1980), simulasi adalah untuk menghasilkan suatu kondisi dari sebuah situasi tertentu, dengan menggunakan model, untuk belajar, mencoba, atau latihan^[2]. Selain itu simulasi juga dapat didefinisikan sebagai teknik menggunakan komputer untuk meniru atau mengimitasi suatu operasi dari berbagai fasilitas atau proses dari dunia nyata^[3]. Definisi lain menyebutkan, simulasi berhubungan dengan permodelan dari suatu proses atau sistem dalam suatu cara tertentu sehingga model tersebut menirukan respon dari sistem actual terhadap suatu kejadian yang terjadi seturut dengan waktu^[4].

2.2. VIDEO GAME

Menurut Jon Cogburn, *video game* adalah bagian dari dunia nyata.^[5] Sedangkan teori lain menyebutkan video game adalah

permainan elektronik yang melibatkan interaksi Antara manusia dengan *user interface* untuk menghasilkan *visual feedback* pada *video device*.^[6] *Video* dalam *video game* berarti segala bentuk alat yang bisa menghasilkan gambar 2 atau 3 dimensi.

2.3. GAME SIMULASI

Game simulasi adalah *game* yang berusaha untuk memasukkan berbagai aktifitas dalam dunia nyata dalam bentuk permainan, untuk berbagai tujuan, seperti pelatihan, analisa, atau prediksi. Biasanya dalam *game* ini tidak ada tujuan yang jelas. Dan pemain biasanya bisa menggerakkan tokoh di dalam game dengan bebas. Beberapa contoh jenis *game* simulasi antara lain *role-play simulation*, *business games*, dan lain – lain.^[7]

2.4. GAS

Pedal gas adalah salah satu komponen mesin yang bertugas untuk memberikan pemasukan bensin ke dalam silinder mesin. Banyaknya bensin yang dimasukkan ke dalam silinder mesin berguna untuk mempercepat putaran mesin sehingga dapat mempercepat laju kendaraan. Mekanisme pedal gas memerlukan penekanan dari pengemudi. Percepatan yang terjadi pada mobil dipengaruhi oleh pedal gas yang diinjak oleh pengemudi. Semakin dalam pedal gas diinjak, maka percepatan yang terjadi akan semakin besar, dan demikian juga sebaliknya.^[8]

2.5. REM

Rem merupakan suatu peranti untuk memperlambat atau menghentikan gerakan roda pada kendaraan.^[8] Sistem rem pada kendaraan merupakan suatu peranti penting keamanan dalam berkendara, tidak berfungsinya rem dapat menimbulkan bahaya dan keamanan berkendara jadi terganggu. Adapun fungsi dari sistem rem itu sendiri adalah :

1. Untuk memperlambat kecepatan atau menghentikan gerakan roda kendaraan.
2. Mengatur kecepatan selama berkendara.
3. Untuk menahan kendaraan saat parkir dan berhenti pada jalan yang menurun atau menanjak.

Prinsip kerja sistem rem adalah mengubah tenaga kinetik menjadi panas dengan cara menggesekan dua buah logam pada benda yang berputar sehingga putarannya akan melambat. Oleh sebab itu komponen rem yang bergesekan ini harus terbuat dari bahan yang tahan terhadap gesekan (tidak mudah aus), tahan panas dan tidak mudah berubah bentuk pada saat bekerja dalam suhu tinggi.

2.6. KOPLING

Kopling atau clutch yaitu peralatan transmisi yang menghubungkan poros engkol dengan poros roda gigi transmisi. Kopling suatu perangkat/sistem yang merupakan bagian dari sistem pemindah. Fungsi kopling adalah untuk memindahkan, memutus dan menghubungkan putaran tenaga mesin ke transmisi, kemudian transmisi mengubah tingkat kecepatan sesuai yang diinginkan dengan lembut dan cepat. Pada bidang otomotif, kopling digunakan untuk memindahkan tenaga motor ke unit transmisi. Dengan menggunakan kopling, pemindahan

gigi-gigi transmisi dapat dilakukan, kopling juga memungkinkan motor juga dapat berputar walaupun tidak dalam posisi netral.

2.7. Steering Wheel

Sistem kemudi pada kendaraan mobil berfungsi untuk mengatur arah laju kendaraan dengan cara membelokkan roda depan. Bila roda kemudi diputar, steering column meneruskan putaran ke roda gigi kemudi. Roda gigi kemudi ini memperbesar momen putar, sehingga menghasilkan tenaga yang lebih besar untuk menggerakkan roda depan melalui sambungan-sambungan kemudi.

Sistem kemudi adalah salah satu sistem pada chassis mobil yang berfungsi untuk merubah arah kendaraan dan laju kendaraan dengan cara menggerakkan atau membelokkan roda-roda depan mobil dan menjaga agar posisi mobil tetap stabil. Cara kerjanya adalah, apabila roda roda kemudi (*steering wheel*) di gerakkan/diputar, kolom kemudi (*steering column*) kemudian meneruskan putaran ke putaran ke roda gigi kemudi (*steering gear*). *Steering gear* ini berfungsi untuk memperbesar momen putar, sehingga menghasilkan tenaga yang lebih besar untuk menggerakkan roda depan melalui sambungan-sambungan kemudi (*steering linkage*).

2.8. PHYSX (Physics Engine SDK)

PhysX SDK^[9] adalah sebuah *physics engine* SDK yang digunakan di dalam sebuah *game*. Dulunya *PhysX* berasal dari sebuah mesin simulasi fisika yang dikenal dengan sebutan *NovodeX*. *NovodeX* sendiri dikembangkan oleh perusahaan Swiss yang bernama *NovodeX AG*. Namun pada tahun 2004, *Ageia* membeli *NovodeX AG* dan mulai mengembangkan *hardware* yang bisa mengakselerasi kalkulasi fisika, dan diberi nama *PhysX PPU (Physics Processing Unit)*. Setelah itu nama SDK-nya diganti dari *NovodeX* menjadi *PhysX*.

Video games yang di dukung oleh akselerasi *hardware* dari *PhysX*, seharusnya bisa menghasilkan tampilan yang lebih halus dan bisa menimbulkan beberapa efek tambahan. *PhysX* sangat berperan penting bagi kebanyakan pembuat game. Hal ini dikarenakan para pembuat game tidak perlu lagi menuliskan sendiri rumus – rumus fisika yang ingin digunakan di dalam game nya, mengingat interaksi fisika yang sangat kompleks di dalam game modern. Sebagai buktinya, *PhysX engine* dan SDK-nya bisa digunakan untuk beberapa platform seperti *Windows*, *Mac*, *Linux*, *PlayStation 3*, *Xbox360* dan *Nintendo Wii*.

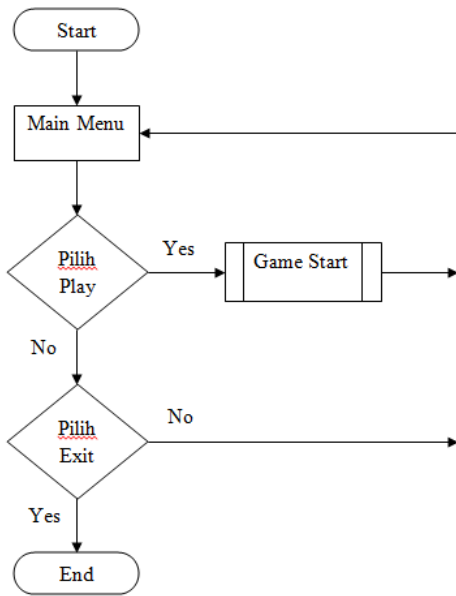
Meskipun desain utama *PhysX SDK* untuk *game developers*, *PhysX SDK* juga banyak digunakan untuk penelitian, pendidikan, dan developer aplikasi simulasi yang membutuhkan *real time performance*. Fitur – fitur yang ada dalam *PhysX SDK* ini antara lain *continuous collision detection*, *raycasting* dan *shape sweeps*, *solvers for rigid body dynamics*, *fluids*, *particles*, *vehicle* dan *character controllers*. Versi *PhysX SDK* yang paling baru ialah *PhysX-3.3.0* dan sudah bisa melakukan *rendering* dengan menggunakan *Microsoft DirectX*. Hal ini sangat menguntungkan bagi *programmer* mengingat versi – versi yang sebelumnya dari *PhysX SDK* belum bisa melakukan *rendering*. Dalam menggunakan *PhysX SDK* sendiri tidaklah mudah. Hal ini dikarenakan *programmer* harus menyesuaikan bahasa dan gaya pemrograman-nya dengan gaya dan bahasa pemrograman

yang digunakan PhysX SDK. Namun, fisika yang dibutuhkan di dalam game sudah disediakan oleh PhysX SDK.

3. DESAIN SISTEM SIMULASI

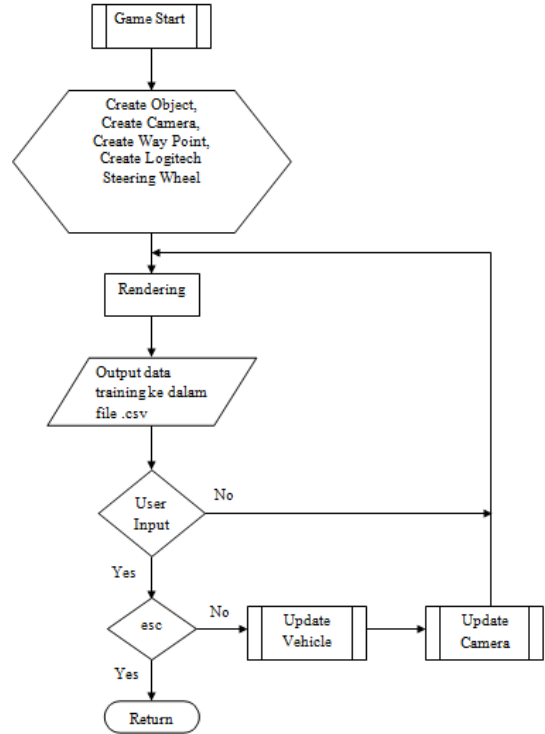
Dalam aplikasi yang akan dibuat, terdapat beberapa tahapan yang harus dilakukan. Pada bagian ini dibuat *flowchart* untuk memperjelas tahapan atau proses-proses yang dilakukan pada program.

Pada Gambar 1 dijelaskan alur program pada saat *user* pertama kali menjalankan aplikasi. Saat aplikasi pertama dijalankan, yang ditampilkan paling awal adalah halaman main menu. Di halaman ini, *user* dihadapkan pada 2 pilihan menu yaitu play atau exit. Apabila *user* memilih menu play, maka aplikasi akan menjalankan prosedur berikutnya yaitu Game Start. Untuk prosedur game start akan dijelaskan pada Gambar 2. Apabila *user* memilih exit, maka *user* akan langsung keluar dari aplikasi.

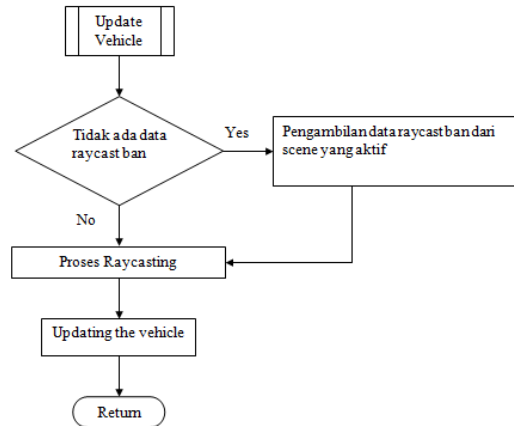


Gambar 1 Flowchart Menu Utama

Pada Gambar 2 dijelaskan alur program pada saat *user* memilih menu *game start* pada menu utama. Yang akan dilakukan oleh program pertama kali adalah membuat mobil, *object*, *camera*, *way point*, *logitech steering wheel*. Setelah semua dibuat, maka program akan melakukan proses *rendering* dimana selama proses *rendering*, semua *image* akan di *generate* menjadi sebuah *object* 3D. Setelah proses *rendering*, maka data yang diperlukan seperti posisi dan sudut elevasi mobil, kecepatan, posisi roda, akan dituliskan ke dalam sebuah file dengan tipe *csv*. Kemudian, akan ada pengecekan terhadap *user input*. Apabila *user* melakukan inputan dari *steering wheel / keyboard*, maka akan dilakukan *update* terhadap posisi mobil dan posisi kamera. Apabila *user* menekan tombol *esc* pada *keyboard*, maka akan kembali ke menu utama.

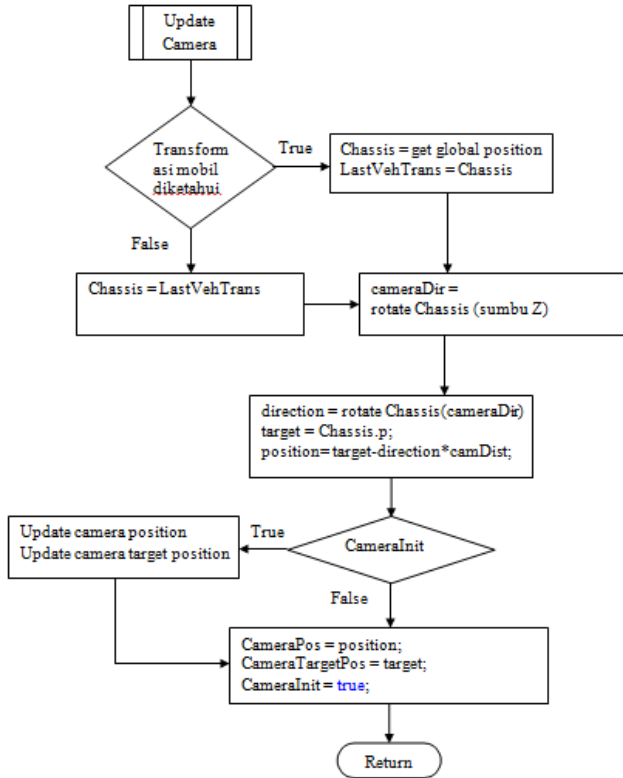


Gambar 2 Flowchart Game Start



Gambar 3 Flowchart Update Vehicle

Pada Gambar 3 dijelaskan alur program pada saat fungsi *update vehicle* dipanggil. Yang akan dilakukan oleh program pertama kali adalah melakukan pengecekan terhadap *raycasting* dari ban mobil, apakah kosong atau tidak. *Raycasting* sendiri adalah proses pengecekan interseksi antara 2 *object*. Apabila kosong, maka akan di isi terlebih dahulu. Setelah itu, akan dipanggil fungsi *PxVehicleSuspension Raycasts* untuk pengecekan *raycasting* roda terhadap tanah yang dilewati. Dan yang terakhir akan dipanggil fungsi *PxVehicleUpdates* untuk *update* posisi mobil.



Gambar 4 Flowchart Update Camera

Pada Gambar 4 dijelaskan alur program pada saat fungsi *update vehicle* dipanggil. Yang akan dilakukan oleh program pertama kali adalah melakukan pengecekan apakah *camera* sudah tahu keberadaan mobil user. Apabila belum, maka akan dicari di mana posisi mobil terakhir. Setelah posisi mobil diketahui, langkah selanjutnya ialah mengatur posisi *camera* dan arah hadap *camera*. Untuk menentukan posisi *camera* dapat dilakukan pengurangan antara *direction* dan *target*, kemudian dikalikan dengan jarak kamera. Sedangkan untuk target dapat mengambil dari titik depan *chassis* mobil menggunakan *chassis.trfm.p*. Setelah menemukan *position* dan *target*, maka dapat dilakukan update *position* dan *target* dengan menggunakan fungsi *dampVec3*.

4. HASIL SIMULASI

Pada bab ini dilakukan uji coba pada program mengenai fitur-fitur yang disediakan. Uji coba dilakukan untuk memastikan aplikasi dapat berjalan dengan baik dan tanpa *error*.

Dalam pengujian Steering Wheel, dapat dilihat bahwa ketika *steering wheel* digerakkan ke kiri, posisi mobil dapat berbelok ke kiri juga, diikuti dengan posisi kamera yang juga berbelok ke kiri. Hal ini menunjukkan bahwa *steering wheel* berfungsi dengan baik, begitu juga dengan kamera. Dapat dilihat pada Gambar 5 sebagai berikut :



Gambar 5 Pengujian Steering Wheel

Pengujian juga dilakukan user menambah kecepatan. Ketika *user* menekan tombol W pada keyboard atau menginjak pedal gas pada steering wheel, maka kecepatan pada mobil akan bertambah. Kecepatan mobil ditunjukkan oleh jarum merah. Seperti yang terlihat pada Gambar 6, meskipun mobil sudah berada pada kecepatan maksimum, namun gigi persneling tetap menunjukkan angka 1. Hal ini dikarenakan mobil menggunakan transmisi manual. Dengan adanya perubahan posisi mobil, diikuti dengan perubahan posisi kamera, maka tidak ada masalah dengan pengujian kamera untuk gerak maju.



Gambar 6 Pengujian Kecepatan Mobil

5. KESIMPULAN

Setelah membuat aplikasi simulator mobil *offroad* ini dapat diambil beberapa kesimpulan yaitu:

- Dengan adanya aplikasi simulator ini, *user* dapat merasakan bagaimana mengendarai mobil di tanah yang tidak rata.
- *User* dapat mempelajari cara mengemudikan mobil di tanah *offroad* dengan mengetahui komposisi *gear* yang pas.

- Aplikasi akan mengeluarkan data yang akan digunakan pada *platform* MA³I di luar tugas akhir ini.

REFERENSI

- [1] Pasila, F., 2012. Control of Massive Parallel Arrays of Three-State Actuators via Artificial Inteligent
- [2] Ehrlich, E., 1980. *Oxford American Dictionary*. Michigan: Oxford University Press.
- [3] Kelton, A. M. L. & W. D., 2000. *Simulation Modeling and Analysis*. 3rd ed. s.l.:McGraw-Hill.
- [4] Schriber, T. (1987). The nature and role of simulation in the design of manufacturing systems. *Simulation in CIM and artificial intelligence techniques*. Society for Computer Simulation, 5–18.
- [5] Jon Cogburn, 2009, *Philosophy Through Video Games*, Routledge, New York
- [6] Travinor, Grant, 2009, *The Art of Video Games*, Willey-Blackwell, United Kingdom
- [7] Aldrich, C., 2009. *The Complete Guide to Simulations and Serious Games*. s.l.:Pfeiffer
- [8] Hutahaeen, R. Y., 2010. *Mekanisme & Dinamika Mesin*. Yogyakarta: Andi.
- [9] Kumar, K., 2013. *Learning Physics Modeling with PhysX*. s.l.:Packt Publishing