

Dokumentacja symulatora pojazdu

Michał Białożyty, Dawid Kopec

Czerwiec 2024

1 Wstęp

W dokumencie zawarto dokumentację gry komputerowej, której celem jest symulacja jazdy samochodem z naciskiem na realistyczne odwzorowanie fizyki pojazdu. Nasz symulator powstał z myślą o zapewnieniu graczom jak najwierniejszego doświadczenia prowadzenia samochodu, uwzględniając wszelkie aspekty fizyczne, takie jak dynamika pojazdu oraz interakcje z otoczeniem. Całość gry powstała z użyciem frameworka Unity.

2 Funkcjonalności

Nasza gra oferuje szereg zaawansowanych funkcji, które składają się na realistyczny symulator jazdy samochodem. Podstawowe aspekty rozgrywki obejmują płynne i precyzyjne poruszanie się pojazdu, realistyczne skręcanie oraz efektywne hamowanie, które uwzględnia dynamikę i masę samochodu. Dodatkowo, symulacja zawieszenia pozwala na wierne odwzorowanie reakcji pojazdu na różne typy nawierzchni i nierówności drogi, co zwiększa realizm doświadczenia. W grze zaimplementowano również zaawansowaną symulację zderzeń, która dokładnie odzwierciedla konsekwencje kolizji i interakcje z otoczeniem. Prędkościomierz dostarcza graczowi bieżących informacji o prędkości jazdy, co jest kluczowe dla kontrolowania pojazdu i utrzymania bezpieczeństwa. Modele samochodów oraz mapy użyte w grze zostały pobrane jako gotowe pakiety, co pozwoliło na skoncentrowanie się na optymalizacji mechanik gry i dostarczenie graczom najwyższej jakości wizualnej oraz technicznej. Zaimplementowano również kilka pozycji kamer aby użytkownik sam mógł wybrać tryb w którym będzie sterował samochodem.

3 Realizacja fizyki

Wiedzę odnośnie fizyki w grach zaczerpnięto z książki pt. "Game Physics" autorstwa Davida H. Eberly'ego. Większość funkcjonalności osiągnięto jednak głównie przy pomocy dostępnych w silniku unity rozwiązań. Silnik Unity realizuje fizykę za pomocą wbudowanego systemu fizyki, który korzysta z bibliotek takich jak PhysX od NVIDIA. Oto kluczowe elementy i procesy realizacji fizyki w Unity:

3.1 Komponenty fizyki

Unity wykorzystuje kilka podstawowych komponentów do obsługi fizyki:

- **Rigidbody:** Dodanie tego komponentu do obiektu powoduje, że staje się on dynamicznym ciałem fizycznym. Oznacza to, że obiekt będzie reagował na siły i kolizje.
- **Colliders:** Te komponenty definiują kształt obiektu fizycznego. Mogą to być proste kształty, takie jak sfera (SphereCollider), kapsuła (CapsuleCollider), czy pudełko (BoxCollider), jak również bardziej złożone kształty (MeshCollider).
- **Joints:** Służą do łączenia dwóch ciał fizycznych, umożliwiając realizację bardziej skomplikowanych interakcji, takich jak zawiasy (HingeJoint) czy przeguby kulowe (BallJoint).

3.2 Symulacja fizyki

Symulacja fizyki w Unity jest realizowana na zasadzie kroków czasowych (time steps):

- **Fixed Timestep:** Symulacja fizyki odbywa się w ustalonych odstępach czasu, które można konfigurować w ustawieniach projektu. Domyślnie jest to 0.02 sekundy (50 Hz).
- **Integracja sił i momentów:** Podczas każdego kroku symulacji, siły i momenty (torques) są integrowane, aby obliczyć nową prędkość i pozycję obiektów.
- **Kolizje i kontakt:** Silnik fizyki wykrywa kolizje między obiektami i oblicza odpowiednie reakcje. Używa do tego algorytmów detekcji kolizji oraz rozwiązywania kontaktów, aby symulować zderzenia i oddziaływania.

3.3 Siły i ruch

Rigidbody w Unity pozwala na aplikowanie różnych rodzajów sił:

- **Force:** Stała siła aplikowana w danym kierunku.
- **Impulse:** Natychmiastowa siła działająca przez krótki czas, używana do symulacji gwałtownych zdarzeń, takich jak wystrzał.
- **Acceleration:** Zmiana prędkości obiektu.
- **Velocity Change:** Natychmiastowa zmiana prędkości.

3.4 Kolizje i detekcja kolizji

Unity zapewnia różne metody detekcji kolizji:

- **OnCollisionEnter, OnCollisionStay, OnCollisionExit:** Metody te są wywoływane, gdy collider obiektu zderza się z innym colliderem.
- **OnTriggerEnter, OnTriggerStay, OnTriggerExit:** Podobnie jak powyższe, ale dla obiektów z ustawioną opcją "Is Trigger".

3.5 Ustawienia fizyki

Unity pozwala na konfigurowanie różnych ustawień fizyki w panelu "Physics" w ustawieniach projektu. Można tam ustawić takie parametry jak:

- **Gravity:** Globalna grawitacja działająca na wszystkie obiekty Rigidbody.
- **Solver Iterations:** Liczba iteracji solvera, który rozwiązuje siły i kolizje. Więcej iteracji zwiększa dokładność, ale obniża wydajność.
- **Collision Detection Mode:** Tryb detekcji kolizji (Discrete, Continuous, Continuous Speculative) wpływający na dokładność i wydajność symulacji kolizji.

3.6 Optymalizacja

Aby symulacja fizyki była efektywna i wydajna, Unity stosuje różne techniki optymalizacyjne:

- **Sleep Mode:** Obiekty Rigidbody mogą "zasypiać" (przestają być aktywnie symulowane), jeśli ich prędkość i moment są poniżej pewnych progów.
- **Layer-based Collision Detection:** Możliwość wyłączenia detekcji kolizji między obiektami znajdującymi się na określonych warstwach (layers).

4 Podsumowanie

Projekt symulatora samochodu stworzony w silniku Unity łączy realizm fizyki pojazdów z przyjemnym doświadczeniem dla graczy. Gra ta ma na celu dostarczenie realistycznych doznań z jazdy samochodem, łącząc szczegółową grafikę, precyzyjny model fizyczny i interaktywne elementy. Dzięki zaawansowanym narzędziom opisanym w rozdziale 3 udało się zaimplementować funkcjonalności związane ze sterowaniem oraz zachowaniem pojazdu.