

# Knockout Racing: Joc de curses amb Intel·ligència artificial

Lucía Aparicio Romero

**Resum**—La intel·ligència artificial (IA) és una de les parts més importants que caracteritzen el gènere de curses en videojocs. Aquesta IA ha de calcular la posició dels cotxes contrincants en el circuit d'una forma realista. Aquest gènere es caracteritza per tenir un conjunt de vehicles que tenen com a objectiu arribar primers a la meta després d'haver donat un número determinat de voltes al circuit. Els vehicles adversaris son agents intel·ligents amb detectors d'obstacles que segueixen el tram del circuit i actuen segons les accions dels altres cotxes i del propi caràcter de la IA.

En aquest projecte s'ha realitzat el disseny, art i gameplay d'un videojoc 3D de curses de cotxes amb intel·ligència artificial controlat amb Leap Motion.

**Paraules clau**—Intel·ligència artificial en videojocs, contrincants, Leap Motion, videojocs 3D, art 3D en videojocs, gameplay, simulador de conducció.

**Abstract**—Artificial intelligence (AI) is one of the most important features that characterize the genre of racing games. This IA has to calculate the position of the car rivals on the circuit. The genre consists in a set of vehicles that aim to reach the finish line first after giving a certain number of laps. The racing car genre is characterized for having a group of vehicles that have to reach the finish line first giving a certain number of laps. The opponent vehicles have an artificial intelligence that allows them to drive on the circuit and act according to the actions of the other cars and the nature of the AI.

In this article is explained the design, art and gameplay a 3D car racing game with artificial intelligence controlled by Leap Motion

**Index Terms**—Artificial intelligence in video games, opponents, Leap Motion, 3D video games, 3D art video games, gameplay, driving simulator.



## 1 INTRODUCCIÓ

Dins del món dels videojocs, el gènere de curses de cotxes es caracteritza per controlar un conjunt de vehicles que tenen com a objectiu arribar primers a la meta després d'haver donat un número concret de voltes a un circuit. Els cotxes que no controla el jugador han de ser dirigits mitjançant una IA. El comportament d'aquesta IA, basat en els agents intel·ligents, pot variar entre cadascun dels cotxes per donar-los diferents caràcters. Un agent intel·ligent és una entitat que percep els seu entorn a través de sensors i actua de manera autònoma.

Dos dels jocs més significatius dins d'aquest gènere son principalment Mario Kart [7] i Trackmania Original [8]. Mentre Mario Kart es centra en la utilització d'objectes per entorpir als contrincants i la possibilitat de controlar el cotxe amb volant i Trackmania Original opta per donar una alta llibertat i varietat als seus jugadors permetent que personalitzin el disseny dels seus cotxes i creïn els seus propis circuits o descarregar circuits creats per la comunitat, fent que la IA s'adapti a qualsevol circuit creat.

Sempre m'han agradat els jocs de conducció, pel que s'ha volgut crear un videojoc propi d'aquest gènere, amb una IA pròpia i la utilització del Leap Motion com a control.

Un d'aquests aspectes diferenciadors és la utilització de Leap Motion [2]. Aquest dispositiu USB consta de dues

càmeres monocromàtiques i tres LED infrarojos amb els que detecta tots els gestos que es fan amb les mans a una distància aproximada d'un metre.

Tenint en compte tot això, l'objectiu principal és el disseny, implementació i test d'un videojoc de gènere de conducció amb les funcionalitats i característiques següents:

- Implementació de la IA pels contrincants basada en agents intel·ligents.
- Generació dels menús, escenaris i HUD del joc.
- Moviment del cotxe pels diferents espais creats mitjançant Leap Motion i teclat.
- Implementació del sistema de curses, comptabilitzant les voltes i el temps transcorreguts.

Per al disseny i la implementació del projecte, s'han utilitzat diverses eines:

- **Unity 5** [1]. Motor per a la creació de videojocs multiplataforma compatible amb la tecnologia Leap Motion.
- **MonoDevelop** [9]. Entorn de programació integrat i lliure, dissenyat principalment per a C#, Java i Python.

- **Photoshop CS** [10]. Programa d'edició d'imatges utilitzat per al disseny dels botons, logo i textures dels objectes.

El document està dividit en diferents seccions. A la segona secció expliquem el Game Concept amb les idees bàsiques del videojoc. A la tercera, es mostra el desenvolupament del projecte, compost per la part d'art i gameplay. A la quarta, expliquem els resultats visuals i de test obtinguts. A la cinquena, expliquem les conclusions i millores del projecte.

## 2 GAME CONCEPT

El projecte consisteix en un joc monojugador en 3D de curses de cotxes amb vista en tercera persona controlat amb Leap Motion. Hi ha diversos escenaris amb diferents propòsits dins del joc. Els objectius del jugador variaran segons el mode de joc que triem (arribar primer a la línia de meta, realitzar el circuit en el mínim temps possible, etc.).

Els oponents tenen una IA amb paràmetres diferents entre cada cotxe i no modificables per l'usuari, fet que fa que cada cotxe actuï de forma diferent. Mentre un cotxe es mostra passiu a les accions de l'usuari, l'altre intenta entorpir-li la conducció. A continuació s'explica més detalladament cadascun dels apartats.

### 2.1 Modes de joc

Els modes del joc depenen segons l'escenari que triïs:

- **Contrarellotge.** L'objectiu consisteix en fer voltes al circuit en el mínim temps possible.
- **Cursa.** L'objectiu consisteix en arribar primer a la línia de meta després de donar un número determinat de voltes al circuit.
- **Terreny de proves.** Escenari amb obstacles on es pot provar el funcionament del cotxe amb Leap Motion amb més calma.

### 2.2 Controls

Per a la conducció del cotxe hi ha dues opcions a utilitzar: teclat o Leap Motion. En qualsevol moment es pot canviar entre un dispositiu o un altre.

El cotxe es pot moure amb teclat i/o amb el dispositiu Leap Motion.

El control per teclat és:

- Fletxa amunt o 'W': Accelerar
- Fletxa avall o 'S': Frenar/Marxa enrere
- Fletxa esquerra o 'A': Girar cap a l'esquerra
- Fletxa dreta o 'D': Girar cap a la dreta.
- Escape: Menú de pausa.
- Control: Reajustament a la posició del cotxe per defecte.

El control amb Leap Motion és:

- Rotació de la mà cap endavant: Accelerar. (Figura 1.a)
- Rotació de la mà cap enrere: Frenar/Marxa enrere. (Figura 1.b)
- Rotació de la mà cap a l'esquerra: Girar cap a l'esquerra. (Figura 1.c)
- Rotació de la mà cap a la dreta: Girar cap a la dreta. (Figura 1.d)

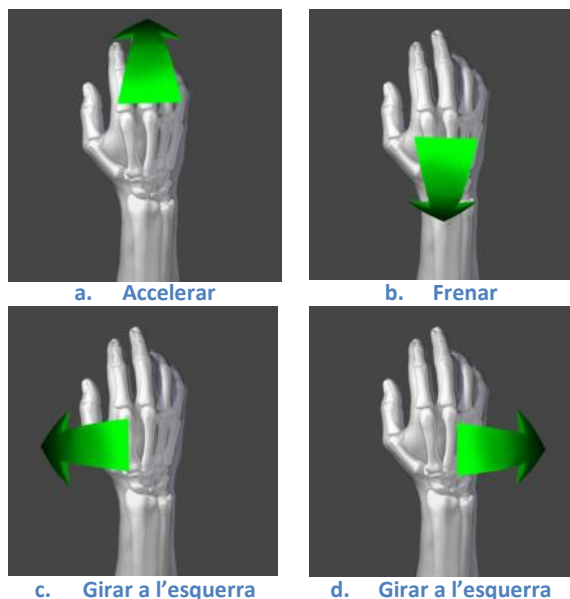


Figura 1. Control amb Leap Motion.

A més, s'utilitza el ratolí per a la navegació pels menús.

### 2.2 Art

Per a cadascun dels modes de joc es vol un circuit senzill de realitzar per a la IA. Tot i així, es vol que tingui una mica de dificultat a certs punts de l'escena, com poden ser corbes tancades o pujades i baixades pronunciades.

Es volen realitzar un total de dues escenes. Una de les quals s'utilitzarà per realitzar els diferents modes de joc. L'altra serà una escena lliure amb diferents obstacles que servirà per provar els controls del cotxe tant per teclat com per Leap Motion.

En quant als cotxes, s'utilitzarà un mateix model de cotxe i l'únic que es canviarà serà el color del cotxe per diferenciar-los entre ells. L'estil serà simple però modern.

### 2.2 Música

Per a cadascuna de les escenes es vol una música diferent per transmetre diferents sensacions.

Per al menú inicial es vol utilitzar una cançó que comenci tranquil·la però passat un temps tingui més força. D'aquesta manera el jugador es pot adonar que es troba al menú inici.

Per als menús de selecció es vol un so més tranquil, ja que triar entre un dels escenaris s'ha de fer amb tranquil·litat per no equivocar-te.

Per a tots els escenaris de cursa es vol una cançó més dinàmica i moguda per donar sensació de pressa. Per últim, per a l'escenari de proves es vol utilitzar la mateixa cançó que al menú inici. D'aquesta forma la cançó serà identificada pel jugador i l'ambient serà més familiar.

### 3 DESENVOLUPAMENT

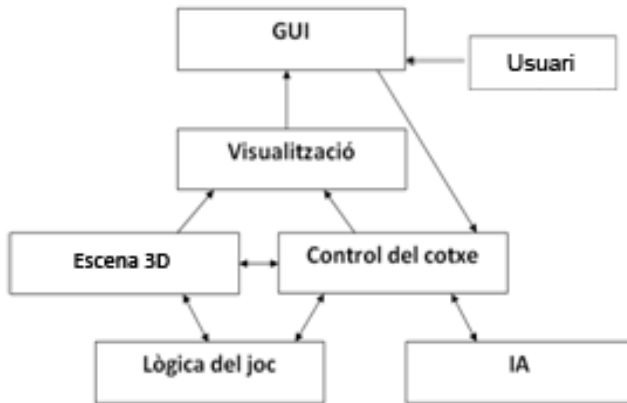


Figura 2. Diagrama de mòduls.

Hem dividit el desenvolupament en dos apartats: art i gameplay.

#### 3.1 Art

S'han dissenyat diferents models i textures per les escenes i objectes del joc:

- **Cotxes.** Tenim tres tipus diferents de cotxes amb diferents textures per diferenciar-los:
  - Disseny pel jugador, de color negre: figura 3.
  - Disseny per a les dues IA, de color verd i vermell: figura 4.



Figura 3. Cotxe del jugador.

S'ha dissenyat una textura de color verd pel cotxe de la primera IA ja que el seu caràcter és tranquil. En canvi, per a la segona IA s'ha utilitzat el color vermell per mostrar més agressivitat.

Per al jugador s'ha optat per un color fosc, ja que dóna un aspecte elegant i neutre.



Figura 4. Cotxes controlats per la IA.

- **Escenaris.** Per a les escenes del circuit i del terreny de proves s'han dissenyat diverses textures, ja que contenien objectes als que li faltava una textura, i per tant es visualitzaven de color gris (figures 5 i 6).

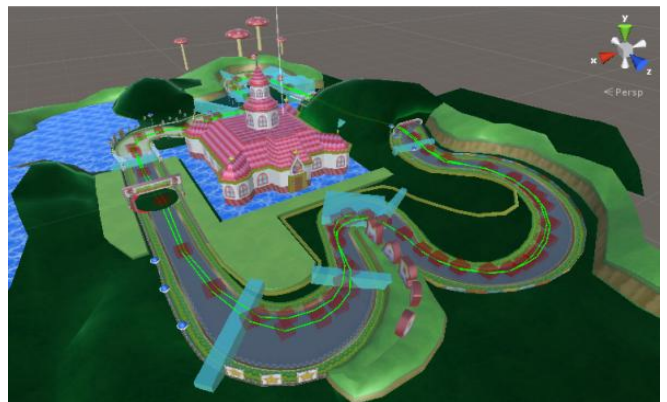


Figura 5. Circuit de Mario.

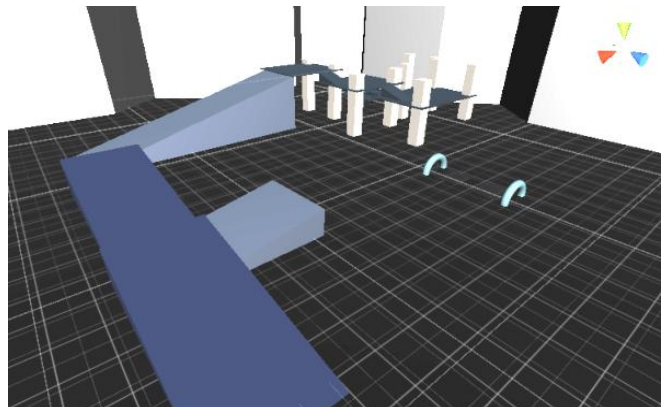


Figura 6. Terreny de Proves.

### 3.2 Gameplay

Amb el diagrama de mòduls de la figura 2 dividim les funcionalitats, per així treballar per parts i després connectar-les entre si per arribar a un resultat conjunt final.

L'usuari envia inputs a través de la GUI amb els quals pot actuar amb el mòdul de control del cotxe.

Des del control del cotxe movem el vehicle a través de l'escena 3D. L'escena 3D accedeix a la lògica del joc per comptabilitzar les voltes i el temps transcorregut utilitzant la posició en escena del cotxe. Els vehicles oponents són controlats pel mòdul del control de cotxe que es connecta amb el mòdul d'IA, que calcula la ruta que ha de seguir.

D'aquesta manera les dades s'intercanvien entre els mòduls i tot es visualitza a través de la GUI mitjançant el mòdul de visualització. A continuació s'explica cada mòdul amb més detall.

#### 3.2.1 IA

Els cotxes oponents es mouen pel circuit gràcies al mòdul de la IA [3][5]. L'algorisme es basa en els agents intel·ligents, els quals tenen sensors per moure's pel seu entorn. L'algorisme està dividit en dues parts. Primer es calcula una ruta pel circuit. Aquesta ruta està prèviament establerta mitjançant uns punts col·locats pel circuit (figura 7). Amb aquests punts es calcula la posició del cotxe i es modifica aquesta posició tenint en compte les accions del jugador. El cotxe conté objectes dins seu que detecten la distància entre el cotxe jugador i el propi objecte i s'utilitzen per determinar si el jugador està dins del radi d'acció i, si així és, s'intenti tancar el pas. Aquests objectes els anomenem sensors. A continuació s'explicarà més detalladament cadascun dels passos.

La primera part s'encarrega de traçar una ruta a través de diversos punts que estan repartits pel mapa. Aquests punts segueixen el tram base que el cotxe ha de realitzar. A la figura 7 es poden veure aquests punts de color vermell repartits pel circuit.

Per fer això necessitem la posició i l'angle del cotxe actuals i la posició del següent punt de ruta. Amb aquestes dades podem saber l'angle necessari per a que el cotxe se situï recte en direcció cap al següent punt de ruta. Per tal de que el gir no sigui tan bruscat s'aplica un angle de gir màxim al cotxe, fent que aquest sigui més suau i natural. Per tal de que compti com que el cotxe ha passat per un punt de ruta hi ha un marge d'error permès. Aquest marge permet que si el cotxe passa a una distància determinada del punt, compti com a correcte i calculi el següent punt.

La segona part només s'aplica a un dels dos cotxes implementats (el cotxe vermell). En cas de que el cotxe del jugador estigui a una distància determinada, tanca el pas i entorpeix la conducció al jugador. Per fer això s'han definit tres sensors en total (figura 8).

El primer sensor detecta si el jugador està dins de l'àrea d'acció. Això es mesura mirant la distància entre el centre del cotxe oponent i el centre del cotxe del jugador. En cas de que el jugador estigui dins d'aquesta àrea tenim dos

sensors a les rodes davanteres del cotxe, les quals ens ajuden a determinar si el jugador es troba a l'esquerra o a la dreta comprovant la seva distància.

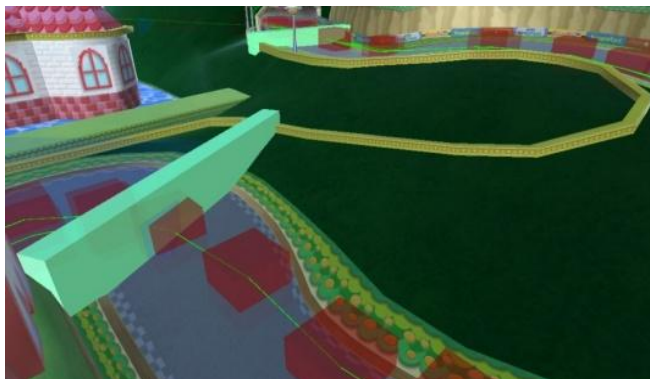


Figura 7. Punts de ruta i zones de velocitat.

Un cop se sap la posició del jugador, el cotxe controlat per la IA girarà en direcció cap al jugador, impedit que sigui avançat i entorpeix-lo. Aquest gir també té un angle de gir màxim i progressiu per que sigui més natural. A més, aquesta acció està limitada per poder realitzar-se només cada cinc segons, ja que volem prioritzar que el cotxe segueixi el tram del circuit, i no que només entorpeixi al jugador. Un cop el cotxe deixa d'entorpeix-lo, recalcula l'angle necessari per arribar al següent punt de ruta.



Figura 8. Sensors del cotxe.

S'ha hagut de limitar la velocitat del cotxe ja que sinó el vehicle arriba a velocitats en les quals derrapa i surt del circuit, ja que s'utilitzen les físiques que ens proporciona Unity. Es fa de dues formes. La primera és limitar la velocitat i acceleració màxima pròpia del cotxe. La segona és crear zones de velocitat adaptativa dins el circuit per fer que aquest s'adapti a cada situació (si es troba amb una corba massa tancada frenarà i si està en una recta accelerarà). A la figura 7 podem veure de color turquesa aquestes zones de velocitat. Si el cotxe topa amb una d'aquestes zones els seus paràmetres canvien segons les dades que continguin aquesta zona.



El cotxes no estan preparats per fer marxa enrere i es poden quedar atrapats en algun punt del circuit pel que s'ha implementat una solució. En cas de que el cotxe no es mogui del lloc durant més de cinc segons voldrà dir que s'ha quedat atrapat, pel que es girarà 90° per que pugui tornar a arrancar i continuar el circuit.

El mòdul està format pels següents scripts:

- **Simple Velocity Limiter.** Encarregat de limitar la velocitat màxima, la acceleració i el fre del cotxe.
- **AI Car\_Script.** Encarregat de traçar un camí entre cadascun dels punts de ruta, d'entorpir al jugador si escau i de detectar i actuar si el cotxe s'ha atrapat.
- **Brake Zone.** Encarregat d'alterar la velocitat del cotxe en una zona determinada. A la figura 7 podem veure dues zones de fre de color turquesa. Quan el cotxe impacta amb una d'aquestes zones els paràmetres d'acceleració i de fre son alterats.
- **Waypoint.** Encarregat d'unir cadascun dels punts de ruta per obtenir un circuit tancat. A les figures 5 i 7 podem veure un conjunt de punts units amb línies. Aquest circuit que es crea és utilitzat després per la IA del cotxe (*AI Car\_Script*) per calcular la ruta per on passarà el cotxe.

### 3.2.2 Control del cotxe

Aquest mòdul controla el cotxe del jugador i rep del mòdul d'IA la posició del cotxe. Hi ha dos tipus de cotxe: jugador i oponent (Intel·ligència artificial).

#### • Jugador

S'ha adaptat les llibreries que ens dona Unity per tal de moure el cotxe a les nostres necessitats a partir de la base que ens donen.

Conté diversos scripts per al seu control:

- **Car Controller.** Determina els paràmetres del cotxe. Els paràmetres més importants són la velocitat màxima, la acceleració, el fre, la velocitat màxima (marxa enrere) i l'angle màxim de les rodes. Ve donat per les llibreries de Unity, pel que s'ha hagut de modificar els paràmetres esmentats.
- **Car User Control.** Encarregat del control del cotxe. Assigna les tecles i els gestos interpretats amb Leap Motion de la mà a determinades accions (Secció 2.2). En el cas del Leap Motion, Unity incorpora llibreries que podem utilitzar per saber la posició i rotació de la mà. Amb aquests paràmetres s'ha associat la rotació en l'eix z per a l'acceleració i l'eix x per al gir. Aquesta rotació ve donada en percentatge, pel que si rotem la mà cap a l'esquerra o la dreta, el valor pot variar entre -1.0 i 1.0. El mateix passa amb el gir amb l'eix z. Aprofitem que ve donat amb percentatge per aplicar-ho a la acceleració i gir depenent dels valors màxims establerts.

Per comoditat s'ha regulat l'angle de la mà a tres nivells diferents per a l'acceleració i fre:

- Si la mà està rotada entre un 35% i un 100% el cotxe accelerarà al màxim.

- Del 25% al 35% accelerarà un 60% de la seva acceleració màxima.
- I del 10% al 25% un 30%.

El mateix passa amb el cas del fre (angles amb valors negatius).

S'ha afegit també la possibilitat de deixar el cotxe en punt mort, fent que ni acceleri ni freni. Això passa quan la ma es troba entre el -10% i 10% de rotació en l'eix z.

- **Car Audio.** Aplica un so determinat depenent de la velocitat actual o la quantitat de fre aplicat.
- **Car Self Righting.** En cas de que el cotxe hagi quedat cap avall, aquest script s'encarregarà de girar-lo automàticament després d'uns segons. Això es pot saber mitjançant el valor de la rotació en l'eix x del cotxe. També s'ha habilitat la opció de girar el cotxe manualment per teclat.

#### • Oponent (Intel·ligència artificial).

El cotxe és controlat pel mòdul d'Intel·ligència artificial explicat al punt 3.1.1.

### 3.2.3 Escena 3D

Conté l'escena, objectes 3D, cotxes i llums de l'escena en una posició, angle i escala concrets. Aquests objectes tenen una física i col·lisió pròpia.

S'han creat dos escenaris en total: el circuit de Mario i el terreny de proves.

El circuit de Mario (figura 5) té tres modes de joc en total: circuit, temps i demo.

El terreny de proves (figura 6) només té un escenari i no té altre propòsit que el de provar els controls del cotxe.

### 3.2.4 Lògica del joc

Aquest mòdul s'encarrega de les funcionalitats de la cursa, com són el cronòmetre i la comptabilització de voltes.

Per obtenir el valor dels minuts i els segons totals en cursa s'ha utilitzat una llibreria anomenada Time, que ens dona els segons que han passat des de que la escena s'ha carregat del tot.

Per obtenir el temps de cadascuna de les voltes s'ha creat una taula on guardem el temps parcial de cada volta. A cada volta li descomptem el temps de les voltes anteriors sobre el temps total, així obtenim el valor de la volta que acabem de fer.

Per comptar les voltes s'han creat quatre sectors, els quals es mostren a la figura 9.

Quan un cotxe passa a través d'un dels sectors, el valor del sector s'actualitza. Un cop es torna a passar pel primer sector, el valor de volta s'incrementa en un. D'aquesta manera només comptarà com a volta si hem realitzat el circuit en el sentit correcte passant per cadascun dels sectors.

Cada cop que un cotxe creua el sector inicial es comprova si s'han realitzat les tres voltes. D'aquesta manera, quan un cotxe ha realitzat totes les voltes s'incrementa en un la posició final del jugador.



Figura 9. Sectors del circuit.

### 3.2.5 Visualització

Conté tota la informació visual amb que l'usuari pot interactuar. Totes les escenes i interfícies han estat creades mitjançant scripts en comptes de les eines que ens proporciona Unity per a la creació de botons i altres elements de la interfície d'usuari, ja que d'aquesta manera és fàcilment editable i reaprofitable.

S'han implementat diferents escenes que canvien segons les opcions triades per l'usuari. La primera escena és la del menú d'inici on podem triar entre jugar o sortir al escriptori (figura 10).

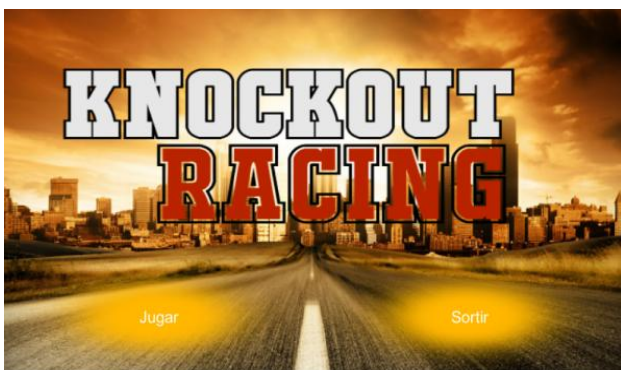


Figura 10. Menú principal.

La segona escena ens permet triar circuit. Podem triar entre dues escenes (circuit de Mario i terreny de proves) o tornar al menú inicial (figura 11).

En cas d'escollir el circuit de Mario, ens permet triar modalitat (demostració, contrarellotge o cursa).

Cadascuna de les diferents modalitats té un menú de pausa (figura 20 en annex) i un menú pel final de la cursa propi (figura 21 en annex), on mostra els resultats i deixa a l'usuari decidir què fer a continuació, ja sigui repetir la cursa, triar un altre o sortir al menú d'inici.

S'ha trobat necessari poder graduar la qualitat de vídeo dins del propi joc ja que no tots els ordinadors poden jugar amb màxima configuració gràfica. D'aquesta manera el jugador pot configurar el mode gràfic segons la potència del seu ordinador i poder jugar de la forma més òptima.

En cas d'haver triat l'escena anomenada terreny de proves, es carregarà directament la escena on podem provar lliurement el cotxe dins d'aquest escenari.

A més dels menús, el mòdul de visualització s'encarrega de mostrar informació per pantalla. Segons la escena en la que estiguem la informació serà diferent. En el cas de la cursa contrarellotge i la cursa contra IA es mostra per pantalla el temps, la volta, el temps a cadascuna de les voltes, el mini-mapa i la velocitat actual. En canvi a la demostració únicament es mostra el mini-mapa.



Figura 11. Menú de selecció.

El mòdul està compost per diversos scripts:

- **Time Lap.** Mostra part de la informació necessària durant la cursa: temps per a cada volta, temps total, volta actual, voltes totals i sistema de direcció.

El sistema de direcció és l'encarregat de mostrar per pantalla la informació de control. Aquesta informació ve donada per una barra horitzontal formada per punts (figures 12 i 13). El punt del mig ens mostra si s'està accelerant, frenant o no s'accelera ni frena. Els punts dels dos costats ens indiquen la força amb la que el cotxe està girant. Per tant, a la figura 12 estaríem accelerant i girant al màxim cap a la dreta. En canvi, a la figura 13 estaríem frenant i girant un 20% cap a l'esquerra.



Figura 12. Sistema de direcció accelerant i girant a la dreta.



Figura 13. Sistema de direcció frenant i girant a l'esquerra.

- **Velocímetre.** El velocímetre agafa com a paràmetre la velocitat màxima i la divideix entre la mida del velocímetre. D'aquesta manera mostra la velocitat actual. Les línies canvien de color gradualment (verd, groc i vermell) conforme anem accelerant (figura 23 de l'annex).

- **Compte enrere.** S'encarrega de mostrar per pantalla el compte enrere al iniciar una cursa. D'aquesta manera el jugador podrà veure quan pot començar a circular.
- **Mini-mapa.** Compostat per una càmera en vista ortogràfica que ens mostra el circuit i les esferes que simbolitzen cadascun dels cotxes.

#### 4 RESULTATS

A la figura 14 podem veure el joc en funcionament en mode cursa just al iniciar l'escenari.



Figura 14. Joc en execució.

A continuació, a la figura 15, podem veure la escena de demostració de la IA. En aquesta escena el cotxe es mou sol a través de l'escenari sense fi. Quan l'usuari obre el menú de pausa té la opció de tornar al menú de selecció per començar a jugar.

Aquesta escena no conté tots els elements de la HUD, ja que aquest escenari només és de mostra i no és controlat per l'usuari.



Figura 15. Joc en execució en el mode de demostració.

A tots els menús de pausa de cadascun dels escenaris tenim la opció de graduar la qualitat de vídeo i el so (figura 24 de l'annex). En quant al so, si accedim a l'apartat de so del menú de pausa se'ns obre un sub-menú que ens permet pujar, baixar o treure completament l'àudio (figura 16).



Figura 16. Sub-menú de so.

A les figures 17 i 18 podem veure el funcionament de l'algorisme realitzat per la conducció dels adversaris. Es poden veure els punts de ruta (Cubs vermells) i la connexió entre aquests punts (Línies verdes).

A la figura 18 podem veure una mostra de com el cotxe vermell, controlat per l'ordinador, a l'entrar dins de l'àrea de detecció, gira cap al costat on es troba el jugador per entorpir-li el pas.

Respecte al Leap Motion, s'ha incorporat una mà a sobre del cotxe i just a sota de la barra de control (figura 19). Aquesta mà és el que detecta el dispositiu Leap Motion i utilitza Unity per realitzar les accions associades. Fent això sabem si el dispositiu està funcionant correctament i ens ajuda a la hora de controlar el cotxe, ja que podem veure com les accions que realitzem es mostren per pantalla.

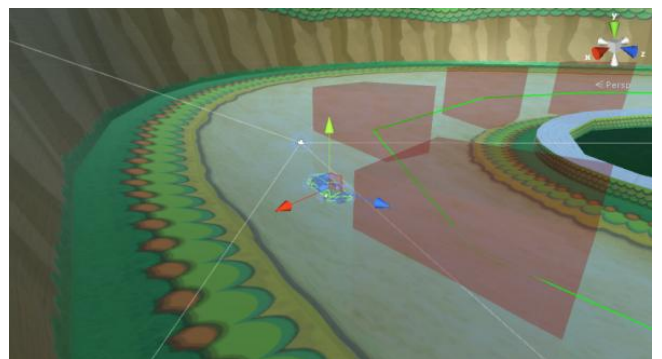


Figura 17. Funcionament IA.

Per comprovar que cadascuna de les funcionalitats implementades funcionessin correctament s'han realitzat proves modulares. S'ha executat el joc i s'han recreat possibles situacions que podien ocórrer. Per exemple, per comprovar que el mòdul de la IA funcionés correctament, s'ha executat l'escena de demostració i s'ha comprovat que pogués realitzar les tres voltes amb el mínim d'errors possibles.

Un cop juntades totes les funcionalitats implementades s'ha realitzat un test d'integració per comprovar que tot en conjunt funcionés correctament. En aquest test d'integració s'ha jugat a cadascuna de les escenes diverses vegades accedint des de l'executable del joc per corroborar que al integrar-ho tot no hem deixat res de banda.

A més dels tests interns s'han realitzat tests amb usuaris.





Figura 18. Funcionament dels sensors de la IA.

Aquests tests externs consisteixen en donar un executable del joc a diverses persones. D'aquesta manera si a alguna d'aquestes persones li passa alguna cosa estranya en una situació concreta, es pot tenir constància i tractar d'arreglar-lo.

S'ha donat l'executable a cinc persones en total. Al principi el fet de controlar el cotxe amb el dispositiu Leap Motion fa que no l'acabin de controlar del tot. Passat un temps els jugadors s'acostumen al seu ús i tenen bones impressions respecte al joc i control del vehicle. Els usuaris joves aprenen ràpid a utilitzar-lo i ho troben entretingut i molt còmode d'utilitzar, en canvi, els usuaris més grans que no solen jugar a videojocs triguen a controlar-lo, tot i que els sembla més divertit que utilitzar el teclat.

Per últim, mostrem els resultats de la modificació de la qualitat de vídeo. S'han realitzat diversos nivells de qualitat per que l'usuari pugui escollir el mode en que el joc li vagi millor. Això s'ha fet ja que per a ordinadors poc potents, si juguem amb la màxima qualitat pot arribar a tenir baixades de frames i limitar la jugabilitat.

S'ha descobert que la qualitat i tipus d'ombres poden arribar a afectar molt la qualitat del joc i per tant el seu rendiment.

Pel que, com es pot observar en la taula 1, la qualitat i tipus d'ombra és el paràmetre que més modifiquem. Pot anar de no tenir-les activades a mostrar només les ombres dures a una distància curta, mostrar les ombres dures i toves a mitja distància i finalment a una distància llunyana. Per reduir més encara la qualitat i per tant el rendiment del joc, toquem la resolució de les textures. Per altra banda, per augmentar més la qualitat del joc, afegim anti-aliasing (reduint l'efecte de dents de serra de l'escena). La primera columna ens mostra el nivell de qualitat, la segona la qualitat de les textures, la tercera si estan les ombres activades i si és així, quin tipus i a quina distància es mostren, la última columna correspon a l'anti-aliasing.

Nom	Qualitat de les textures	Ombres	Antialiasing.
<b>Molt Dolenta</b>	50%	No.	No.
<b>Dolenta</b>	100%	No.	No.
<b>Normal</b>	100%	Dures. Distància: 20	No.
<b>Bona</b>	100%	Dures i toves. Distància: 40	No.
<b>Molt Bona</b>	100%	Dures i toves. Distància: 70	2x
<b>Fantàstica</b>	100%	Dures i toves. Distància: 150	2x

Taula 1. Qualitats de vídeo



Figura 19. Control amb Leap Motion.

## 5 CONCLUSIONS I MILLORES

- S'ha desenvolupat un videojoc del gènere de curses mitjançant un algorisme basat en els agents intel·ligents.
- En la part d'art, s'han dissenyat models de cotxes i textures per a l'escena i els cotxes utilitzant Photoshop CS.
- S'ha desenvolupat un conjunt d'scripts que han permès als cotxes oponents finalitzar els circuit i reaccionar a les accions del jugador.
- S'ha implementat el control del cotxe mitjançant Leap Motion, a part de controlar-lo amb teclat.
- S'han implementat diverses funcionalitats per donar objectius al jugador, com són arribar primer a la meta abans que els oponents o realitzar el circuit en el menor temps possible.
- S'ha après a utilitzar Unity per a la creació de videojocs en 3D.



En conclusió, ha sigut un treball gratificant i interessant de realitzar amb el que he après moltes coses i estic satisfeta amb els resultats obtinguts.

Al principi no tenia massa experiència amb Unity pel que em va costar acostumar-m'hi, però en un parell de setmanes i amb ajudar de diversos tutorials [4][5][6] va ser fàcil adaptar-m'hi i treballar amb ell.

Com a millores podríem incloure:

- Millorar el sistema per desencallar el cotxe. Es podria realitzar un sistema que fes maniobres per poder sortir i continuar la cursa amb normalitat.
- Afegir diferents tipus de pistes amb diverses temàtiques i que cadascuna d'elles tingui repercussió en el control del cotxe. Per exemple, afegir una pista de gel en la qual el cotxe derraparia molt.
- Afegir més oponents i més comportaments als cotxes. D'aquesta manera el joc seria més variat i amb més graus de dificultat.
- Afegir nivell de dificultat a les curses, per que s'adapti al nivell del jugador i que no arribi a avorir-se. Per això es podria afegir climatologia i diferents tipus de superfície en cursa per tenir diferent tipus de lliscament i, per tant, els controls es facin més difícils.

## AGRAÏMENTS

M'agradaria agrair al meu tutor Enric Martí pel suport i assessorament realitzat durant tot el projecte. Sobretot el fet d'haver-me donat la oportunitat de poder programar amb el dispositiu Leap Motion, ja que ha sigut una experiència per a mi.

Agrair també als meus amics per recolzar-me durant tot el transcurs del projecte i haver-me donat consells a la hora de provar el joc.

## BIBLIOGRAFIA

- [1] <http://unity3d.com/es> , web amb tota la informació, documentació i descàrregues de Unity 5. (Data darrer accés: 14/11/2015)
- [2] <https://www.leapmotion.com/> , web amb informació relacionada amb el dispositiu Leap Motion. (Data darrer accés: 06/01/2016)
- [3] <http://www.unityracingkit.com/> , web amb eines per a la implementació de cotxes intel·ligents. (Data darrer accés: 13/12/2015)
- [4] <http://www.raywenderlich.com/78675> , web amb explicacions sobre la GUI de Unity. (Data darrer accés: 14/11/2015)
- [5] <https://fatboxsoftware.wordpress.com/category/unity-project/> , blog amb tutorials per crear diferents funcionalitats a un videojoc de cotxes (Data darrer accés: 04/01/2016)
- [6] <http://pierresemaan.com/category/leap-motion/> , blog amb tutorials per crear diverses funcionalitats amb Leap Motion. (Data darrer accés: 06/01/2016)

- [7] <https://www.nintendo.es/Juegos/Wii-U/Mario-Kart-8-765384.html> , pàgina oficial de Mario Kart per a Wii. (Data darrer accés: 08/02/2016).
- [8] <https://official.trackmania.com> , pàgina oficial de Trackmania Original. (Data darrer accés: 08/02/2016).
- [9] <http://www.monodevelop.com/> , pàgina oficial de l'entorn de programació monodevelop. (Data darrer accés: 08/02/2016).
- [10] <http://www.adobe.com/es/products/photoshop.html> , pàgina web oficial del programa d'edició d'imatges Photoshop. (Data darrer accés: 08/02/2016).

## APÈNDIX

En aquest apèndix mostrem diversos resultats que no s'han mostrat a la secció de resultats. A continuació podem veure els dos menús que poden sortir en escena: el menú de pausa i el menú de fi de cursa (figures 20 i 21) esmentats en aquest article.



Figura 20. Menú de pausa.



Figura 22. Botons de selecció d'escena. D'esquerra a dreta i de dalt a avall: sense seleccionar, amb el ratolí a sobre i fent l'acció de clicar.



Figura 21. Menú de fi de cursa.

Per als menús de selecció s'ha creat un marc que ens indica l'estat dels botons. Tenim tres estats (figura 22): el botó sense seleccionar, el botó quan tenim el ratolí per sobre i el botó quan estem clicant-lo.

En quant a la interfície en joc, mostrem com s'ha realitzat el velocímetre del cotxe (figura 23). Aquest està representat per una capsa on tenim diversos nivells representats amb barres horitzontals. Quant més plena està la capsa a més velocitat va el cotxe. A més, el color és progressiu

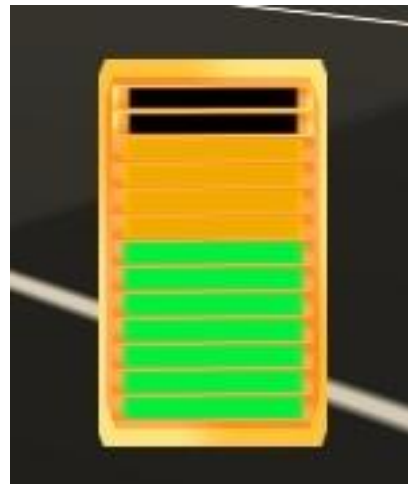


Figura 23. Velocímetre.