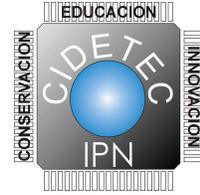




INSTITUTO POLITÉCNICO NACIONAL
CENTRO DE INNOVACIÓN Y DESARROLLO
TECNOLÓGICO EN CÓMPUTO



SIMULADOR DE TIRO CON CAPTURA LASER

T E S I S

QUE PARA OBTENER EL GRADO DE:
MAESTRÍA EN TECNOLOGÍA DE CÓMPUTO:

PRESENTA:

Carlos Alarcón Villarejo

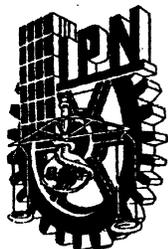
DIRECTORES DE TESIS

Dr. Gabriel Sepúlveda Cervantes

M. en C. Juan Carlos González Robles

México, DF

Enero 2015



INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México, D.F. el día 19 del mes de Enero del año 2015, el (la) que suscribe **Carlos Alarcón Villarejo** alumno (a) del Programa de Maestría en tecnología de Cómputo con número de registro **B121271**, adscrito a **Centro de Innovación y Desarrollo Tecnológico en Cómputo**, manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección del **Dr. Gabriel Sepúlveda Cervantes** y del **M. en C. Juan Carlos González Robles** cede los derechos del trabajo intitulado **Simulador de tiro con captura láser**, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a las direcciones **c_alarcon527@hotmail.com**, **gsepulvedac@ipn.mx** y **igrobles@ipn.mx**. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Carlos Alarcón Villarejo

Agradecimientos

El presente trabajo de tesis me gustaría agradecerle primeramente a dios por bendecirme y guiarme por el camino correcto a lo largo de mi vida.

A mi madre que siempre me ha dado su apoyo incondicional, así como los valores y consejos para hacer siempre lo correcto. A ti mamá que me conoces perfectamente, has sido mi amiga y me amas con todos mis defectos y virtudes. Gracias por todas tus enseñanzas.

A mi padre que siempre esta viendo por el bien de la familia. Gracias por la mano firme que tuviste al educarnos y por que siempre estas al pendiente de que no nos falte nada.

A mis hermanas que siempre me han dado su apoyo, mostrando que muchas veces las acciones dicen mas que mil palabras, las quiero mucho.

A toda mi familia que a pesar de que se encuentran lejos siempre han estado al pendiente de mí, brindandome su cariño y apoyo.

A todos mis amigos que han estado conmigo en las buenas y en las malas, todos ellos siendo parte de una nueva familia, conviviendo la mayor parte de nuestro tiempo en estos últimos años. Gracias amigos por todos sus consejos y su ayuda para el termino de este trajo de tesis así como en mi vida personal.

A mis asesores el Dr. Gabriel Sepúlveda y el M. en C. Juan Carlos González, asi como mi comite tutorial el cual esta integrado por: el Dr. Edgar Alfredo Portilla, el Dr. Juan Carlos Herrera y el M.en C. Eduardo Vega, todos ellos que me han brindado su apoyo tanto emocional como profesionalmente. Gracias por todas sus enseñanzas y paciencia que han tenido hacia mí.

Resumen

En esta tesis se presenta el desarrollo de un simulador de tiro, el cual permite entrenar habilidades de puntería y capacidad de reacción, pudiendo ser utilizado para entretenimiento, así como para entrenamiento para cuerpos policiales, militares o personas que practican la cacería. Este simulador se compone de una cámara web la cual visualiza un ambiente virtual emitido por un proyector, encargándose así mismo de detectar cuando un punto laser es disparado hacia éste, enviando las imágenes adquiridas hacia una computadora para su procesamiento y la generación de disparos dentro de un ambiente virtual.

Para la adquisición de imágenes se utilizó una cámara OptiTrack V-120, la cual tiene una resolución de 640x480 elementos de imagen (píxeles, picture elements) y trabaja con 120 fotografías por segundo. La segmentación de las imágenes recibidas se hizo mediante el método de umbralización, el cual se encarga de separar el fondo, que en este caso es la aplicación proyectada de los objetos encontrados (punto laser emitido por el usuario).

Los modelos 3D utilizados en la aplicación fueron creados en Blender e importados hacia Unity 3D, que es el software encargado de realizar el procesamiento de las imágenes, los cálculos matemáticos para generar las trayectorias reales de los disparos generados dentro del ambiente virtual y todas las animaciones necesarias para el correcto funcionamiento del simulador.

Abstract

In this thesis is presented the development of a shooting simulator, which allows marksmanship skills and response capacity training, and can be used for entertainment, as well as a trainer for police and military bodies, or people who practice hunting. This simulator consists of a web camera that displays a virtual environment generated by a projector, and is also in charge of detecting when a laser spot is fired into it, sending the acquired images to a computer for processing and generating shots within a virtual environment.

An OptiTrack camera V-120 was used for image acquisition, with a 640x480 pixels resolution and 120 frames per second. The segmentation of the images received was done by the thresholding method, which is responsible of separating the background, which in this case is the displayed application and the found objects (laser point emitted by the user).

The 3D models used in this application were created with Blender and imported to Unity 3D, that is the software responsible for image processing, the mathematical calculations to generate the real trajectories of the shots generated within the virtual environment and all necessary animations for correct operation of the simulator.

Índice general

Agradecimientos	I
Resumen	I
Abstract	V
1. Introducción	1
1.1. Estado del arte	1
1.1.1. Simuladores de tiro comerciales	2
1.1.2. Simulador de tiro con foto diodos	7
1.1.3. Simuladores de tiro utilizados en videojuegos	7
1.1.4. Trabajos realizados para el estudio de Balística	9
1.2. Planteamiento del problema	12
1.3. Justificación	12
1.4. Objetivo General	12
1.4.1. Objetivos Particulares	12
1.4.2. Diagrama a bloques del sistema	13
1.4.3. Descripción del sistema	13
1.4.4. Organización del documento	14
2. Realidad Virtual y sus componentes necesarios para la simulación	15
2.1. ¿Qué es Realidad Virtual?	15
2.1.1. Componentes clásicos de un sistema de Realidad Virtual	16
2.1.2. Componentes necesarios para la Realidad Virtual	17
2.2. Software utilizado para Realidad Virtual	18
2.2.1. <i>Unity 3D</i>	18
2.2.2. Blender	23
2.2.3. GIMP	28
2.3. Elementos necesarios para inmersión en simuladores de RV	33

3. Balística	35
3.1. Balística interior	36
3.2. Balística Exterior	37
3.2.1. Trayectoria en el vacío	38
3.2.2. Trayectoria con influencia atmosférica	42
3.2.3. ¿Que es el Modelado?	42
3.2.4. Modelado de la trayectoria de un proyectil	43
4. Visión por computadora	51
4.1. Etapas del procesamiento de imágenes	52
4.2. Umbralización	53
4.2.1. ¿Qué es un láser?	53
4.2.2. Representación del punto laser	55
4.3. Mapeo Espacio cámara - Mundo virtual	55
4.3.1. Distorsión de imagen	57
4.3.2. Matriz de transformación de perspectiva	57
5. Integración	61
5.1. Integración de las ecuaciones de Balística en Unity 3D	61
5.2. Modelos 3D	65
5.3. Escenas Unity 3D	68
6. Conclusiones y Trabajos a Futuro	71
6.1. Resultados	71
6.1.1. Validación del Modelado	71
6.1.2. Conclusiones	74
6.1.3. Trabajos a futuro	74

Índice de figuras

1.1. Campo de tiro móvil [10].	2
1.2. Pantallas individuales [10].	2
1.3. Casa modular digital de tiro [10].	3
1.4. Campo de tiro modular [10].	3
1.5. Simulador virtual de entrenamiento <i>FATSL7</i> [9].	3
1.6. <i>Simulador Policial Disparar/No disparar</i> [7].	4
1.7. <i>Simulador V-Range</i> [8].	4
1.8. <i>Simulador V-100</i> [8].	5
1.9. <i>Simulador V-180</i> [8].	5
1.10. <i>Simulador V-300</i> [8].	5
1.11. Simulador de tiro con carabina Crossman [5].	6
1.12. Fuerzas armadas de Estados Unidos entrenando en simulador <i>VICE</i> [6].	6
1.13. Aplicación por computadora para evaluación de disparo [3].	7
1.14. Pistola Zapper.	7
1.15. Funcionamiento de la pistola Zapper.	8
1.16. Movimientos del Wiimote.	8
1.17. Play Station Move y cámara PS Eye.	9
1.18. Comparación entre la teoría y la práctica para pistola 9mm [13].	10
1.19. Comparación entre la teoría y la práctica para revolver .38 especial[13].	10
1.20. Comparación entre la teoría y la práctica para rifle .22LR[13].	10
1.21. Diagrama a bloques del simulador de tiro.	13
1.22. Elementos principales que componen el Sistema.	13
2.1. Componentes clásicos de un sistema de Realidad Virtual. De Burdea y Coiffet [11].	15
2.2. Vista general de la interfaz de Unity 3D.	19
2.3. Ventana Scene de Unity 3D.	20
2.4. Ventana Game de Unity 3D.	20
2.5. Ventana Hierarchy de Unity.	21

2.6. Ventana Inspector de unity.	21
2.7. Ventana Project de Unity 3D.	22
2.8. Ventana Console de Unity 3D.	22
2.9. Ventana Playmaker de Unity.	23
2.10. Vista general de la interfaz de Blender.	25
2.11. Ventana Info de Blender.	25
2.12. Ventana 3D view de Blender.	26
2.13. Ventana Timeline de Blender.	26
2.14. Ventana Outliner de Blender.	27
2.15. Ventana Properties de Blender.	27
2.16. Vista general de la interfaz de GIMP.	29
2.17. Caja de herramientas en GIMP.	30
2.18. Ventana de imagen en GIMP.	30
2.19. Ventana de historial de cambios en GIMP.	31
2.20. Ventana que muestra las capas y los canales en GIMP.	31
2.21. Ventana que muestra las paletas de colores, patrones y degradados en GIMP.	32
2.22. Ventana que muestra los diferentes tipos de pinceles en GIMP.	32
3.1. Partes de un cartucho.	36
3.2. Anima con estrias.	37
3.3. Trayectoria del proyectil en línea recta.	37
3.4. Trayectoria del proyectil (parábola).	37
3.5. Parámetros establecidos a partir de la trayectoria parabólica.	38
3.6. Trayectoria parabólica simulada.	41
3.7. Sistema fuerza-masa.	42
3.8. Amortiguador para fricción viscosa.	43
3.9. Fuerzas que afectan al proyectil.	48
3.10. Trayectoria del proyectil afectada por el viento.	49
4.1. Procesamiento de imagen mediante segmentación [16].	51
4.2. Etapas del procesamiento de imágenes [20].	52
4.3. Cámara V120-SLIM.	52
4.4. Radiación emitida por longitud de onda para varias temperaturas.	54
4.5. Áreas de trabajo utilizadas en el procesamiento de imagen con Optitrack V120.	56
4.6. Plano umbralizado, mostrando de color azul el fondo y de blanco un punto laser.	56
4.7. Transformación del área deforme a un rectángulo uniforme y normalizado.	57
4.8. Coordenadas utilizadas para el modelo de la cámara.	58

4.9. Calibrando una de las 4 esquinas del plano umbralizado.	60
5.1. Trayectoria 2D simulada en Unity 3D en modo edición.	62
5.2. Trayectoria 3D simulada en Unity 3D en modo edición.	62
5.3. Configuración de parámetros para trayectoria 3D en Unity 3D (modo edición). .	63
5.4. Trayectoria 2D simulada en Unity 3D en modo PLAY.	64
5.5. Configuración de parámetros para Trayectoria 2D en Unity 3D (modo PLAY). .	64
5.6. Modelo de un pato realizado en Blender.	65
5.7. Modelo de un pato realizado en Blender e importado a Unity 3D.	65
5.8. Modelo de un venado realizado en Blender.	66
5.9. Modelo de un venado realizado en Blender e importado a Unity 3D.	66
5.10. Modelo de una casa de tiro realizado en Blender.	67
5.11. Modelo de una casa de tiro realizado en Blender e importado a Unity 3D.	67
5.12. Escena de Unity para calibración de la cámara Optitrack.	68
5.13. Casa de tiro con patos como objetivos.	68
5.14. Escena de la casa de tiro con patos como objetivos.	69
5.15. Escena de un terreno con venados que aparecen aleatoriamente.	69
6.1. Simulación que describe la trayectoria de un proyectil, sin efecto del viento. . . .	72
6.2. Simulación que describe la trayectoria de un proyectil, con efecto del viento. . .	72
6.3. Simulación del modelo matemático con un disparo a 0 grados con pistola 9mm. .	73
6.4. Simulación del modelo matemático con un disparo a 15 grados con pistola 9mm.	73

Capítulo 1

Introducción

Con el paso del tiempo, el avance tecnológico crece continuamente, esto da lugar a que nuevas tecnologías permitan a los usuarios tener una mejor interacción con los sistemas de realidad virtual (Sistemas RV). Hoy en día existen diferentes dispositivos para manipular dichos sistemas, ya sea unidireccionales (teclado, ratón, mandos de control, etc.) o bidireccionales (dispositivos hápticos, pantallas de toque, etc.), gracias a esta mejoría de interacción se han creado simuladores más cercanos a la realidad.

Existen diferentes tipos de simuladores tales como: carreras, manejo de camiones, vuelo, disparo, y juegos mecánicos, entre otros. Las aplicaciones de estos simuladores se extienden desde usos básicos, como son entretenimiento en plazas comerciales, hasta modos avanzados utilizados para entrenamiento policial o militar, dependiendo del público al cual estén dirigidos.

Actualmente se están desarrollando simuladores de tiro en países como Francia, España y Estados Unidos de América, los cuales permiten entrenar habilidades tales como la velocidad de reacción y la puntería, siendo utilizados en clubes de tiro, academias policiales y por cazadores durante las temporadas de veda.

El presente trabajo de investigación muestra la integración de diferentes tecnologías para el diseño e implementación de un simulador de tiro que representa los disparos con armas de fuego, y el cual mediante la detección de un punto laser a través de una cámara, genera un disparo desde el mundo real a un mundo virtual.

1.1. Estado del arte

En esta sección se presenta la revisión de algunos trabajos de Realidad Virtual, que han sido desarrollados en el sector de simuladores de tiro a nivel mundial. Se presenta una pequeña descripción de su funcionamiento y de sus características, destacando los aspectos que han servido como referencia para este trabajo de investigación.

1.1.1. Simuladores de tiro comerciales

Laser Shot [10], es una empresa dedicada al entrenamiento de militares y policías profesionales, mediante el uso de réplicas de armas que al accionar el mecanismo de disparo generan un punto laser infrarrojo sobre una superficie, proyectando el disparo en un ambiente virtual como el que se muestra en las figuras 1.1 y 1.2. Este punto laser es detectado por una cámara, la cual se encarga de mapear sus coordenadas e integrarlo como un disparo dentro de dicho ambiente.

Así mismo, este sistema cuenta con la opción de utilizar armas reales dentro de una casa modular en donde se proyecta un ambiente virtual sobre sus paredes (ver figura 1.3), o en un campo de tiro modular como el que se muestra en la figura 1.4. Estas opciones utilizan el mismo procedimiento para la generación del disparo dentro de un ambiente virtual, pero con la diferencia de que al realizar disparos los proyectiles impactados en las paredes son detectados mediante cámaras térmicas, mapeando estos puntos de impacto dentro de dicho ambiente.



Figura 1.1: Campo de tiro móvil [10].



Figura 1.2: Pantallas individuales [10].



Figura 1.3: Casa modular digital de tiro [10].



Figura 1.4: Campo de tiro modular [10].

La empresa francesa *Meggitt Training Systems* [9], ofrece un simulador de entrenamiento virtual llamado *FATSL7* (ver figura 1.5). Este sistema proyecta videos sobre una pared, los cuales son grabados con personas reales y permiten simular diferentes situaciones de peligro en donde los policías deben reaccionar rápidamente. Estos videos ya vienen cargados en el simulador, sin embargo se cuenta con la opción de grabar nuevos escenarios e instalarlos. Otra opción que brinda el simulador es la de soportar entrenamiento simultáneo de múltiples usuarios, usando hasta ocho armas controladas por el sistema para entrenamiento en equipo.

Figura 1.5: Simulador virtual de entrenamiento *FATSL7* [9].

La empresa *Ti Training* [7] realizó un simulador de disparo, el cual permite practicar con escenarios llamados *Shoot/Dont shoot* o por su traducción al español *Disparar/No disparar*. Esta simulación ayuda a capacitar a oficiales de policía en situaciones en las que debe decidir cuándo disparar a un sospechoso o atacante, haciendo frente a escenarios de la vida real, como se observa en la figura 1.6.



Figura 1.6: *Simulador Policial Disparar/No disparar* [7].

La empresa *VirTra* [8] ofrece simuladores de disparo que se basan en el mismo funcionamiento de la detección de un punto laser sobre una proyección, la cual es visualizada mediante una cámara. Estos simuladores se clasifican por sus ángulos de visión, los cuales van desde una hasta cinco pantallas:

- *V-range* (ver figura 1.7).- Simulador utilizado para practicar puntería, el cual se divide en diferentes cabinas, cada una para un solo usuario.



Figura 1.7: *Simulador V-Range* [8].

- *V-100* (ver figura 1.8).- Simulador utilizado para practicar puntería y capacidad de reacción ante situaciones de combate urbano, permitiendo el entrenamiento en equipo.



Figura 1.8: *Simulador V-100* [8].

- *V-180* (ver figura 1.9).- Simulador que cuenta con tres pantallas las cuales generan un ángulo de visión de 180 grados, permitiendo entrenamiento en equipo, limitado al tamaño de la plataforma que sostiene las pantallas.



Figura 1.9: *Simulador V-180* [8].

- *V-300* (ver figura 1.10).- Este simulador utiliza la misma plataforma del *V-180*, pero se agregan dos pantallas más, para generar un ángulo de visión de 300 grados.



Figura 1.10: *Simulador V-300* [8].

Lasersniper [5], es un simulador de tiro con diferentes escenarios en los que se incluyen: cazador de patos, pueblo vaquero y galería de tiro policial. Este simulador viene con dos armas auténticas que han sido modificadas, se trata de dos carabinas *Crossman modelo 1077* que proyectan un punto laser rojo, ver figura 1.11.



Figura 1.11: Simulador de tiro con carabina Crossman [5].

La empresa *DAS* [6] ofrece el simulador *Virtual Interactive Combat Environment (VICE)*, el cual se compone de distintas cabinas para formar líneas de entrenamiento, permitiendo su configuración según el número de usuarios que entrenan al mismo tiempo y el espacio en donde se instalará dicho simulador. Cada cabina está integrada por un proyector, sistema de sonido, CPU, cámara y una replica de arma que dispara un punto laser infrarrojo. *VICE* (ver figura 1.12) está diseñado para entrenar habilidades tales como puntería, estrategias de combate y habilidad de reacción ante situaciones de peligro, todas ellas necesarias para militares, policías y agentes de seguridad nacional. Cuenta con entrenamientos de manera individual y en equipo, en escenarios tanto urbanos como rurales.



Figura 1.12: Fuerzas armadas de Estados Unidos entrenando en simulador *VICE* [6].

1.1.2. Simulador de tiro con foto diodos

En [3] se desarrolló un simulador de tiro, el cual utiliza foto diodos organizados en filas y columnas para detectar un punto laser mediante un microcontrolador. Estas lecturas del microcontrolador son enviadas hacia una computadora, la que se encarga de almacenar los datos obtenidos y evaluar los disparos realizados de manera gráfica (ver figura 1.13), generando un historial de rendimiento de cada tirador.



Figura 1.13: Aplicación por computadora para evaluación de disparo [3].

1.1.3. Simuladores de tiro utilizados en videojuegos

Pistola Zapper de Nintendo

En [4] se describe el funcionamiento de la pistola Zapper (figura 1.14) de Nintendo, la cual salió a la venta en 1985 por dicha compañía y fue compatible con muchos juegos en los que se utilizaba el mismo procedimiento para la detección del disparo.



Figura 1.14: Pistola Zapper.

Entre los juegos mas conocidos destaca *"Duck Hunt"*, éste es un famoso juego en el que un perro cazador le dispara a los patos que van apareciendo en la pantalla. El usuario utiliza la pistola Zapper la cual esta integrada por un fotodiodo en la punta del cañón, que es el encargado de detectar cambios de iluminación. El juego esta configurado de tal forma que al apretar el gatillo de la pistola Zapper la pantalla de la televisión se pone de color negro durante un fotograma, al siguiente fotograma se pinta de color blanco un cuadrado que cubre al objetivo (en este caso el pato), detectando el color blanco sobre el fondo negro determinando así que estábamos apuntando correctamente al objetivo. Este procedimiento se muestra en la figura 1.15.



Figura 1.15: Funcionamiento de la pistola Zapper.

Wii Remote de Nintendo

El wii Remote (control remoto wii) conocido como Wiimote, es el control principal utilizado en el sistema Wii de Nintendo. Éste esta integrado por un acelerómetro que es el instrumento encargado de medir aceleración debida a la gravedad, así envía hacia la consola datos de movimientos relativos en todos los ejes de coordenadas detectando giros e inclinaciones [22]. Además el Wiimote detecta la luz que emite la Barra sensor de la consola para un mejor funcionamiento. Esta barra mide aproximadamente 20 cm de longitud y cuenta con cinco LEDs infrarrojos en cada extremo de la barra. El sensor óptico del Wiimote localiza estos LEDs para generar una pantalla virtual muy cerca de la pantalla real de la televisión, tomando en consideración la posición de la barra con respecto a la televisión y la configuración de ésta. Así el Wiimote sirve como el puntero de un ratón en la pantalla virtual trasladando esos datos hacia la televisión como se muestra en la figura 1.16.

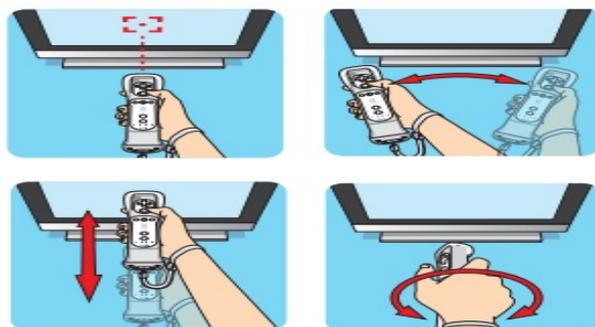


Figura 1.16: Movimientos del Wiimote.

Play Station Move

Play Station Move es un control de Play Station 3 [21], el cual esta integrado por un giroscopio y un acelerómetro, los cuales se encargan de detectar la inclinación del control que es manipulado por el usuario, así mismo cuenta con una esfera luminosa multicolor que es detectada por la cámara PS Eye, el control y la cámara se observan en la figura 1.17. La esfera luminosa es detectada por la cámara aunque ésta no se este apuntando directamente, gracias a esto es posible calcular la distancia entre el control y la pantalla mediante el tamaño que se registra de la esfera luminosa. La integración de los componentes descritos anteriormente permiten al usuario utilizar el control como un puntero de un ratón manipulando el sistema Play Satation 3.



Figura 1.17: Play Station Move y cámara PS Eye.

1.1.4. Trabajos realizados para el estudio de Balística

En [13] se realizó un estudio sobre balística exterior, la cual es una subdivisión de la balística y será explicada posteriormente en el Capítulo 3. En dicho estudio se describen los principales conceptos de esta rama de la balística, así como las clasificaciones de movimientos que generan los proyectiles durante su recorrido, incluyendo las diferencias de trayectorias entre el vacío y trayectorias en la atmósfera; para ello se consideran las principales características de un proyectil como son su forma geométrica, material y calibre, las cuales influyen en dichas trayectorias. A continuación se muestran los resultados de disparos realizados a diferentes distancias con el cañón del arma en forma horizontal y con los ángulos de corrección para que el proyectil impacte en la zona deseada, producidos con diferentes armas, como son: pistola 9mm (ver figura 1.18), revolver .38 especial (ver figura 1.19) y rifle .22 LR (ver figura 1.20).

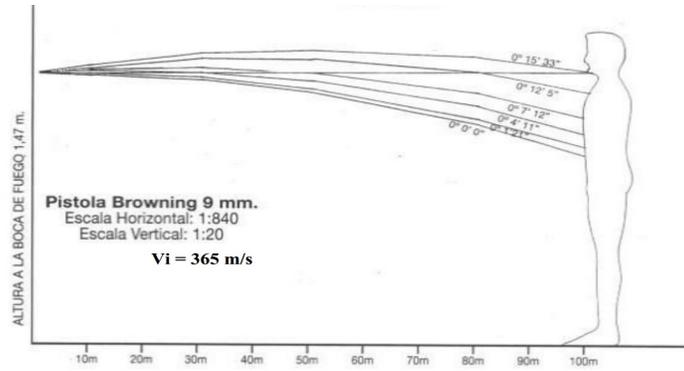


Figura 1.18: Comparación entre la teoría y la práctica para pistola 9mm [13].

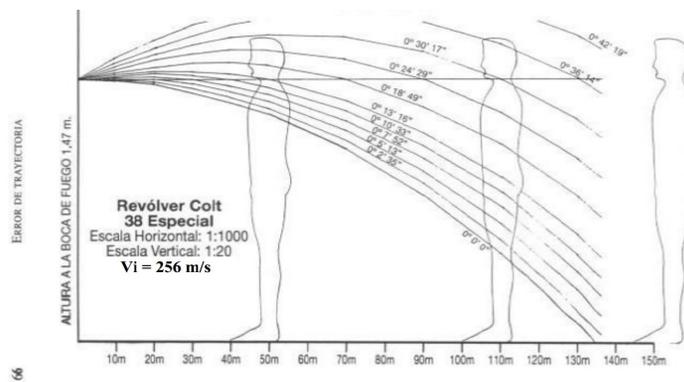


Figura 1.19: Comparación entre la teoría y la práctica para revólver .38 especial[13].

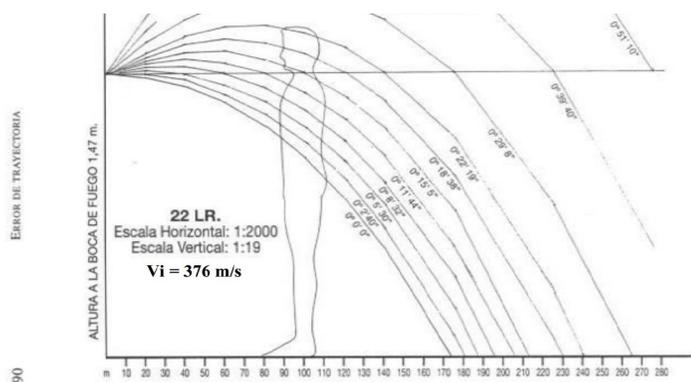


Figura 1.20: Comparación entre la teoría y la práctica para rifle .22LR[13].

En el curso de balística [14] se definen los conceptos mas importantes involucrados en su estudio, mismos que servirán de referencia en capítulos posteriores. Estos conceptos ayudarán a un mejor entendimiento de cada una de sus etapas, así como la historia de los inicios de la pólvora hasta la formación de una bala y sus características actuales. Otro punto muy importante que se considera en esta referencia son las características y propiedades de los calibres más comunes utilizados en armas convencionales, así como la normatividad general para manejo de explosivos.

En el manual de tiro [19] se pueden identificar los diferentes tipos de armas de fuego que establece el Registro Nacional de Armas (RENAR), así como su clasificación y subdivisión. También explica como diferenciar los cartuchos y cada uno de sus calibres, describiendo detalladamente las piezas que componen a las armas de fuego, incluyendo algunos consejos para la limpieza y cuidados que se deben tener en su manejo. Por último, se han tomado en consideración los consejos que se brindan en esta referencia para empuñar el arma y controlar el retroceso que ésta genera, así como la postura que se debe tener para efectuar un buen disparo.

1.2. Planteamiento del problema

La adquisición de simuladores de realidad virtual, principalmente en el extranjero, requiere de inversiones elevadas. Un sistema equivalente al desarrollado en este trabajo de tesis puede alcanzar un costo de 70 millones de pesos, cantidad que se incrementa cuando se consideran los costos asociados al mantenimiento, actualización del equipo y capacitación, entre otros rubros. Además, debe tomarse en cuenta que su uso en academias de cuerpos policiales o militares tales como en el adiestramiento de las fuerzas armadas mexicanas requiere de políticas estrictas de seguridad, mismas que pueden verse comprometidas si se recurre a proveedores externos.

1.3. Justificación

Considerando los costos y dependencia tecnológica de estos simuladores que no se desarrollan en México, se propone realizar un simulador de tiro mediante la integración de diferentes tecnologías, que permita la independencia tecnológica a un costo accesible, cuyas características sean equivalentes o superiores a las de los desarrollos disponibles en el mercado, y que permita entrenar diferentes habilidades tales como la velocidad de reacción y la puntería. Adicionalmente, el sistema contará con cálculos balísticos, los cuales permitirán una simulación más cercana a la realidad debido a que las trayectorias de las balas disparadas serán generadas considerando su forma geométrica y diferentes combinaciones de parámetros del clima.

1.4. Objetivo General

Realizar un simulador de tiro el cual capture un laser mediante una cámara web y utilice ese punto de referencia para representar un disparo en un mundo virtual.

1.4.1. Objetivos Particulares

- Procesamiento de imagen mediante cámara web.
- Captura de laser mediante cámara web.
- Segmentación de imagen por medio de umbralización.
- Mapeo del espacio cámara web a espacio mundo virtual.
- Desarrollo del ambiente virtual con siluetas para práctica de tiro.
- Instrumentación de un arma de plástico, para interactuar con el ambiente virtual.
- Desarrollo de ecuaciones de balística para determinar trayectoria de las municiones.

1.4.2. Diagrama a bloques del sistema

El sistema propuesto para el desarrollo del simulador de tiro, se puede ver en la figura 1.21.

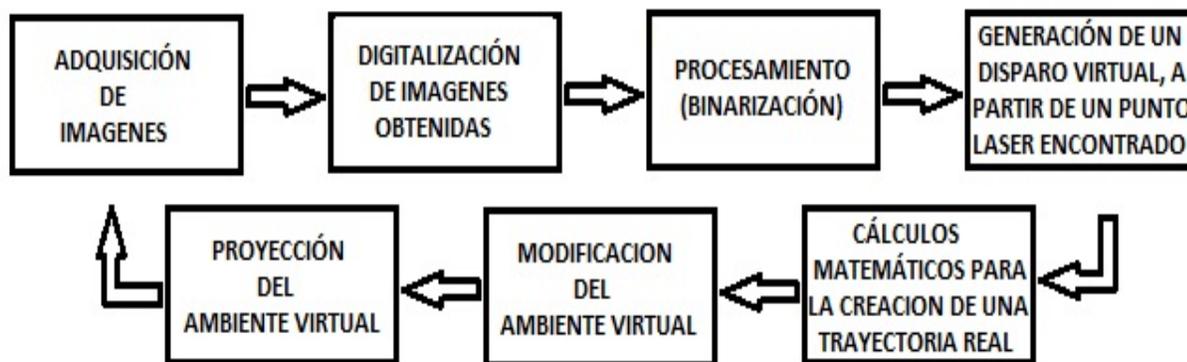


Figura 1.21: Diagrama a bloques del simulador de tiro.

1.4.3. Descripción del sistema

Los elementos que componen el simulador de tiro se describen en la figura 1.22.

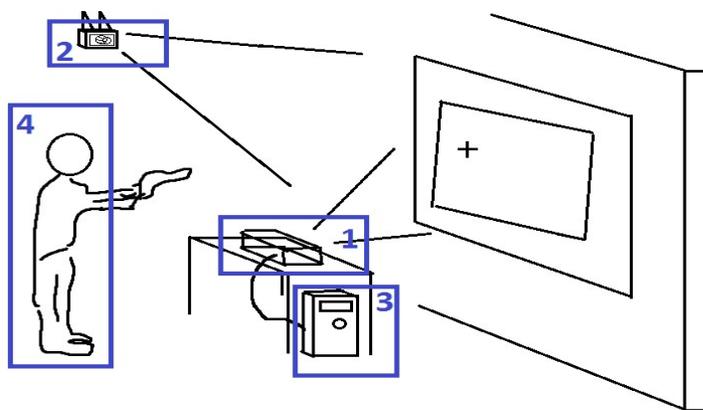


Figura 1.22: Elementos principales que componen el Sistema.

1. Proyector.- Es el encargado de proyectar el ambiente virtual de la aplicación, sobre una superficie perpendicular al piso, ya sea una pared o una pizarra blanca de papel montada a ésta.
2. Cámara web.- Es una cámara digital, cuya función principal es la de capturar las imágenes que emite el proyector, así como la captura del punto laser que es disparado por el usuario, transmitiendo estos datos hacia la computadora.
3. Computadora.- En ella se ejecuta la aplicación que genera el ambiente virtual, el cual es necesario para la interacción con el usuario y emitido por el proyector. Así mismo, en la

computadora se realizan todos los cálculos balísticos que son necesarios para generar una trayectoria real del proyectil dentro de la aplicación. La computadora recibe los datos de la cámara para realizar el procesamiento, generando una imagen binaria la cual se compone de un fondo y un punto laser detectado como un objeto, mapeando esas coordenadas obtenidas hacia el ambiente virtual.

4. Usuario.- Es la persona que interactúa con la aplicación; esta interacción permite al usuario entrenar habilidades de puntería, así como la adquisición de conocimiento sobre las trayectorias reales de un proyectil.

1.4.4. Organización del documento

En este capítulo se realiza una introducción sobre los simuladores de Realidad Virtual, enfocándose en simuladores de tiro, así mismo se describen simuladores de tiro comerciales, particulares y simuladores utilizados en videojuegos, junto con algunos trabajos relacionados en el área de balística. También se menciona el planteamiento del problema, así como el objetivo general, objetivos particulares y la justificación de este trabajo de tesis.

En el capítulo 2 se describe lo que es Realidad Virtual, así como los componentes necesarios para que se lleve a cabo una simulación de manera correcta. También se muestran el software utilizado en este trabajo de tesis para generar el simulador de tiro.

El capítulo 3 presenta definiciones de los conceptos básicos en el área de balística, así como las herramientas matemáticas que incluye los parámetros y cálculos necesarios para determinar el modelo que describe la trayectoria de un proyectil.

En el capítulo 4 se muestra el método de umbralización el cual mediante el procesamiento de imágenes obtenidas por una cámara Optitrack V120, permitiendo la detección de un punto laser generado en un mundo real y mapeado a un mundo virtual como un disparo.

El capítulo 5 es la parte principal de este trabajo de tesis, dado que en él se integran las ecuaciones de balística, modelos 3D y el procesamiento de imágenes para generar un simulador de tiro que describe trayectorias reales de proyectiles estándar avalados por la OTAN.

El capítulo 6 incluye las conclusiones, los resultados obtenidos y el trabajo a futuro que se propone para mejorar el simulador de tiro.

Capítulo 2

Realidad Virtual y sus componentes necesarios para la simulación

2.1. ¿Qué es Realidad Virtual?

La Realidad Virtual (RV) es una simulación generada por computadora, la cual utiliza un conjunto de imágenes para crear un ambiente no real que es utilizado para interactuar con el usuario. Este ambiente no es estático, ya que responde a entradas de usuarios (movimiento de usuario, gestos, comandos, etc.); la computadora detecta estas entradas de usuario y modifica el ambiente virtual instantáneamente. Así, la RV es el medio que permite tener una experiencia simulada, cercana a la realidad física, pero reduciendo el peligro de ésta y creando además escenarios no posibles en el mundo real. Existen cinco componentes clásicos en un sistema de RV, los cuales se muestran en la figura 2.1.

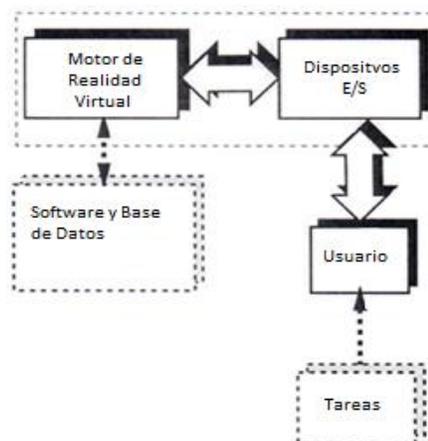


Figura 2.1: Componentes clásicos de un sistema de Realidad Virtual. De Burdea y Coiffet [11].

2.1.1. Componentes clásicos de un sistema de Realidad Virtual

1. Dispositivos de E/S (Entrada/Salida).- Para que exista una interacción computadora-usuario se requieren interfaces especiales diseñadas para introducir comandos de un usuario en la computadora y proveer retroalimentación desde la simulación al usuario. Estos dispositivos son usados ya sea por entradas de usuario (como seguidores de posición, guantes o teclados) o salidas (como gafas de realidad virtual, pantallas, movimientos de brazos robóticos, etc.).
2. Motor de realidad virtual.- Es un componente clave de cualquier sistema de RV, el cual lee los dispositivos de entrada, accede a las bases de datos que dependen de la tarea, realiza los cálculos requeridos para actualizar el estado del mundo virtual, y envía los resultados a las pantallas de salida. Durante una simulación de RV es imposible predecir todas las acciones de usuario y almacenar todos los estados correspondientes del mundo virtual en memoria, por lo que el mundo virtual es creado y destruido constantemente.
3. Software y base de datos.- Otro aspecto importante es el modelado del mundo virtual. Esto significa en primer lugar la asignación de dispositivos de E/S con la escena de la simulación, lo cual requiere de calibración de las características del usuario así como del código de comunicación de E/S; el siguiente paso, siguiendo la asignación de dispositivos de E/S, es la base de datos de los objetos que poblarán el mundo virtual. Esto significa modelar la forma del objeto, apariencia, restricciones cinemáticas, comportamiento inteligente, y características físicas (peso, inercia, dureza, etc.).
4. Usuario.- El usuario es aquella persona o grupo de personas que interactúan con el sistema de realidad virtual.
5. Tareas.- Son los procesos que se ejecutan dentro del sistema.

2.1.2. Componentes necesarios para la Realidad Virtual

En la RV son necesarios cuatro elementos para experimentarla, los cuales se mencionan a continuación:

1. Mundo virtual.- Es un espacio imaginario comúnmente manifestado a través de un medio. Este tipo de mundos son generados en una computadora mediante algún programa y se visualizan en las pantallas correspondientes.
2. Inmersión.- Este término es usado en dos formas:
 - a) Inmersión física.- Estimulo sintético de los sentidos del cuerpo a través del uso de tecnología; esto no implica todos los sentidos o que el cuerpo entero sea inmerso o envuelto.
 - b) Inmersión mental.- Se produce cuando la persona deja de percibir de forma clara su medio natural al concentrar toda su atención en un medio artificial y tener la sensación de estar dentro de éste.
3. Retroalimentación sensorial.- El sistema RV proporciona retroalimentación sensorial directa a los participantes con base en su posición física. Para tener una retroalimentación inmediata se requiere el uso de computadoras de alta velocidad como dispositivos mediadores.
4. Interactividad.- Para que la RV parezca auténtica debe responder a las acciones del usuario; esto es llamado interactividad.

2.2. Software utilizado para Realidad Virtual

2.2.1. *Unity 3D*

Unity 3D es un motor de videojuegos multiplataforma disponible para Windows y OS X, permitiendo desarrollar videojuegos para Windows, OS X, Linux, Play Station 3, Xbox 360, Nintendo Wii, Ipad y Android. Es una herramienta multiplataforma de fácil aprendizaje, con la cual se pueden desarrollar desde videojuegos a mundos virtuales.

Existen dos tipos de licencia: uno que es gratuito pero con muchas limitaciones, entre las más importantes se encuentra una marca de agua la cual no se puede desactivar. En la versión Pro se incluyen tutoriales, ejemplos de proyectos, soporte y actualizaciones futuras sin necesidad de volver a pagar. Un complemento muy importante es el Asset Store, el cual es una herramienta en la que miles de usuarios comparten de forma gratuita o con costo una amplia gama de paquetes, en los que se incluyen modelos, contenidos interactivos, texturas, scripts, etc., que sirven para facilitar el uso de Unity.

Para este proyecto se pueden crear figuras geométricas básicas, así como terrenos usando el editor; sin embargo, Unity 3D no posee herramientas para la elaboración de modelos 3D por lo que es necesario un software adicional de modelado 3D. Algunas aplicaciones compatibles son: Maya, Blender, Cinema 4D, 3ds MAX. En este caso se utiliza Blender, ya que al importar a Unity los modelos en formato .Blend (formato de modelos nativo de Blender), los cambios realizados en dicho objeto se actualizarán automáticamente en el proyecto de Unity, sin necesidad de reimportar los modelos 3D. Así mismo, al importar los modelos 3D también se importan automáticamente todas las animaciones que estos contengan, dándoles movilidad.

Unity cuenta con una versión personalizada de MonoDevelop para la creación y edición de scripts los cuales nos permiten interactuar con objetos en nuestro mundo virtual. Por ejemplo la creación o destrucción de un objeto, abrir una puerta, encender o apagar una luz, inteligencia artificial de personas virtuales, etc.

Otra opción de este motor, es el editor de partículas, las cuales ayudan a una mejor simulación, por ejemplo: de fuego de una veladora, humo saliendo de una chimenea, gotas de lluvia, etc. Añadiendo a esto efectos de sonido para una simulación mas realista.

Interfaz de Unity 3D

En la figura 2.2 se muestra la pantalla principal de Unity 3D.

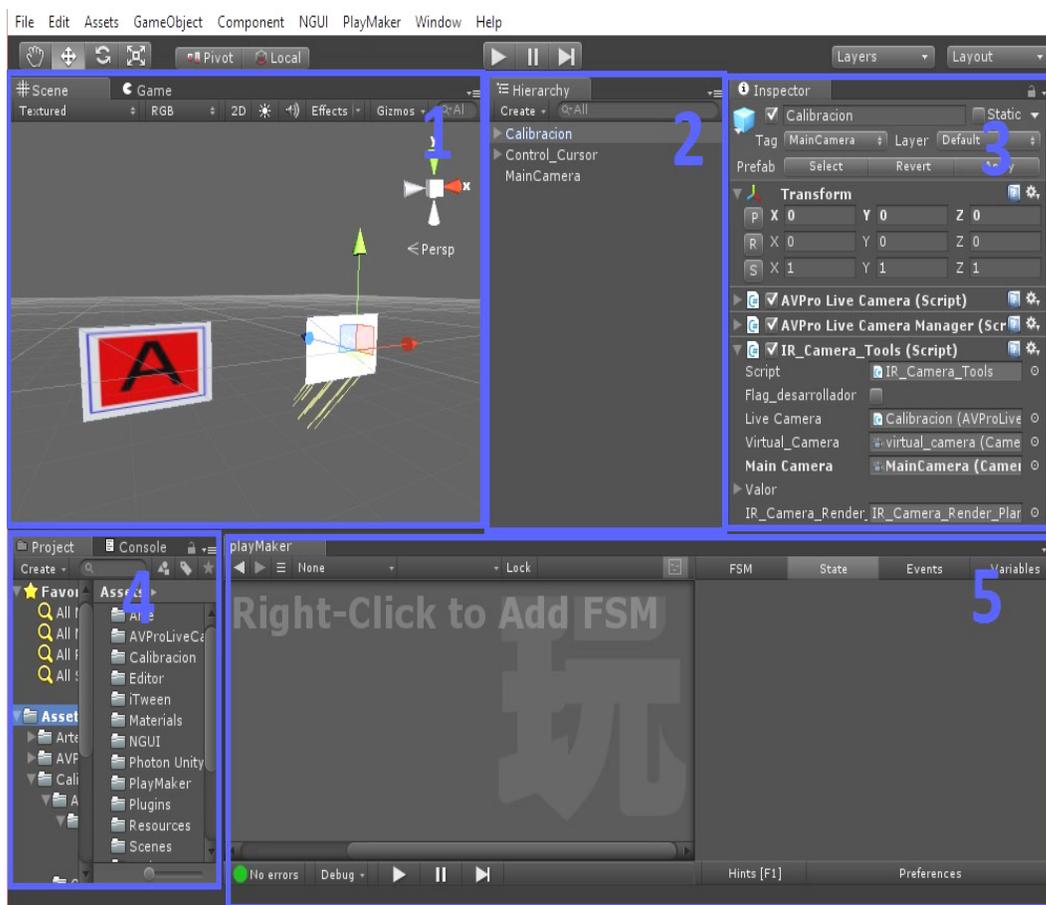


Figura 2.2: Vista general de la interfaz de Unity 3D.

A continuación se describen las ventanas que componen la interfaz de Unity 3D.

1. Ventana Scene y ventana Game.

- **Ventana Scene.**-La ventana Scene se utiliza para seleccionar y posicionar objetos dentro de nuestro ambiente virtual. Estos objetos pueden ser: cámaras, luces, modelos 3D, etc. En esta ventana se puede navegar en el ambiente virtual mediante el mouse o las flechas del teclado. Para este punto, Unity proporciona pulsaciones de teclado para las operaciones más comunes. Así mismo, si se selecciona un objeto que se encuentra en nuestra escena, la vista se moverá hasta centrarse en dicho objeto, dándonos opciones como girarlo, moverlo, alejarse, acercarse u orbitar sobre él, como se puede observar en la figura 2.3.

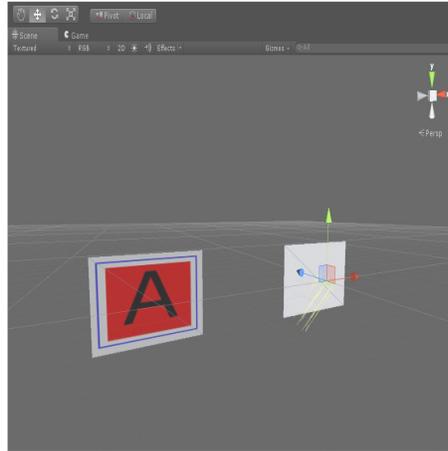


Figura 2.3: Ventana Scene de Unity 3D.

- Ventana Game.- La ventana Game muestra lo que esta siendo visto por la cámara utilizada en el proyecto. Mientras no este corriendo la aplicación, esta vista permanecerá estática. Así mismo, esta ventana tiene tres botones en la parte superior izquierda para configuraciones previas. El primer botón (Maximize on play) al estar habilitado, maximizará nuestra ventana de Game al 100 % al inicializar la aplicación; el segundo botón (Stats), muestra el rendimiento gráfico de la aplicación; por último el botón (Gizmos), mientras este habilitado mostrará todos los gizmos que aparecen en la escena, ver figura 2.4.



Figura 2.4: Ventana Game de Unity 3D.

- Hierarchy.- Esta ventana contiene todos los objetos que se encuentran en la escena actual. Estos objetos pueden ser Gameobjects vacíos, modelos, prefabs, entre otros, y pueden unirse varios de estos para formar uno nuevo el cual contendrá todos los objetos unidos, quedando como padre de estos el objeto que no fue arrastrado al momento de la unión. Al mover al objeto padre a través de la ventana Scene, todos los objetos hijos se moverán y girarán también en la misma dirección y la misma cantidad de coordenadas, ver figura 2.5.

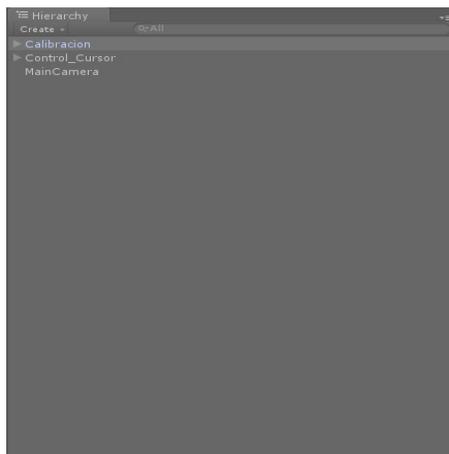


Figura 2.5: Ventana Hierarchy de Unity.

- Inspector.- En esta ventana se muestran todas las propiedades de los objetos que se encuentran en el proyecto. Estas propiedades pueden ser modificadas directamente en esta ventana, ver figura 2.6.

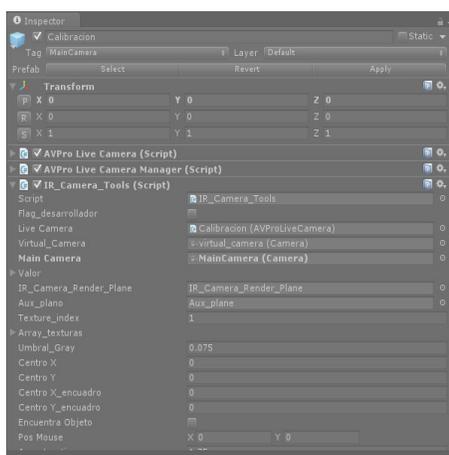


Figura 2.6: Ventana Inspector de unity.

4. Ventana Project y ventana Console.

- Ventana Project.- En esta ventana se muestran todos los archivos que son utilizados en el proyecto, así como modelos 3D, prefabs y scripts, ver figura 2.7.

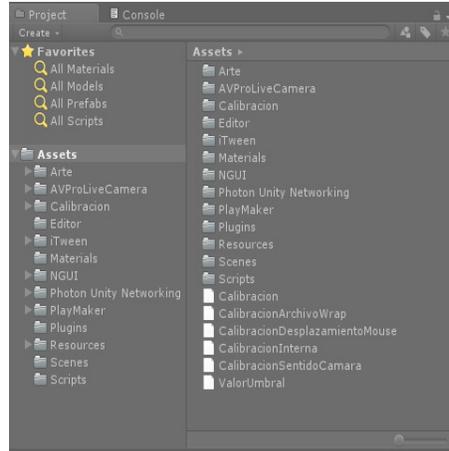


Figura 2.7: Ventana Project de Unity 3D.

- Ventana Console.- Esta ventana muestra los mensajes generados por el usuario, advertencias, y errores del proyecto, ver figura 2.8.

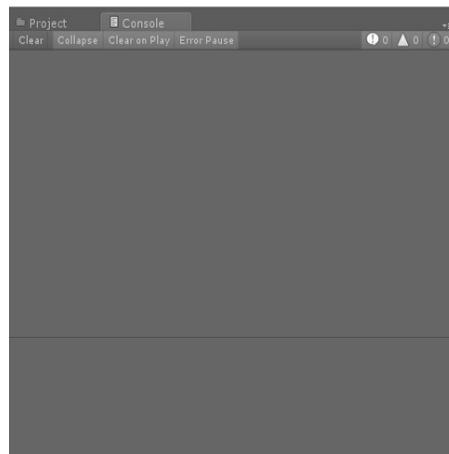


Figura 2.8: Ventana Console de Unity 3D.

5. Playmaker.- Playmaker es un complemento para unity 3D el cual se basa en programación de máquinas de estado. En cada estado se pueden realizar múltiples acciones, como son animaciones de modelos, suma de enteros, comparaciones de cadenas, etc. Mediante eventos se hacen las transiciones de estado, permitiendo tener múltiples estados con conexión uno a uno, ver figura 2.9.



Figura 2.9: Ventana Playmaker de Unity.

2.2.2. Blender

Blender es un software libre multiplataforma con licencia GNU, gracias a esto, es modificado y compartido constantemente por varias personas que corrigen los errores y mejoran su funcionalidad, contribuyendo a su crecimiento y protegiéndolo de intentos de apropiación que restrinjan esas libertades a los usuarios.

Esta diseñado principalmente para el modelado de objetos 3D, pudiendo crear animaciones de estos en ambientes iluminados (incluyendo sombras), generando renderizados de imágenes o vídeos según sea el propósito de su uso. Así mismo, cuenta con la opción de desarrollo de videojuegos ya que posee un motor de juegos interno.

Una gran ventaja de este software es su tamaño, que es muy pequeño comparado con otros paquetes de modelado 3D, pero cuenta con una gran variedad de herramientas que permiten obtener resultados de muy buena calidad. Algunas de estas herramientas son generación de figuras geométricas básicas, incluyendo curvas, líneas, mallas, etc., así como cálculos para realizar transformaciones como escala de objetos, translaciones y rotaciones.

Entre algunas de sus características encontramos las siguientes:

- Modelado 3D.- Permite la creación de modelos 3D a partir de figuras geométricas, edición de aristas, unión de vértices, deformación de mallas, etc.
- Asignación de Materiales.- Configurar las propiedades físicas de los objetos.
- Texturizado de modelos.- Se cubren las superficies de los objetos con texturas para generar una imagen más realista.
- Sistema de Huesos en modelos 3D.- Un hueso es asignado a cada grupo de vértices, de este modo cuando movamos un hueso, se moverán todos los vértices que pertenezcan a ese grupo permitiendo una mejor manipulación del modelo 3D.
- Animación.- Se guardan las posiciones de uno o varios objetos en diferentes fotogramas para dar la sensación de movimiento de dichos objetos.
- Desarrollo de juegos en el sistema.- Se programan acciones por medio del lenguaje Python para el desarrollo de videojuegos en blender.
- Renderizado.- Se le llama así al proceso de generar una imagen o vídeo a partir de modelos 3D, tomando en consideración colores, texturas e iluminación.
- Detección de colisiones.- Los modelos 3D cuentan con colisionadores los cuales tienen como funcionalidad detectar cuando el objeto entra en contacto con otro.
- Simulación de comportamientos físicos.- Permite la simulación de algunos comportamientos de la física, como por ejemplo el comportamiento de un fluido, o de una pelota que rebota.

Interfaz de blender

En la figura 2.10 se presenta la interfaz de blender.

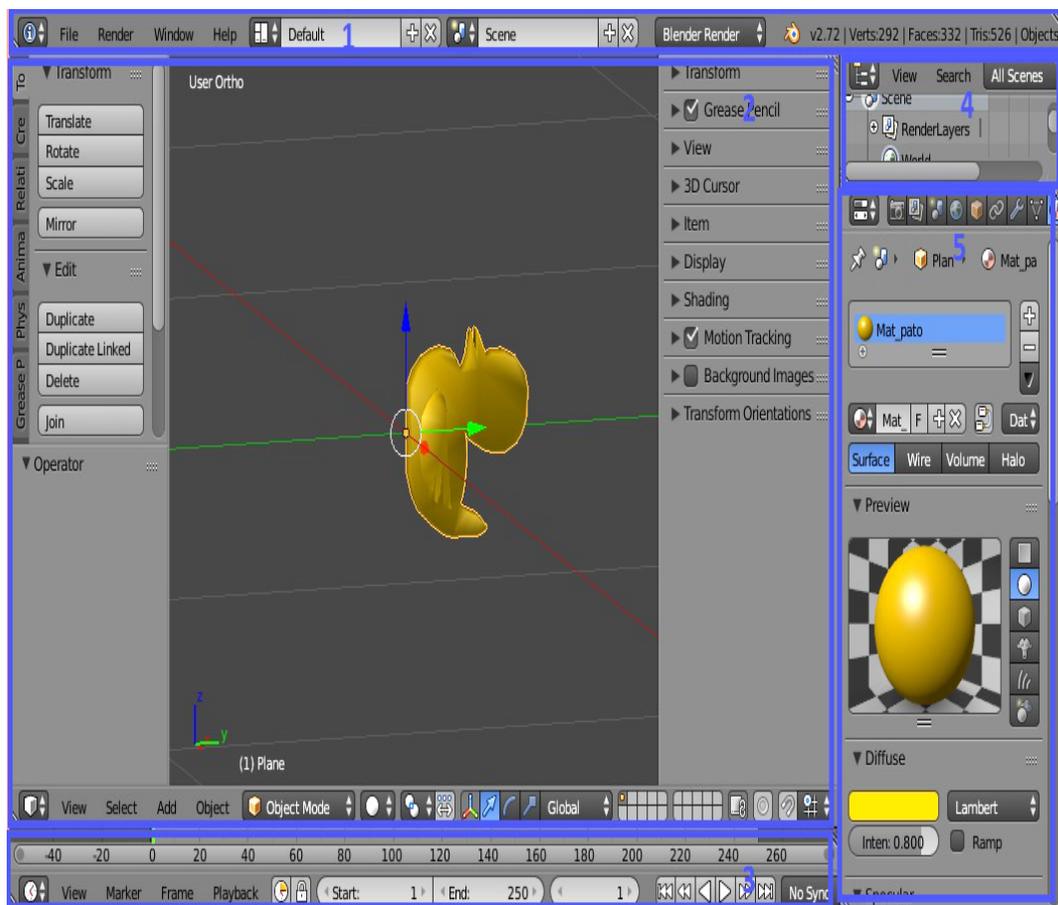


Figura 2.10: Vista general de la interfaz de Blender.

A continuación se describen las ventanas que componen la interfaz de blender.

1. Ventana Info- Esta ventana muestra la barra de menú principal, así como los mensajes de error generados en blender. También muestra toda la información sobre los modelos, como son: número de vértices, número de caras, el tamaño de memoria que utilizan y sus nombre, ver figura 2.11.



Figura 2.11: Ventana Info de Blender.

2. Ventana 3D view.- En esta ventana se visualiza el modelo 3D así como todas las herramientas que son utilizadas para su modificación, entre éstas se encuentran: herramientas de transformación como son: rotación, escalamiento, translación; herramientas de edición: duplicar, unir y borrar objetos; creación de figuras geométricas: planos, círculos, mallas, cilindros, conos, esferas; cambios de vista para el modelo 3D, así como selección por caras, aristas o vértices, ver figura 2.12.

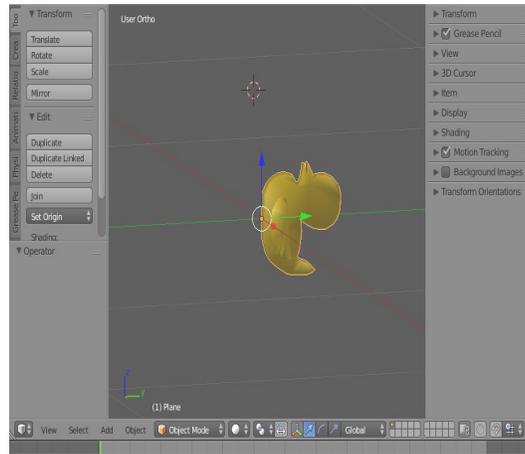


Figura 2.12: Ventana 3D view de Blender.

3. Ventana Timeline.- En esta ventana se muestran los diferentes fotogramas en los que puede ser configurado el modelo 3D para añadir animaciones a este, ver figura 2.13.

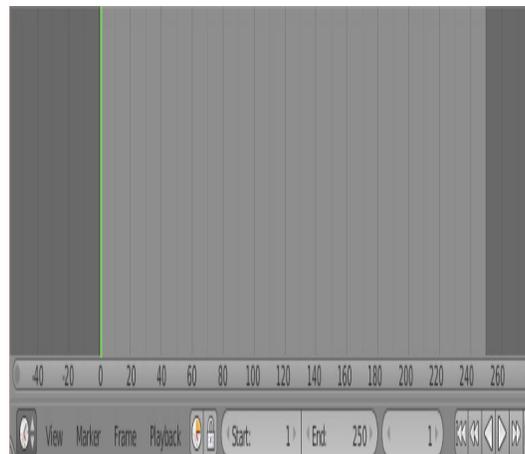


Figura 2.13: Ventana Timeline de Blender.

Ventana Outliner.- En esta ventana se muestran todos los objetos que existen en la escena, así como el nombre y material de cada uno de ellos, ver figura 2.14.

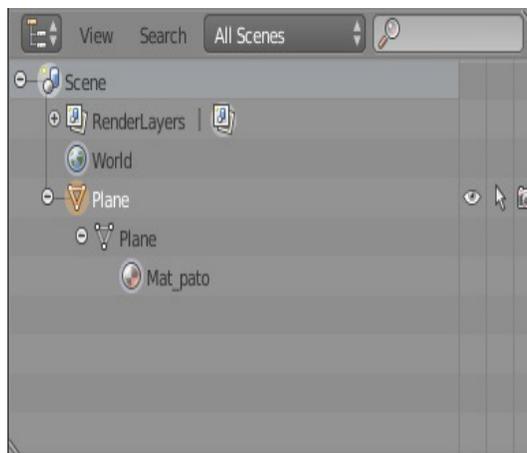


Figura 2.14: Ventana Outliner de Blender.

Ventana Properties.- En esta ventana (figura 2.15) se muestran las propiedades de los objetos seleccionados. Estas propiedades son: el tipo de material, texturas utilizadas, modificadores, grupo de vértices, entre otras.

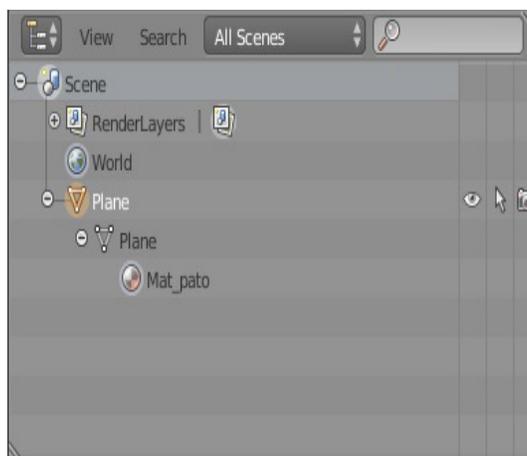


Figura 2.15: Ventana Properties de Blender.

2.2.3. GIMP

GIMP por sus siglas en inglés (GNU Image Manipulation Program) es un programa utilizado para la edición de imágenes digitales, estas imágenes pueden ser desde fotografías digitales hasta dibujos realizados por usuarios. Al igual que Blender, GIMP se encuentra bajo la licencia GNU, compartiendo las mismas características de modificación de éste. Así mismo, es un programa multiplataforma, estando disponible en las más utilizadas que son Linux, Windows y Mac OS X.

Debido a que es un software libre, ha ido actualizándose constantemente mejorando sus herramientas de edición y soportando muchos formatos de imágenes entre los cuales destacan: MPEG, PNG, JPG, Gif y BMP.

Su interfaz dispone de varios idiomas entre los que podemos encontrar: español, inglés, coreano, catalán, alemán, italiano, francés y ruso, entre otros.

GIMP permite la modificación de cada objeto en una imagen en forma independiente debido a que divide la imagen en capas facilitando su edición solamente en las partes de la imagen que nos interesan. Si guardamos la imagen en capas quedará con formato de .xcf para su futura edición en GIMP o se puede exportar en los formatos antes mencionados una vez que se termina la edición de ésta.

Entre las herramientas que utiliza GIMP encontramos las siguientes:

- Medición de longitud en cm y pulgadas.
- Creación y edición de capas.
- Menú para deshacer cambios realizados.
- Herramientas de pintado como: lápiz, pincel, relleno, etc.
- Herramientas de modificación de escala, perspectivas y de deformación.
- Herramientas de creación y edición de texto.
- Herramientas de creación y edición de colores.
- Herramientas de creación y edición de iluminación y sombras.
- Herramientas de selección de objetos por medio de colores, rectángulos, círculos o lazo manual.

Interfaz de GIMP

La interfaz de GIMP se muestra en la figura 2.16.

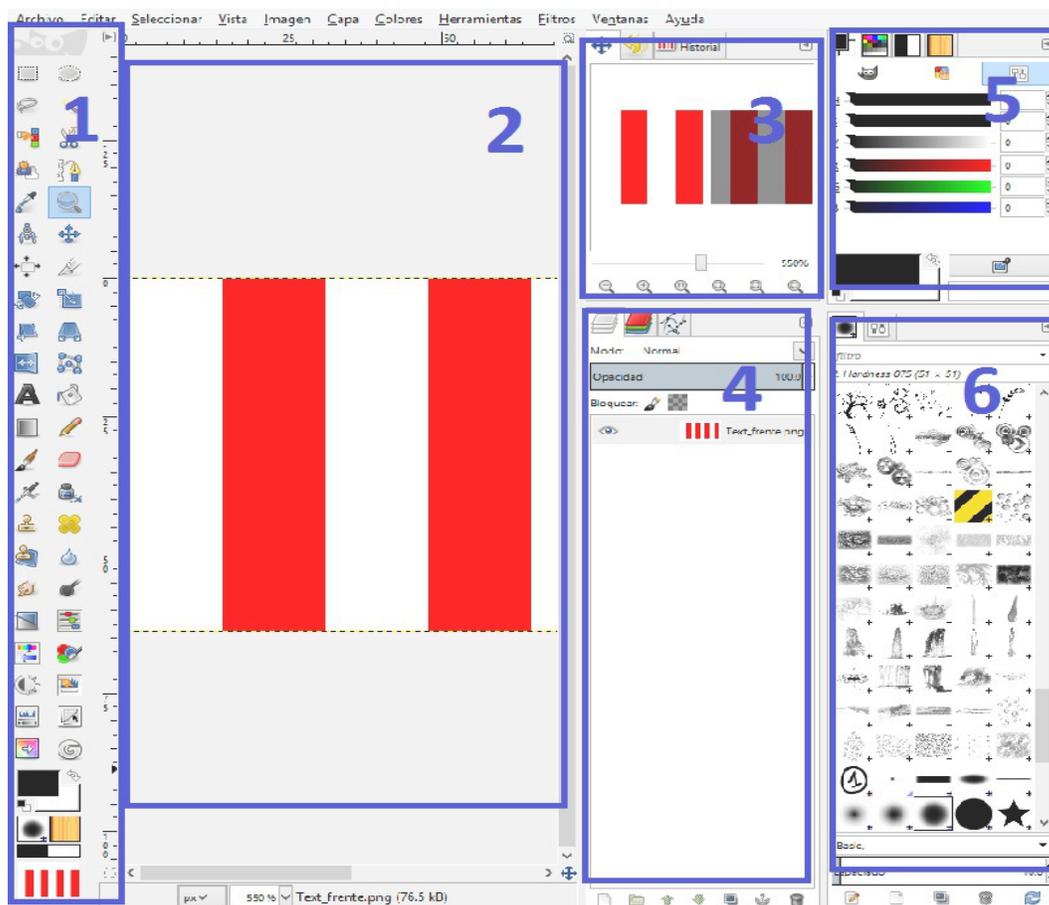


Figura 2.16: Vista general de la interfaz de GIMP.

A continuación se describen las ventanas que componen la interfaz de GIMP.

1. Caja de herramientas.- Esta ventana, contiene los iconos de herramientas que nos permiten llevar a cabo diferentes acciones sobre las imágenes. Entre estas acciones encontramos: opciones de modificación generales de imagen como son: recortar, cambiar tamaño, escalar, inclinar, rotar o cambiar perspectiva; selección de regiones como son: rectangular, elíptica, libre o por color; herramientas de texto; herramientas para pintar: lápiz, pincel, borrador y cubos, ver figura 2.17.



Figura 2.17: Caja de herramientas en GIMP.

2. Ventana de imagen.- En esta ventana se muestran las imágenes que se desean editar en GIMP. Alrededor de la imagen se muestran unidades de medida, estas pueden ser pixeles, milímetros, centímetros, metros, pulgadas, pies o yardas; permitiendo una mejor edición con medidas reales indicando coordenadas dentro de la imagen. Así mismo, si se hace click sobre las reglas y se arrastra el ratón sobre la imagen se puede crear líneas que servirán de guía para delimitar espacio de trabajo, ver figura 2.18.

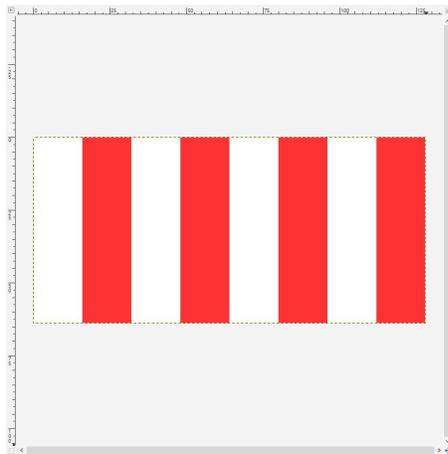


Figura 2.18: Ventana de imagen en GIMP.

3. Historial de cambios.- En la figura 2.19 se muestra la ventana de la información sobre las operaciones que se han llevado a cabo en la imagen.

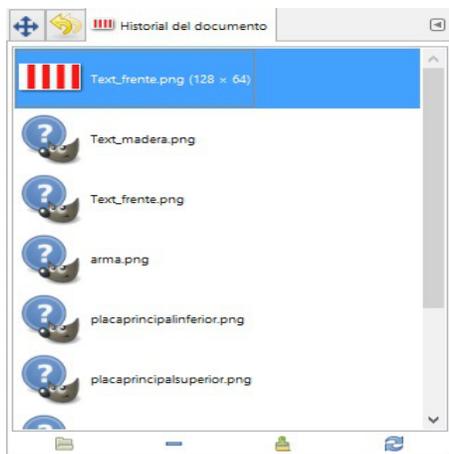


Figura 2.19: Ventana de historial de cambios en GIMP.

4. Ventana de capas y canales.- En esta ventana se muestran las diferentes capas y los canales de la imagen activa, como se puede ver en la figura 2.20. Aquí se selecciona la capa con la cual queremos trabajar, ocultando las demás, así mismo, se muestra el canal alfa de cada capa, el cual contiene información sobre que tan transparente es una capa en cada uno de sus pixeles.

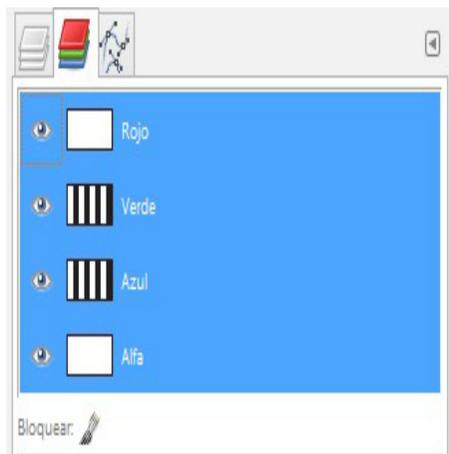


Figura 2.20: Ventana que muestra las capas y los canales en GIMP.

5. Ventana de patrones, degradados y paletas.- En esta ventanas se muestran las diferentes paletas de colores de donde se escoge un color solido con el cual pintar, ver figura 2.21. Así mismo, se encuentran patrones y degradados los cuales permiten aplicar distintos efectos a las imágenes.

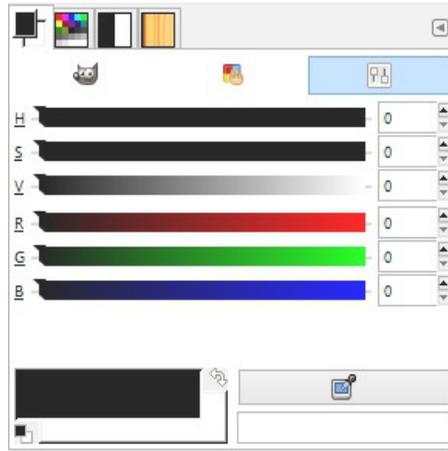


Figura 2.21: Ventana que muestra las paletas de colores, patrones y degradados en GIMP.

6. Pinceles y opciones de herramientas.- En esta ventana encontramos las diferentes formas de pinceles que pueden ser utilizadas en GIMP. Así mismo encontramos las opciones de herramientas, las cuales nos muestran todas las opciones existentes para la herramienta que se está utilizando en el momento de la selección, ver figura 2.22.

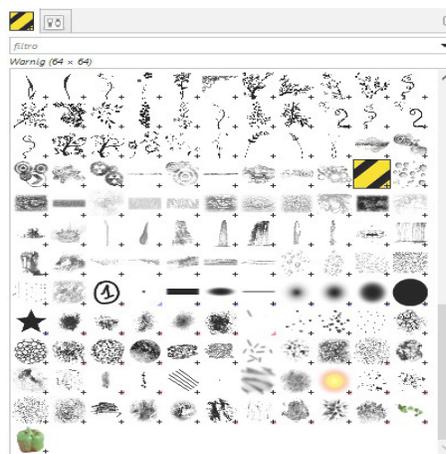


Figura 2.22: Ventana que muestra los diferentes tipos de pinceles en GIMP.

2.3. Elementos necesarios para inmersión en simuladores de RV

Las aplicaciones más comunes de los sistemas de RV son los simuladores, estos son de gran ayuda ya que permiten realizar todo tipo de pruebas sin que tenga consecuencias negativas, como pérdidas materiales o humanas. Por ejemplo; en aplicaciones militares se simula un combate aéreo, la ventaja de simulación en este caso sería ahorrar combustible al pilotear una nave por largas horas, en caso de que el piloto impactara con una montaña o fuera derribado por otro avión, solo terminaría la simulación pero en la vida real no le pasaría nada al usuario. Para que los simuladores de realidad virtual sean inmersivos requieren los siguientes elementos:

- Escenas 3D.- Se necesita un software que vaya cargando las escenas del ambiente virtual, las cuales están formadas de: geometrías, imágenes, personajes, etc.
- Navegación.- Es la forma en la cual el usuario se podrá mover a través de las escenas e ir interactuando con objetos de ellas.
- Colisiones.- Dentro de las escenas habrá objetos los cuales no podrán ser atravesados, por ejemplo una pared.
- Animaciones.- Estas se refieren a objetos que deben tener movimiento dentro de las escenas, ya sean personas, autos, etc.
- Física.- Para incrementar el realismo de un ambiente virtual es necesario que los objetos cuenten con físicas similares a las de objetos reales, ya sea la gravedad, su fricción, fuerza, etc.
- Inteligencia artificial.- Dentro del ambiente virtual los objetos también deben de tener algún tipo de inteligencia, por ejemplo un automóvil respetando algún semáforo o deteniéndose para no atropellar a personajes dentro de la escena.
- Sonidos.- Para mayor realismo debemos tener sonidos dentro del ambiente virtual, ya sea de personas hablando, perros ladrando, etc.
- Espacios envolventes.- Se necesita de cascos, cabinas, pantallas, etc., los cuales den la sensación de que la escena 3D envuelve al usuario.

Capítulo 3

Balística

El término balística (del griego *ballein*, "lanzar, arrojar") ha sido definido como la ciencia que estudia el movimiento, el alcance y la dirección de los proyectiles.

La balística es una disciplina compleja que se relaciona con otras ciencias como las matemáticas, la física y la química, ya que se encarga de estudiar el comportamiento del proyectil (bala) desde que éste se encuentra en un estado de reposo dentro del arma, inicia su movimiento a través del cañón de ésta y alcanza su velocidad máxima (necesaria para su desplazamiento a través del aire) al salir al exterior, hasta impactar con un cuerpo el cual al ser atravesado deja al proyectil nuevamente en estado de reposo.

La balística se divide en tres partes fundamentales [24],[18]:

1. Balística interior. Estudia los fenómenos que ocurren en el interior del arma, los cuales mediante liberación de energía, impulsan al proyectil a través del cañón hasta que traspasa la boca de esta. En este momento el proyectil adquirió la velocidad y el ángulo de inclinación correctos para impactar en el blanco deseado.
2. Balística exterior. Estudia la trayectoria del proyectil (trayectoria real y en el vacío), desde que éste sale de la boca del cañón hasta que finaliza su movimiento o penetra en el blanco. Su comportamiento depende de los siguientes factores:
 - La velocidad inicial.
 - La fuerza de gravedad.
 - La resistencia del aire.
3. Balística de efecto. Estudia el comportamiento del proyectil al impactar y atravesar el blanco al que va dirigido, transfiriéndole la energía cinética que posee en ese momento.

3.1. Balística interior

Como se mencionó anteriormente, se encarga de todos los fenómenos que ocurren dentro del arma, necesarios para poner en movimiento al proyectil. Esta acción se produce al transformar la energía generada por la expansión de los gases (producidos por la combustión de la pólvora) en energía cinética, la cual es utilizada para propulsar los proyectiles hacia blancos específicos. En la figura 3.1 se describen las partes de un cartucho.

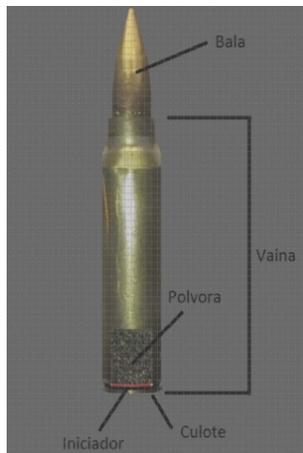


Figura 3.1: Partes de un cartucho.

En las armas de fuego actuales, la balística interna se divide en tres etapas de funcionamiento:

1. Accionamiento del disparador y percusión.- Cuando el gatillo (disparador) es presionado por el tirador, se libera un resorte (que se encontraba comprimido), desplazando rápidamente al percutor hacia el culote de la vaina.
2. Ignición.- Cuando el percutor impacta en el culote de la vaina, la mezcla fulminante es comprimida, generando una llamarada que se propaga en el iniciador, encendiendo todos los granos de pólvora, ocasionando su explosión.
3. Recorrido del proyectil en el ánima (espacio interior del cañón).- Los gases producidos por la explosión, impulsan al proyectil a lo largo del ánima experimentando una aceleración que lo hace adquirir su velocidad inicial, la cual es modificada al entrar a las estrías del cañón (figura 3.2), debido a que estas dotan al proyectil de un giro sobre su propio eje para adquirir mayor velocidad y estabilidad cuando este salga de la boca del cañón.

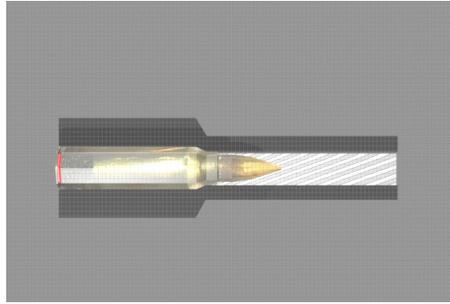


Figura 3.2: Anima con estrias.

3.2. Balística Exterior

En el momento en que el proyectil abandona la boca del cañón, inicia su trayectoria con una velocidad inicial la cual lo haría seguir en línea recta con dirección del eje del cañón como se muestre en la figura 3.3.

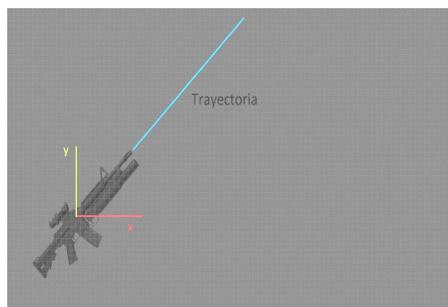


Figura 3.3: Trayectoria del proyectil en línea recta.

Siguiendo la trayectoria antes mencionada, el proyectil recorrería distancias iguales en tiempos iguales. Sin embargo, al salir al exterior es afectado por la fuerza de gravedad, la cual atrae al proyectil hacia el suelo a cada momento. Así mismo, si despreciamos la fuerza del viento y la resistencia del aire, se genera una trayectoria ideal, conocida como trayectoria en el vacío (de forma parabólica), ver figura 3.4.

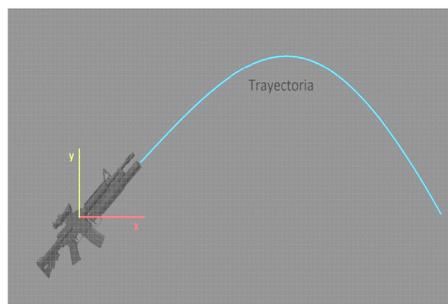


Figura 3.4: Trayectoria del proyectil (parábola).

3.2.1. Trayectoria en el vacío

Estudiando el comportamiento antes mencionado, tomamos la boca del arma como altura inicial y punto de partida, obteniendo los siguientes parámetros para su análisis matemático en un plano (X, Y) , figura 3.5.

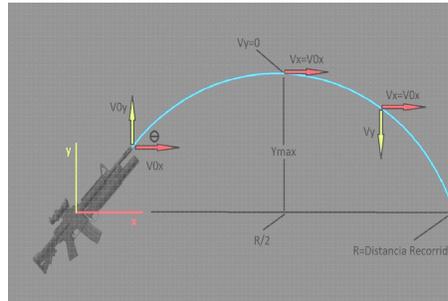


Figura 3.5: Parámetros establecidos a partir de la trayectoria parabólica.

Haciendo coincidir las coordenadas (X, Y) con las coordenadas de la trayectoria del proyectil, y tomando en cuenta la boca del arma como el origen. Tenemos:

$$\vec{r}_0 = (X_0, Y_0) \quad (3.1)$$

donde:

\vec{r}_0 = Vector de posición inicial.

$X_0 = 0$

Y_0 = Altura inicial.

$$\vec{V}_0 = (V_{0X}, V_{0Y}) \quad (3.2)$$

donde:

\vec{V}_0 = Vector de velocidad inicial.

V_0 = Velocidad inicial.

$V_{0X} = V_0 \cos \theta$

$V_{0Y} = V_0 \sin \theta$

$$\vec{a} = (a_X, a_Y) \quad (3.3)$$

donde:

\vec{a} = Vector de aceleración inicial.

$a_x = 0$

$a_y = -g$

La componente de la velocidad en x es constante, mientras que la de y cambia con el transcurso del tiempo. Debido a que la componente en y es afectada por la velocidad inicial y la gravedad, unimos estas dos velocidades para obtener la velocidad en cualquier tiempo t .

Integrando 3.3 y uniendo con 3.2, tenemos el vector de velocidad del proyectil:

$$\vec{V} = (V_X, V_Y) \quad (3.4)$$

donde:

$$V_X = V_0 \cos \theta$$

$$V_Y = V_0 \sin \theta - gt$$

Integrando 3.4 y uniendo con 3.1, tenemos el vector de posición del proyectil:

$$\vec{r} = (X, Y) \quad (3.5)$$

donde:

$$X = V_0 t \cos \theta$$

$$Y = Y_0 + V_0 t \sin \theta - \frac{1}{2}gt^2$$

Podemos determinar la altura máxima (Y_{max}), el alcance del proyectil (R) y el tiempo (t_{max}) en alcanzar la altura máxima en función de V_0 , g y θ . Debido a que en la altura máxima, el componente de la velocidad en y , es igual a cero, tomamos la ecuación correspondiente de 3.4 para determinar t_{max} :

$$V_Y = V_0 \sin \theta - gt_{max}$$

$$0 = V_0 \sin \theta - gt_{max}$$

$$gt_{max} = V_0 \sin \theta$$

$$\therefore \boxed{t_{max} = (V_0 \sin \theta) / g}$$

Tomando Y de 3.5 y el valor obtenido de t_{max} podemos determinar Y_{max} (altura máxima) en función de V_0 , θ .

$$Y_{max} = Y_0 + V_0 t_{max} \sin \theta - \frac{1}{2}gt^2$$

Pero:

$$t_{max} = (V_0 \sin \theta) / g \qquad t_{max}^2 = ((V_0 \sin \theta) / g)^2$$

Reemplazando tenemos:

$$Y_{max} = Y_0 + V_0 \sin \theta (V_0 \sin \theta / g) - \frac{1}{2} g (V_0 \sin \theta / g)^2$$

$$Y_{max} = Y_0 + V_0 \sin \theta (V_0 \sin \theta / g) - \frac{1}{2} g ((V_0)^2 \sin^2 \theta / g^2)$$

$$Y_{max} = Y_0 + ((V_0)^2 \sin^2 \theta / g) - ((V_0)^2 \sin^2 \theta / 2g)$$

$$\boxed{Y_{max} = Y_0 + \left(\frac{V_0^2 \sin^2 \theta}{2g} \right)}$$

El alcance R , es la distancia horizontal recorrida por el proyectil, se puede determinar con la componente en X del vector de posición en la ecuación 3.5 y tomando en cuenta el tiempo de vuelo (tiempo que permanece el proyectil en el aire). Utilizando la componente en Y del vector de posición en la ecuación 3.5:

$$Y = Y_0 + V_0 t_{vuelo} \sin \theta - \frac{1}{2} g (t_{vuelo})^2$$

Tomando la trayectoria de una parábola simétrica en el origen $(0, 0)$ y sabiendo que Y será igual a cero cuando el proyectil deje de estar en el aire, determinamos el tiempo de vuelo:

$$0 = 0 + V_0 t_{vuelo} \sin \theta - \frac{1}{2} g (t_{vuelo})^2$$

$$V_0 t_{vuelo} \sin \theta = \frac{1}{2} g (t_{vuelo})^2$$

$$\boxed{t_{vuelo} = \frac{(2V_0 \sin \theta)}{g}}$$

Utilizando la componente en X del vector de posición en 3.5 y reemplazando el tiempo de vuelo, obtenemos R :

$$R = V_0 t_{vuelo} \cos\theta$$

$$R = V_0 \frac{2V_0 \sin\theta}{g} \cos\theta$$

$$R = 2 \frac{(V_0)^2 \sin\theta \cos\theta}{g}$$

Utilizando la propiedad: $2\sin\theta \cos\theta = \sin 2\theta$

$$\therefore \quad \boxed{R = (\sin 2\theta V_0^2)/g} \quad \text{Alcance máximo horizontal}$$

Se realizó una simulación (figura 3.6) tomando en cuenta la velocidad inicial de 852m/s correspondiente a la carabina M4, con un ángulo de disparo de 45 grados, una altura inicial de 1.8 metros y tomando en cuenta la fuerza de gravedad pero sin considerar los efectos del viento, obteniendo los siguientes resultados:

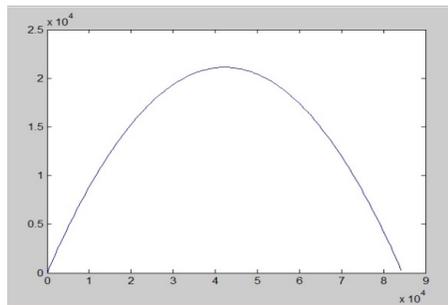


Figura 3.6: Trayectoria parabólica simulada.

$$R = 84500,0 \quad \text{metros.}$$

$$Y_{max} = 21127,0 \quad \text{metros}$$

$$t_{max} = 65,6599 \quad \text{segundos.}$$

3.2.2. Trayectoria con influencia atmosférica

En el estudio anterior se analizó la trayectoria de un proyectil en el vacío con condiciones ideales, siendo este afectado únicamente por la fuerza de gravedad. Sin embargo, para describir una trayectoria real se deben considerar una serie de factores los cuales afectan el movimiento del proyectil; dichos factores dependen de éste como son: el peso, el calibre, la geometría y el movimiento giroscópico con el que sale del arma; otros sin embargo, dependen del ambiente o medio en el que se desplaza como son: densidad, temperatura, viscosidad y la presión. Todos estos factores deben conjuntarse para obtener un tiro con gran precisión.

3.2.3. ¿Que es el Modelado?

Un modelo es una representación de sistemas o procesos, que tiene como fin analizarlos, describirlos y predecirlos. Los modelos dinámicos son aquellos que se obtienen por medio de un análisis de una situación física y que son descritos mediante ecuaciones diferenciales. Estas ecuaciones nos permiten conocer el comportamiento del sistemas permitiéndonos modificarlo.

La formulación de un modelo matemático comienza con la consideración de los factores que afectan al sistema. Debido a que analizaremos un sistema físico, necesitamos conocer los factores que conciernen al balance de fuerzas. En el movimiento de los elementos de un sistema se describen varias dimensiones las cuales son: traslación, rotación y una combinación de ambas.

Traslación

El cambio de posición de un proyectil se describe mediante el movimiento de traslación. Las variables que describen el movimiento de traslación son la aceleración, la velocidad y el desplazamiento. La ley del movimiento de Newton establece que *la suma algebraica de las fuerzas que actúan sobre un cuerpo rígido en una dirección dada es igual al producto de la masa por su aceleración en la misma dirección*, ver figura 3.7. Se puede expresar como:

$$\sum \text{fuerzas} = ma$$

$$\text{donde:} \begin{cases} m = \text{masa} \\ a = \text{aceleración} \end{cases}$$

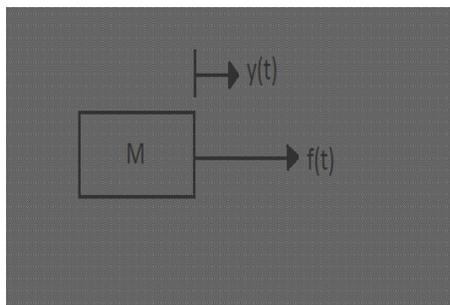


Figura 3.7: Sistema fuerza-masa.

La figura 3.7 muestra un sistema en el cual una fuerza $f(t)$ actúa sobre un cuerpo con una masa m describiendo la siguiente ecuación:

$$f(t) = ma(t) = m \frac{d^2 y(t)}{dt^2} \quad (3.6)$$

Fricción

Cuando el proyectil avanza a través del aire, debido a que este es un fluido, genera una fuerza de fricción la cual frena el desplazamiento del proyectil. A este tipo de fricción se le conoce como fricción viscosa y generalmente se representa como un amortiguador el cual se muestra en la figura 3.8:

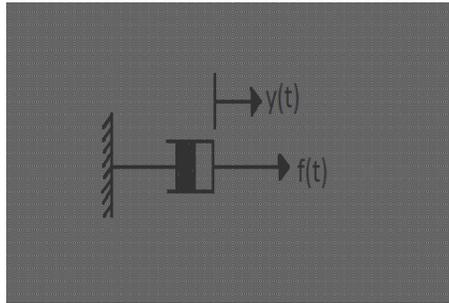


Figura 3.8: Amortiguador para fricción viscosa.

3.2.4. Modelado de la trayectoria de un proyectil

Para modelar el comportamiento de la trayectoria de un proyectil nos basamos en la ley de Newton descrita anteriormente, dada por la ecuación 3.6. Donde m es la masa del proyectil, y $f(t)$ son las fuerzas actuando sobre él, las cuales están compuestas por:

1. Fuerza de arrastre [12].- A lo largo del tiempo se han realizado modelos matemáticos de resistencia aerodinámica para describir la fricción del aire, el cual frena el desplazamiento de los proyectiles a lo largo de su trayectoria. La fuerza de arrastre depende de la forma del proyectil, así como de los factores climáticos, y esta definida por la siguiente ecuación:

$$F_d = \frac{1}{2} \rho A v^2 C_d \quad (3.7)$$

donde:

F_d = fuerza de arrastre, N .

ρ = densidad del aire, $\frac{kg}{m^3}$.

A = área de la sección transversal del proyectil, m^2 .

v = velocidad del proyectil, $\frac{m}{s}$.

C_d = coeficiente de arrastre (adimensional).

- a) Densidad del aire.- Desde 1982 el Comité Internacional de Pesas y Medidas (CIPM) ha publicado fórmulas para determinar la densidad del aire, dichas formulas han sido cambiadas y mejoradas constantemente. Entre los cambios mas significativos se encuentran los valores de las constantes de presión de saturación del vapor, el factor de fugacidad y la constante molar de los gases, modificándolos aproximadamente con un margen de 30×10^{-6} entre cada cambio de fórmula.

En el año 2008 se publicó la última fórmula por A. Picard, conocida como CIPM-2007. A diferencia de las anteriores, esta nueva fórmula incluye una evaluación de la incertidumbre, aplicando el método de la Guía para la Expresión de la Incertidumbre en Medición (GUM), dando como resultado de la fórmula una incertidumbre de aproximadamente 22×10^{-6} , la cual se toma en cuenta para las incertidumbres de presión atmosférica, humedad y temperatura.

La fórmula general para determinar la densidad del aire en [17] se describe a continuación:

$$\rho_a = \frac{pM_a}{ZRT} \left[1 - x_v \left(1 - \frac{M_v}{M_a} \right) \right] \quad (3.8)$$

donde:

ρ_a = densidad del aire, $\frac{Kg}{m^3}$.

p = presión atmosférica, Pa .

M_a = masa molar del aire seco, $\frac{kg}{mol}$.

Z = factor de compresibilidad.

R = constante molar de los gases, $\frac{J}{molK}$.

T = temperatura termodinámica del aire, K .

x_v = fracción molar del vapor de agua.

M_v = masa molar del agua, $\frac{Kg}{mol}$.

La guía del Sistema Interamericano de Metrología [15], presenta algunas simplificaciones de la fórmula general 3.8, en esta tesis se utilizó la versión exponencial debido a que en varios cálculos fue la que mostró resultados más precisos tomando en consideración diferentes situaciones climáticas. A continuación se muestra dicha fórmula:

$$\rho_a = \frac{0,34848p - 0,009h_r \exp(0,061t)}{273,15 + t} \quad (3.9)$$

donde:

ρ_a = densidad del aire, $\frac{Kg}{m^3}$.

p = presión atmosférica, hPa .

h_r = humedad relativa en %.

t = temperatura ambiente en C.

- b) Área de la sección transversal del proyectil.- El área es la superficie que se encuentra encerrada por una figura geométrica plana. Es por ello que para calcular el área del proyectil, se debe tomar una intersección de éste, que se encuentra en tres dimensiones, con un plano dado en dos dimensiones. Llamando a esta intersección sección transversal.

Al realizar un corte transversal al proyectil, obtenemos la figura geométrica de un círculo cuyo diámetro es igual al calibre de éste, dando como resultado la siguiente fórmula para calcular el área de la sección trasversal:

$$A = \frac{\pi \text{calibre}^2}{4} \quad (3.10)$$

donde:

π = Es el valor del cociente de la longitud de una circunferencia entre su diámetro. Su valor aproximado es 3.1416, con decimales infinitos.

calibre = Es el diámetro interno de un proyectil.

- c) Velocidad del proyectil.- Es la velocidad que lleva el proyectil a lo largo de su recorrido. Al inicio el proyectil alcanza su máxima velocidad al salir de la boca del cañón de arma, esta velocidad depende de los fenómenos ocurridos en el interior del arma.
- d) Coeficiente de arrastre.- El coeficiente de arrastre se define en función de la masa del proyectil, así como del tipo de material del que esta hecho y de su forma geométrica. Los conceptos relacionados con estas características son: densidad seccional y coeficiente balístico.
- Densidad seccional.- La densidad seccional es un factor muy importante, ya que permite al proyectil obtener un mayor alcance, debido a que toma las propiedades del material del cual esta hecho. Por ejemplo si disparamos dos proyectiles con las mismas dimensiones pero uno hecho de plástico y el otro

de plomo, sabemos que el de plomo tendría un mayor alcance e impactaría con mucho mas fuerza que el que esta hecho de plástico. Esto es debido a la densidad del material con que están hechos.

Así mismo, si disparamos dos proyectiles con el mismo diámetro y el mismo material, pero uno más largo que el otro, el que es más largo tendrá mayor densidad seccional que el corto, y esta mayor densidad le permitirá retener su velocidad por mayor tiempo. La densidad seccional se calcula mediante la siguiente fórmula matemática:

$$DS = \frac{m}{d^2} \quad (3.11)$$

donde:

DS = densidad seccional, $\frac{kg}{m^2}$.

m = masa del proyectil, kg .

d = calibre del proyectil, m .

- Coeficiente Balístico.- El coeficiente balístico (CB) es un valor adimensional que representa la capacidad de un proyectil para atravesar el aire, cuanto más grande sea este valor, el proyectil superará la resistencia del aire con mayor facilidad. Para determinar su valor se toma en cuenta la densidad seccional del proyectil, agregándole a esta, un factor de forma i , el cual es determinado por la punta del proyectil, favoreciendo a los puntiagudos y afectando a los de punta plana. La fórmula para calcular el coeficiente balístico es la siguiente:

$$CB = \frac{DS}{i} = \frac{m}{id^2} \quad (3.12)$$

donde:

CB = coeficiente balístico.

DS = densidad seccional.

i = factor de forma del proyectil.

- Modelo G.- A finales del siglo 19 se comenzó a buscar un parámetro estándar que sirviera de punto de referencia para la comparación de diferentes tipos de punta que tienen los proyectiles, tomando en cuenta también el tipo de geometría así como su peso. El primer estándar fue establecido por la comisión Gavre de la artillería naval francesa, de donde se tomo la letra “G” para establecer el modelo G1, el cual contenía características específicas para determinar el valor del coeficiente balístico.

Actualmente se fabrican diferentes tipos de puntas para los proyectiles, asignándoles diferentes perfiles balísticos a cada uno de ellos. Aún se sigue utilizando la letra “G” para dichos perfiles, sin embargo se han diseñado perfiles desde G1 hasta G8, estableciendo el coeficiente balístico en base a pruebas bajo condiciones controladas y dependiendo de las formas geométricas de los proyectiles.

La fórmula que relaciona el modelo G de los proyectiles con el coeficiente de arrastre es la siguiente:

$$C_g = \frac{C_d}{i} \quad (3.13)$$

donde:

C_g = coeficiente de arrastre referente al modelo G de los proyectiles.

Como se describió anteriormente la densidad seccional se obtiene a partir de la masa del proyectil y de su calibre, así mismo el coeficiente balístico agrega a esta configuración un factor de forma que dependerá de la punta del proyectil. Una forma de calcular el coeficiente de arrastre es relacionando los parámetros antes mencionados con las fórmulas 3.11, 3.12 y 3.13, obteniendo lo siguiente:

$$C_d = C_g \frac{m}{(CB)d^2} \quad (3.14)$$

El coeficiente de arrastre ayuda a la aerodinámica de los proyectiles y es calculado mediante la fórmula 3.14. De aquí el coeficiente balístico es definido por el fabricante o por el laboratorio de investigación en medición balística del Ejército de los Estados Unidos Americanos, es por ello que a pesar de que su valor es adimensional, esta determinado para trabajar con unidades de masa dadas en libras y diámetros en pulgadas. Para trabajar con unidades del sistema internacional, se debe redefinir el modelo G en unidades $\frac{kg}{m^2}$ añadiendo un factor de conversión de 0.0014223, obteniendo la fórmula del coeficiente de arrastre en unidades SI, mostrándose a continuación:

$$C_d = 0,0014223 C_g \frac{m}{(CB)d^2} \quad (3.15)$$

2. Fuerza gravitacional (gm), en donde g es la aceleración de la gravedad.

Modelo sin consideración del viento

Tomando en consideración que el proyectil se desplaza en un plano (x, y) en el que su componente en x representa la distancia recorrida, y su componente en y describe la altura alcanzada por el proyectil. Así mismo, las fuerzas que afectan al proyectil se descomponen en estas dos dimensiones. Si descartamos los efectos del viento, obtenemos como resultado el comportamiento mostrado en la figura 3.9:

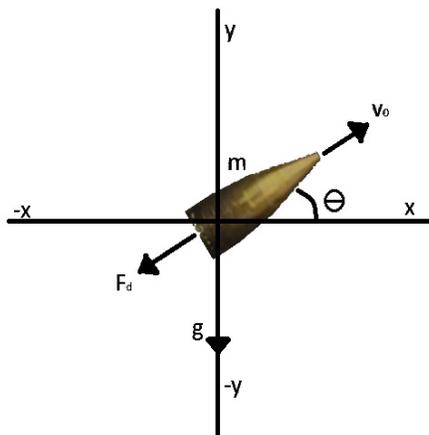


Figura 3.9: Fuerzas que afectan al proyectil.

En la figura 3.9 se muestran las fuerzas que influyen al movimiento del proyectil, observando que la velocidad inicial le da un empuje hacia adelante dependiente del ángulo θ (ángulo que existe entre el vector velocidad y el eje x), mientras que la fuerza de arrastre reduce su velocidad. Así mismo, la fuerza de gravedad lo atrae hacia el suelo, dando como resultado las siguientes ecuaciones:

$$a = -g \sin \theta - \frac{1}{2} \rho A v^2 C_d \quad (3.16)$$

$$\dot{\theta} = \frac{-g \cos \theta}{v} \quad (3.17)$$

$$\dot{x} = v \cos \theta \quad (3.18)$$

$$\dot{y} = v \sin \theta \quad (3.19)$$

Modelo considerando la acción del viento

Anteriormente se consideró que el proyectil viajaba en un plano (x, y) , en el que el eje x representa la distancia recorrida, mientras que el eje y representa la altura alcanzada por el proyectil. Tomando en cuenta las fuerzas que influyen en la trayectoria del proyectil se obtuvo el modelo matemático en la ecuación 3.16 que representa la aceleración del proyectil. A este modelo se le aplica una pequeña modificación para agregar la acción del viento, la cual se verá reflejada tanto en la dirección positiva o negativa del eje z como en el avance o retroceso del proyectil a través del eje x , descartando su efecto en el eje y , debido a que el viento no sopla de manera vertical. Tomando en consideración lo antes mencionado, obtenemos el siguiente comportamiento en la trayectoria del proyectil, el cual se ve afectado por la fuerza del viento, ver figura 3.10.

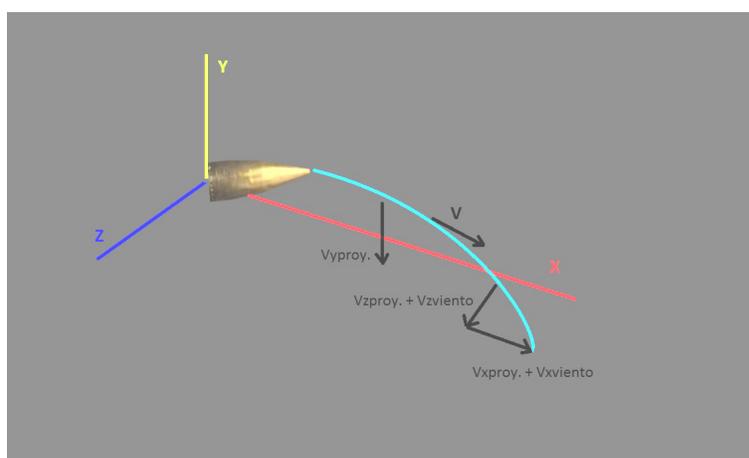


Figura 3.10: Trayectoria del proyectil afectada por el viento.

Así mismo, agregando la acción del viento a nuestro modelo matemático en 3.16, obtenemos las siguientes ecuaciones:

$$Ev = \frac{vel_p + vel_v}{V} \quad (3.20)$$

donde:

Ev = efecto del viento, $\frac{m}{s}$.

vel_p = velocidad del proyectil, $\frac{m}{s}$.

vel_v = velocidad del viento, $\frac{m}{s}$.

$$V = \sqrt{(velx_p + velx_v)^2 + vely_p^2 + (velz_p + velz_v)^2} \quad (3.21)$$

$$a = -g + \frac{1}{2m} \rho A v^2 C_d Ev \quad (3.22)$$

Capítulo 4

Visión por computadora

La visión por computadora es una rama de la ingeniería que incluye métodos para la adquisición, procesamiento, análisis y extracción de información de imágenes. Este conjunto de imágenes son tomadas de un mundo real mediante una cámara, y se le extraen sus propiedades, destacando e ignorando ciertas características para su análisis en una computadora. Algunas aplicaciones de la visión por computadora pueden ser localización de objetos en imágenes digitales, reconocimiento de personas, evaluación de calidad de piezas de ensamblaje, identificación de objetos, etc. En este capítulo se presenta la aplicación de técnicas de visión por computadora para la detección de un punto laser que es emitido en el mundo real y mapeado de un espacio real a un espacio virtual. Cuando las imágenes capturadas por la cámara son enviadas a las computadora, ésta las procesa de manera bidimensional debido a que la pantalla en donde las visualizamos es rectangular, donde cada uno de sus elementos es conocido como píxel, almacenando cada uno, un valor concreto y creando una matriz de puntos de dimensiones $M \times N$. Un ejemplo del procesamiento de imagen se muestra en la figura 4.1, en la cual se trata de identificar en una imagen los píxeles que pertenecen a los objetos mediante la segmentación.

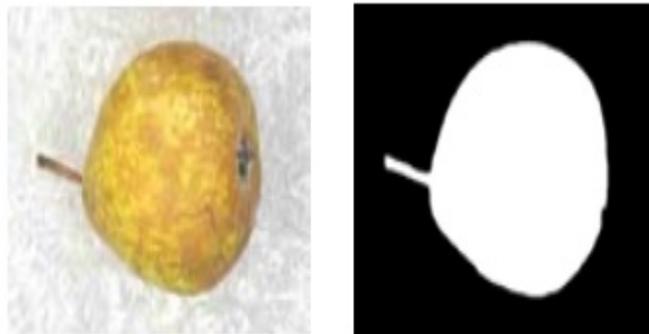


Figura 4.1: Procesamiento de imagen mediante segmentación [16].

4.1. Etapas del procesamiento de imágenes

Para extraer la información requerida de las imágenes, es necesario aplicar técnicas que mejoran la calidad de la imagen original. Estas técnicas consisten en etapas para el procesamiento de imágenes, las cuales se muestran en la figura 4.2.

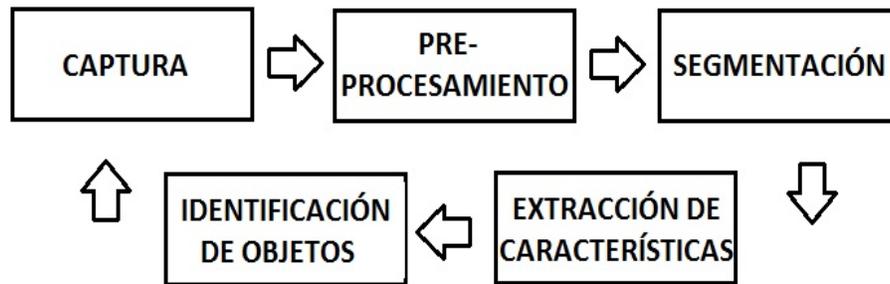


Figura 4.2: Etapas del procesamiento de imágenes [20].

- Captura (Adquisición de la imagen).- En esta etapa se obtiene la imágenes o secuencias de imágenes desde una cámara (ver figura 4.3) en el mundo real y enviadas a una computadora para su análisis bidimensional.



Figura 4.3: Cámara V120-SLIM.

- Preprocesamiento.- En esta etapa se eliminan ruidos en la imagen por medio de filtros o se aumenta el contraste, para extraer solo la información deseada. Se utilizó el plugin AVProLiveCamera para tomar las imágenes de la cámara, convertirlas en textura y asignarlas a un plano 2D dentro de Unity.
- Segmentación.- En esta etapa se subdivide la imagen en regiones(grupos de píxeles). La segmentación debe detenerse cuando la representación de la imagen es más fácil de analizar. En la segmentación se le asigna una etiqueta a cada píxel, dando como resultado conjuntos de píxeles que compartan características similares, como pueden ser el color, la intensidad o la textura. Los algoritmos de segmentación se basan en dos propiedades básicas de los valores de nivel de gris:

- Discontinuidad.- Consiste en dividir la imagen a partir de los cambios en los niveles de gris entre los pixeles. Las técnicas que utilizan las discontinuidades son:
 - Detección de puntos aislados
 - Deteccion de lineas
 - Deteccion de bordes
- Similitud .- Consiste en dividir la imagen y agrupar los pixeles que contengan características similares, conforme a criterios previamente asignados. Las técnicas que utilizan la similitud son:
 - Umbralizacion
 - Crecimiento de regiones

Extracción de características.- En esta etapa se seleccionan y extraen las características apropiadas para la identificación de los objetos deseados.

Identificación de objetos.- En esta etapa se realiza una clasificación de los objetos identificados, uniendo a los que comparten las mismas características en el mismo conjunto.

4.2. Umbralización

Uno de los métodos mas importantes de la segmentación es el de la umbralización. Este método consiste en convertir una imagen en escala de grises a una nueva con solo dos niveles mediante una simple binarización. Estos niveles se configuran a partir de umbrales(límites), convirtiendo en negro a todos los pixeles con nivel de gris menor al valor de umbral seleccionado y convirtiendo en blanco los que son menores. Generando una separación entre los objetos de interés y un fondo de la escena en un mundo virtual. En esta tesis se utiliza una cámara V120-SLIM la cual utiliza un filtro físico, que tiene como objetivo obstruir la luz del ambiente real, permitiendo obtener un fondo oscuro, ensombreciendo las imágenes obtenidas por la cámara. Una vez obtenido el fondo oscuro, asignamos un valor de umbral que esta normalizado, por lo que varia de 0 a 1 y que servirá para binarizar la imagen separando a los dos conjuntos. En este caso, el primer conjunto se refiere al fondo oscuro y el segundo conjunto será el haz de luz de un laser. De este modo, cualquier píxel que cumpla la condición $\text{píxel} > \text{umbral}$, pertenecerá al haz de luz de un laser, en caso contrario pertenecerá al fondo.

4.2.1. ¿Qué es un láser?

Un láser (Light Amplification by Stimulated Emission of Radiation) por su traducción al español: amplificación de luz por emisión estimulada de radiación, es un dispositivo que utiliza la emisión inducida, para generar un haz de luz capaz de concentrar la emisión en un rango espectral muy estrecho.

Si calentamos un objeto a 200 C podremos notar al acercar la mano a éste que emite cierta radiación invisible llamada calor o radiación infrarroja [2]. Así mismo si se sigue aumentando la temperatura esa radiación ira tomando progresivamente color tenue rojo, amarillo y azul. en la figura 4.4 se muestra la radiación emitida por longitud de onda para varias temperaturas, en donde se observa que a medida que aumenta la temperatura, se emite mas energía en forma de radiación.

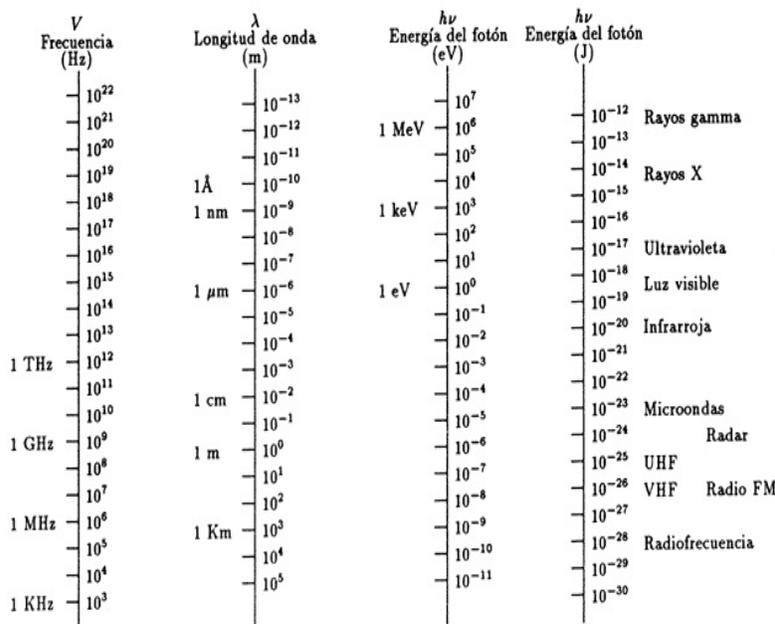


Figura 4.4: Radiación emitida por longitud de onda para varias temperaturas.

Las características del laser que se tomaron en cuenta para el uso de éste, son las siguientes:

Sin embargo, la luz procedente de un láser se diferencia de la de una linterna en cuatro aspectos básicos:

1. El láser es capaz de emitir un haz de luz con un diámetro milimétrico concentrando su energía en un solo punto.
2. La luz que es emitida por el láser no se dispersa fuera de su diámetro de éste.
3. El haz de luz del láser es continuo generándose siempre en el mismo sitio en donde se esta apuntado.
4. El láser produce luz de un solo color, sin embargo permite generar distintos colores tanto visibles como invisibles para el ojo humano.

4.2.2. Representación del punto laser

Debido a pruebas realizadas, en cuanto a luz y reflejos en el cuarto de pruebas, se estableció un umbral adimensional de 0.075. En la etapa de umbralización se realiza una separación entre el fondo y objetos, en este caso un punto laser. Sin embargo, debido a que se toman todos los píxeles de la imagen obtenida, el punto laser es leído en varios píxeles contiguos. Es por ello que se debe localizar el centroide del conjunto de dichos píxeles, determinado por las siguientes ecuaciones:

$$C.x = \frac{\sum posx}{w} \quad (4.1)$$

donde:

$C.x$ = coordenada del centroide en el eje x.

$posx$ = coordenada de un píxel encontrado en el eje x.

w = ancho de la imagen representado en píxeles.

$$C.y = \frac{\sum posy}{h} \quad (4.2)$$

donde:

$C.y$ = coordenada del centroide en el eje y.

$posy$ = coordenada de un píxel encontrado en el eje y.

h = alto de la imagen representado en píxeles.

4.3. Mapeo Espacio cámara - Mundo virtual

Para realizar un mapeo del espacio de la cámara a un mundo virtual se debe realizar una calibración tomando en cuenta dos áreas de trabajo utilizadas en la adquisición y procesamiento de la imagen obtenida por la cámara V120 (figura 4.3). Estas se describen a continuación en la figura 4.5:



Figura 4.5: Áreas de trabajo utilizadas en el procesamiento de imagen con Optitrack V120.

1. Área de proyección de la ventana de la aplicación.- Esta área representa la ventana de la aplicación la cual es emitida mediante un proyector hacia la pared de donde la cámara adquiere las imágenes.
2. Área de visión de la cámara V120.- Ésta área representa las imágenes obtenidas por la cámara, la cual debe ser mayor a la proyección de la ventana de la aplicación.

Dentro de la aplicación de Unity 3D, se coloca un plano que sea del mismo tamaño que la ventana donde se mostrará la aplicación, en este caso el tamaño de la imagen producida por el proyector. Una vez colocado este plano, se le aplica una textura la cual mediante código representa una umbralización, mostrando de color azul el fondo y de color blanco un punto laser detectado por la cámara, como se muestra en la figura 4.6.

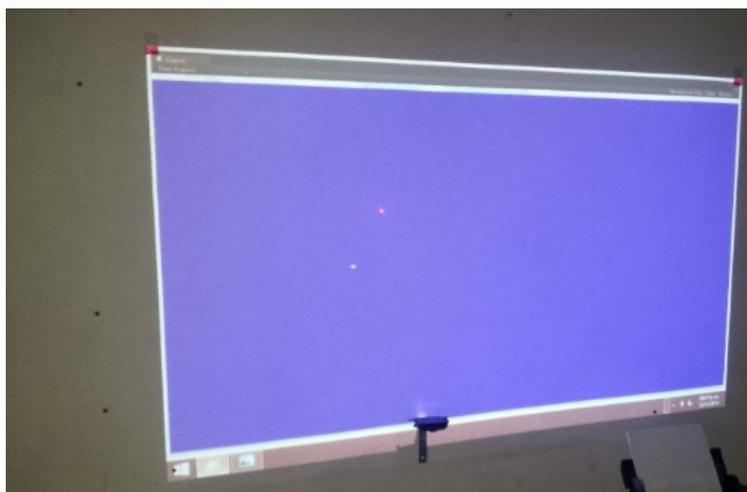


Figura 4.6: Plano umbralizado, mostrando de color azul el fondo y de blanco un punto laser.

4.3.1. Distorsión de imagen

La distorsión de imagen conocida por sus siglas en inglés como *Image Warping*, es la acción de transformar una imagen a otra con diferente perspectiva, mapeando las coordenadas de la imagen original a otras de un espacio deseado. El mapeo usualmente esta definido por las funciones $x(u, v)$ y $y(u, v)$, y es usado en el procesamiento de imagen para remover distorsiones de las imágenes que generalmente son obtenidas mediante una computadora. En la figura 4.7 se observa la imagen generada por el proyector, la cual se encuentra distorsionada debido a que éste se encuentra girado y desplazado por lo que se le aplica una transformación para obtener una imagen en un rectángulo uniforme y normalizado. Una vez realizada la transformación de la imagen, cuando un punto laser sea detectado en el centro de la imagen distorcionada, este se proyectara en el centro de la imagen normalizada.

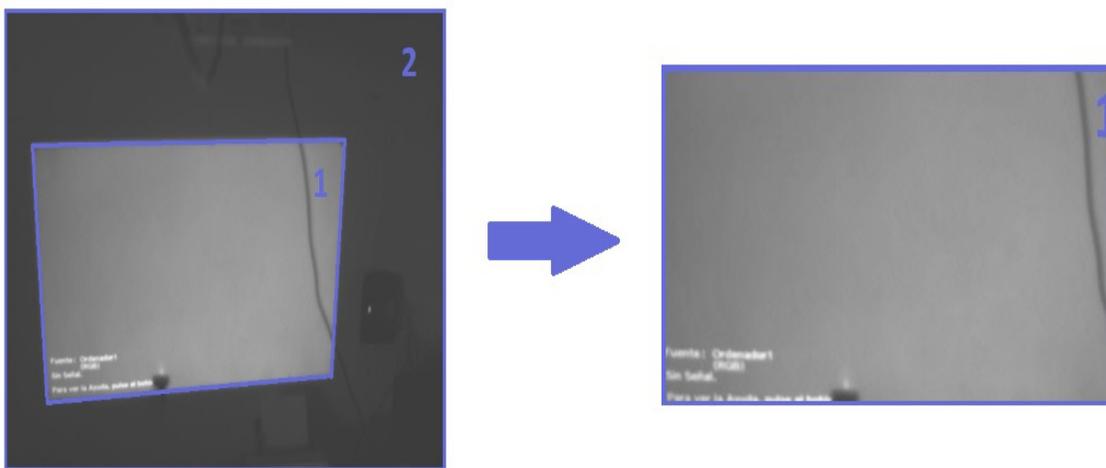


Figura 4.7: Transformación del área deforme a un rectángulo uniforme y normalizado.

4.3.2. Matriz de transformación de perspectiva

Como podemos observar anteriormente en la figura 4.6, el punto laser detectado por la cámara no coincide en la misma posición que el punto creado en la escena de la aplicación. Esto se debe a que la cámara visualiza el área de proyección como un área deforme debido a que ésta toma los puntos de un plano en el mundo real y los mapea a un plano proyectivo [23], es por ello que se realiza la correcta calibración para que la cámara transforme esa área deforme a un área rectangular uniforme.

Se define el modelo de la cámara [1] como $V_2 = MV_1$, en donde V_2 es el vector de coordenadas del mundo real, V_1 es el vector de coordenadas del plano que se desea obtener y M es la matriz de transformación para 3 dimensiones.

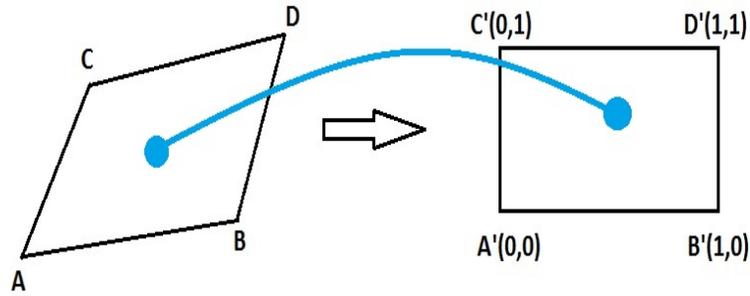


Figura 4.8: Coordenadas utilizadas para el modelo de la cámara.

A partir de las coordenadas de la figura 4.8, podemos escribir el modelo de la cámara de la siguiente forma:

$$\begin{pmatrix} A'_1 W \\ A'_2 W \\ 1W \end{pmatrix} = \begin{pmatrix} m_{00} & m_{01} & m_{02} \\ m_{10} & m_{11} & m_{12} \\ m_{20} & m_{21} & 1 \end{pmatrix} \cdot \begin{pmatrix} A_1 \\ A_2 \\ 1 \end{pmatrix}$$

Como podemos observar, $W = m_{20}A_1 + m_{21}A_2 + 1$, permitiendo reescribir el modelo de la cámara de la siguiente forma:

$$\begin{pmatrix} A'_1 \\ A'_2 \\ 1 \end{pmatrix} = \frac{\begin{pmatrix} m_{00} & m_{01} & m_{02} \\ m_{10} & m_{11} & m_{12} \\ m_{20} & m_{21} & 1 \end{pmatrix} \cdot \begin{pmatrix} A_1 \\ A_2 \\ 1 \end{pmatrix}}{\begin{pmatrix} m_{20} & m_{21} & 1 \end{pmatrix} \cdot \begin{pmatrix} A_1 \\ A_2 \\ 1 \end{pmatrix}}$$

Debido a que las coordenadas que se desean obtener son para un plano normalizado y tomando en cuenta el modelo de la cámara, se obtienen las siguientes ecuaciones:

$$0 = \frac{m_{00}A_1 + m_{01}A_2 + m_{02}}{m_{20}A_1 + m_{21}A_2 + 1}$$

$$0 = \frac{m_{10}A_1 + m_{11}A_2 + m_{12}}{m_{20}A_1 + m_{21}A_2 + 1}$$

$$1 = 1$$

$$1 = \frac{m_{00}B_1 + m_{01}B_2 + m_{02}}{m_{20}B_1 + m_{21}B_2 + 1}$$

$$0 = \frac{m_{10}B_1 + m_{11}B_2 + m_{12}}{m_{20}B_1 + m_{21}B_2 + 1}$$

$$1 = 1$$

$$0 = \frac{m_{00}C_1 + m_{01}C_2 + m_{02}}{m_{20}C_1 + m_{21}C_2 + 1}$$

$$1 = \frac{m_{10}C_1 + m_{11}C_2 + m_{12}}{m_{20}C_1 + m_{21}C_2 + 1}$$

$$1 = 1$$

$$1 = \frac{m_{00}D_1 + m_{01}D_2 + m_{02}}{m_{20}D_1 + m_{21}D_2 + 1}$$

$$1 = \frac{m_{10}D_1 + m_{11}D_2 + m_{12}}{m_{20}D_1 + m_{21}D_2 + 1}$$

$$1 = 1$$

Reagrupando las ecuaciones anteriores, tenemos:

$$\begin{aligned} m_{00}A_1 + m_{01}A_2 + m_{02} &= 0 \\ m_{10}A_1 + m_{11}A_2 + m_{12} &= 0 \\ m_{00}B_1 + m_{01}B_2 + m_{02} - m_{20}B_1 - m_{21}B_2 &= 1 \\ m_{10}B_1 + m_{11}B_2 + m_{12} &= 0 \\ m_{00}C_1 + m_{01}C_2 + m_{02} &= 0 \\ m_{10}C_1 + m_{11}C_2 + m_{12} - m_{20}C_1 - m_{21}C_2 &= 1 \\ m_{00}D_1 + m_{01}D_2 + m_{02} - m_{20}D_1 - m_{21}D_2 &= 1 \\ m_{10}D_1 + m_{11}D_2 + m_{12} - m_{20}D_1 - m_{21}D_2 &= 1 \end{aligned}$$

Reescribiendo el modelo de la cámara obtenemos la siguiente matriz, la cual nos permite la transformación del área de trabajo vista por la cámara, a un rectángulo uniforme con valores normalizados para su ancho y largo los cuales nos permiten la detección de un punto laser en un espacio cámara y mapearlo a un mundo virtual utilizado en la aplicación.

$$\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} A_1 & A_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & A_1 & A_2 & 1 & 0 & 0 \\ B_1 & B_2 & 1 & 0 & 0 & 0 & -B_1 & -B_2 \\ 0 & 0 & 0 & B_1 & B_2 & 1 & 0 & 0 \\ C_1 & C_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_1 & C_2 & 1 & -C_1 & -C_2 \\ D_1 & D_2 & 1 & 0 & 0 & 0 & -D_1 & -D_2 \\ 0 & 0 & 0 & D_1 & D_2 & 1 & -D_1 & -D_2 \end{pmatrix} \cdot \begin{pmatrix} m_{00} \\ m_{01} \\ m_{02} \\ m_{10} \\ m_{11} \\ m_{12} \\ m_{20} \\ m_{22} \end{pmatrix}$$

Esta matriz es evaluada en unity 3D mediante el script de calibración, convirtiendo el área deforme en una área rectangular uniforme (ver figura 4.7), tomando como parámetros las 4 esquinas del área de trabajo. Para ello se coloca el laser en cada una de las 4 esquinas del plano como se muestra en la figura 4.9.

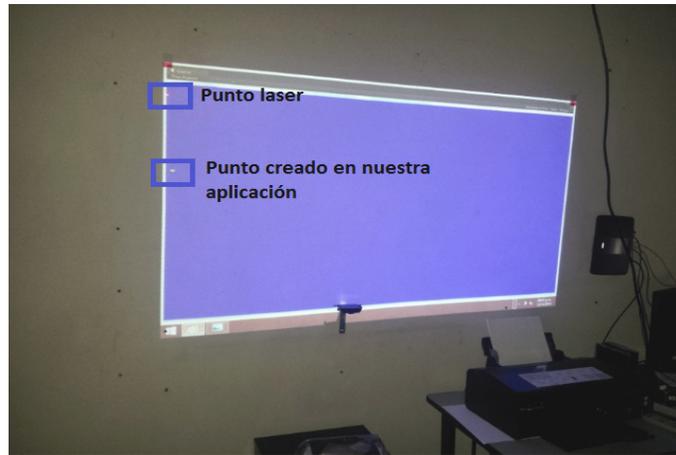


Figura 4.9: Calibrando una de las 4 esquinas del plano umbralizado.

Capítulo 5

Integración

5.1. Integración de las ecuaciones de Balística en Unity 3D

En el capítulo 3 se obtuvo el modelo matemático para determinar la trayectoria de un proyectil, tomando en cuenta condiciones atmosféricas como son: presión, humedad, temperatura, etc., así como características del proyectil como son: calibre, tipo de punta, peso, etc. Este modelo fue simulado obteniendo valores muy cercanos a la realidad.

En este capítulo se programa dicho modelo, realizando sus cálculos y asignación de variables mediante Unity 3D. Se probó en modo edición para trayectorias en 2D (ver figura 5.1) y 3D (ver figura 5.2), obteniendo resultados (ver figura 5.3), que son muy similares a los obtenidos anteriormente en el capítulo 3, diferenciando por máximo 4 metros a más de 900 metros de distancia.

Resultados obtenidos por simulación en el capítulo 3:

$$\begin{aligned}X_{max} &= 940,90 \quad \text{metros.} \\Y_{max} &= 561,82 \quad \text{metros} \\Z_{max} &= 96,90 \quad \text{metros}\end{aligned}$$

Resultados obtenidos en Unity 3D en modo edición:

$$\begin{aligned}X_{max} &= 937,55 \quad \text{metros.} \\Y_{max} &= 554,82 \quad \text{metros} \\Z_{max} &= 97,18 \quad \text{metros}\end{aligned}$$

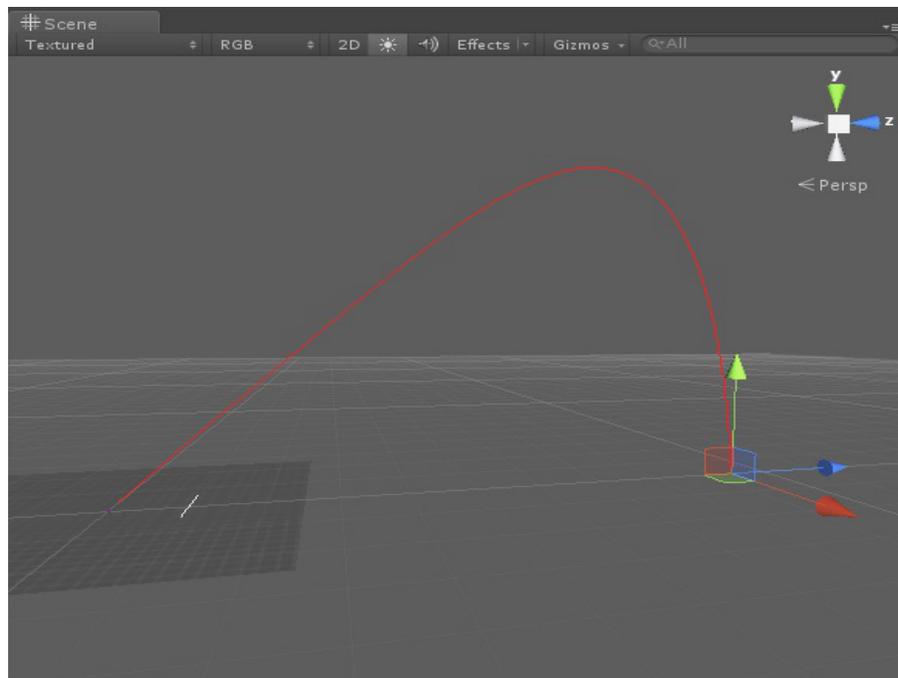


Figura 5.1: Trayectoria 2D simulada en Unity 3D en modo edición.

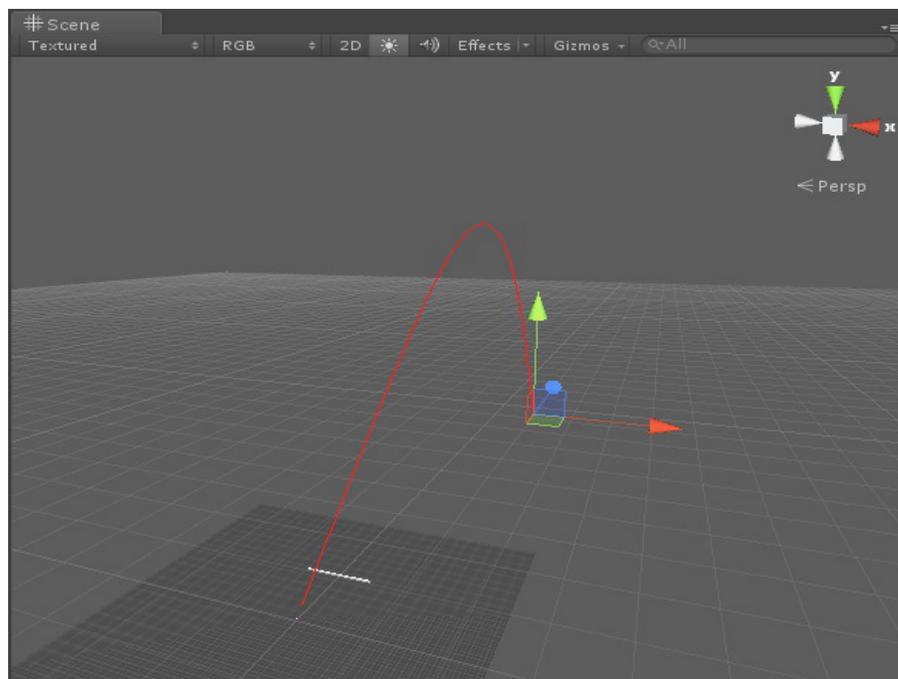


Figura 5.2: Trayectoria 3D simulada en Unity 3D en modo edición.



Figura 5.3: Configuración de parámetros para trayectoria 3D en Unity 3D (modo edición).

Para interactuar con la aplicación, ésta debe ser ejecutada en modo PLAY (ver figura 5.4), debido a esto se integraron las ecuaciones de balística en la escena, tomando los parámetros necesarios para su correcto funcionamiento. Dando los siguientes resultados (ver figura 5.5):

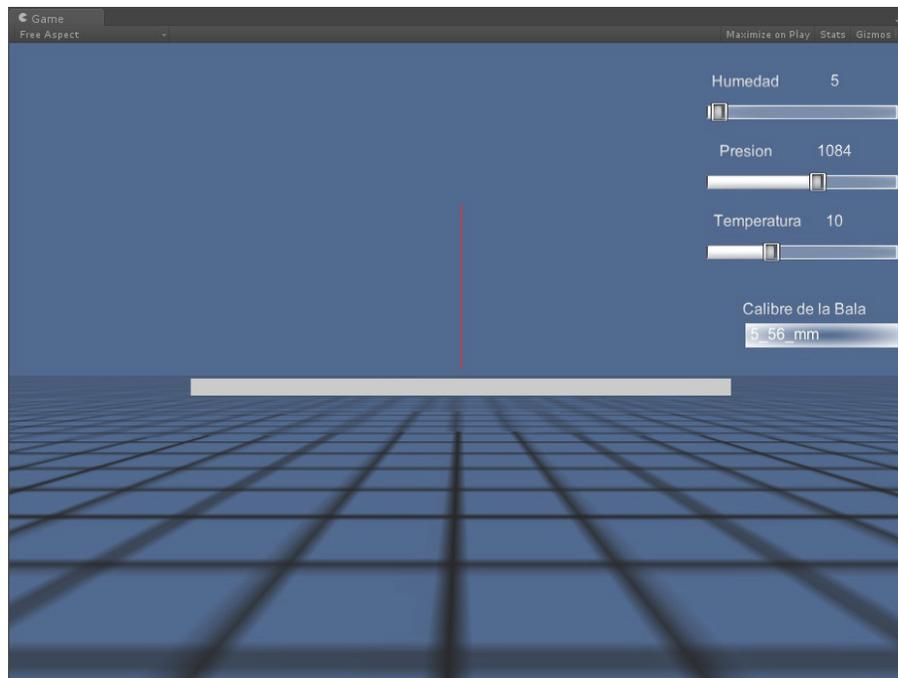


Figura 5.4: Trayectoria 2D simulada en Unity 3D en modo PLAY.

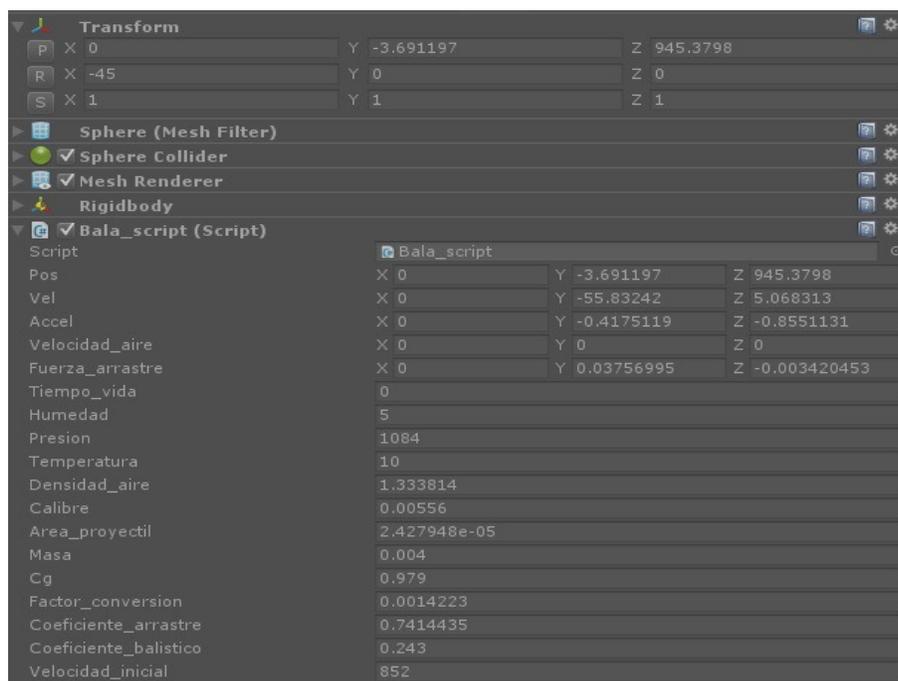


Figura 5.5: Configuración de parámetros para Trayectoria 2D en Unity 3D (modo PLAY).

5.2. Modelos 3D

Se realizaron modelos 3D (ver figuras: 5.6, 5.8 y 5.10) en la interfaz de Blender mencionada en el capítulo 2, y se importaron a Unity 3D (ver figuras: 5.7, 5.9 y 5.11) para ser agregados a la escena de nuestra aplicación. A continuación se muestran las imágenes de los modelos en Blender e importados en Unity 3D:

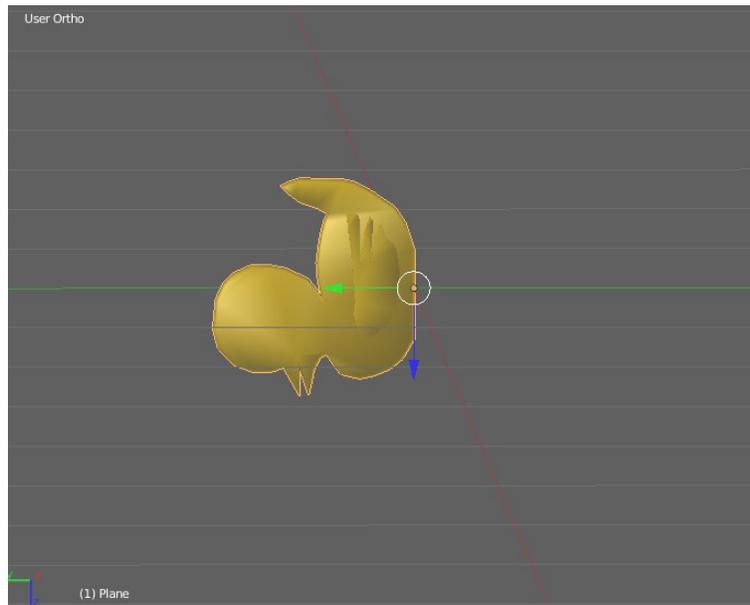


Figura 5.6: Modelo de un pato realizado en Blender.

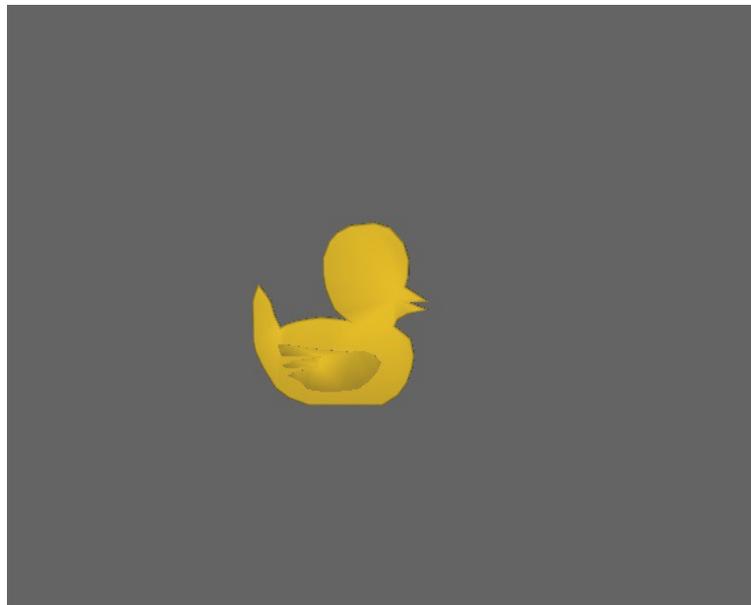


Figura 5.7: Modelo de un pato realizado en Blender e importado a Unity 3D.

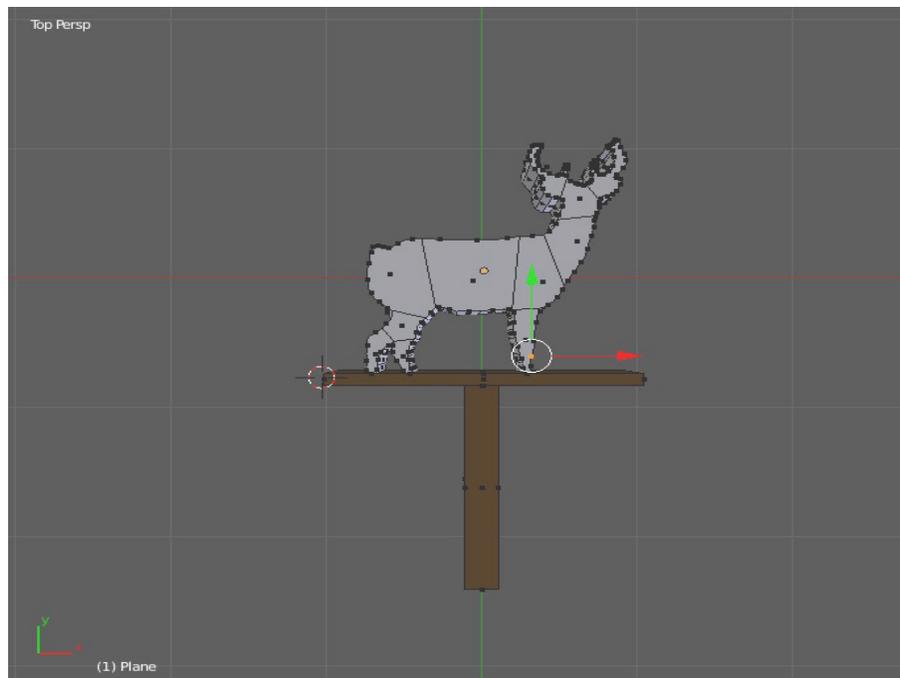


Figura 5.8: Modelo de un venado realizado en Blender.

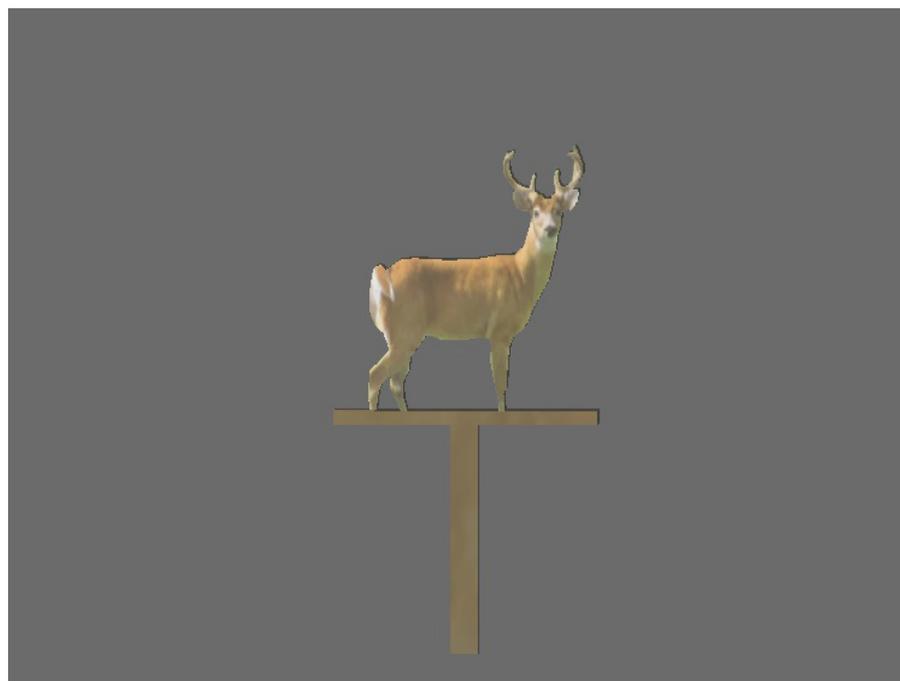


Figura 5.9: Modelo de un venado realizado en Blender e importado a Unity 3D.



Figura 5.10: Modelo de una casa de tiro realizado en Blender.



Figura 5.11: Modelo de una casa de tiro realizado en Blender e importado a Unity 3D.

5.3. Escenas Unity 3D

En la aplicación de Unity 3D se programaron 3 escenas en las que se integraron los modelos 3D antes mencionados.

1. La primera escena es de la calibración de la cámara (ver figura 5.12). Como se explicó en el capítulo 4, consta de la adquisición de imagen de la cámara Optitrack, detección de un punto laser por medio del método de umbralización, y calibración de la ventana de proyección con la detección del punto laser.



Figura 5.12: Escena de Unity para calibración de la cámara Optitrack.

2. La segunda escena que se programó, fue una casa de tiro (ver figura: 5.13 y 5.14), en la que se encuentran varios modelos 3D de patos a los cuales se les dispara con el punto laser y deben ser derribados.

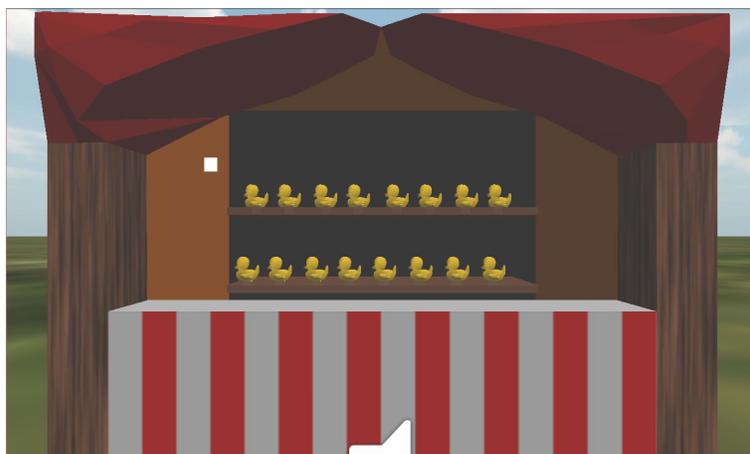


Figura 5.13: Casa de tiro con patos como objetivos.

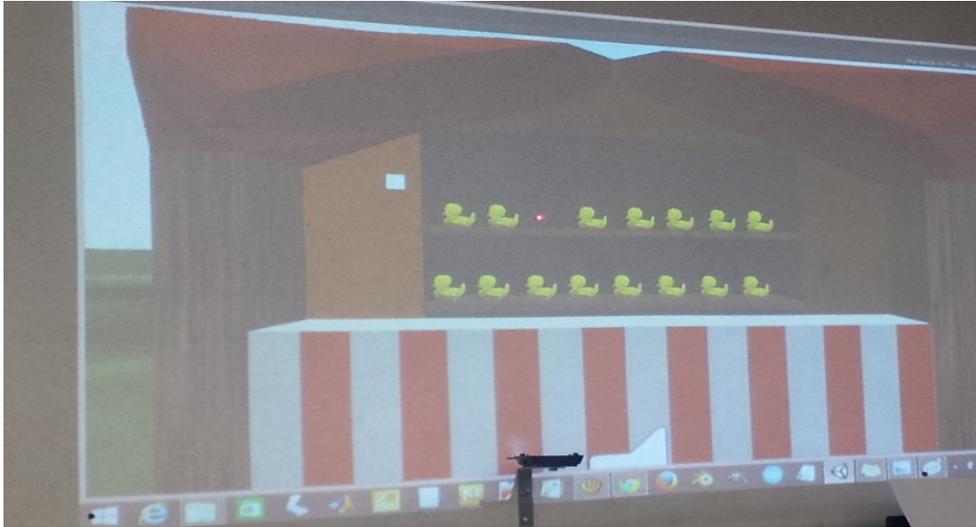


Figura 5.14: Escena de la casa de tiro con patos como objetivos.

3. La tercera escena es de un terreno en donde aparecen siluetas de venados aleatoriamente, como se muestra en la figura 5.15.

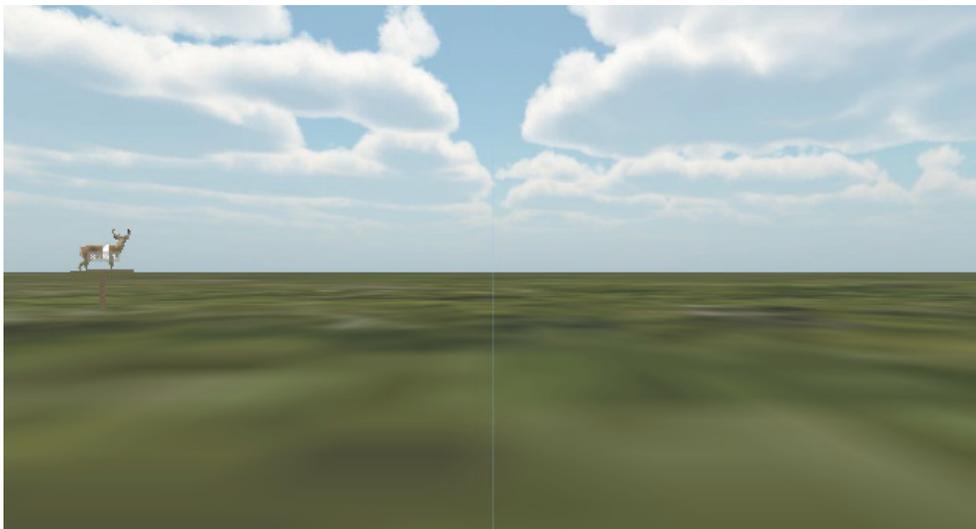


Figura 5.15: Escena de un terreno con venados que aparecen aleatoriamente.

Capítulo 6

Conclusiones y Trabajos a Futuro

6.1. Resultados

En el capítulo 3 se estableció el modelo matemático para describir el comportamiento de una bala al salir del cañón del arma, conociendo así su trayectoria y todos los factores que la modifican. El siguiente paso después de generar el modelo, es el de la validación de éste, lo que significa que la solución calculada del modelo debe ser lo suficientemente aproximada a la realidad. En este capítulo se muestran los resultados obtenidos así como la validación del modelo matemático antes mencionado.

6.1.1. Validación del Modelado

La primera prueba de validación del modelo es revisar si la solución satisface un comportamiento físico obvio. En este caso por intuición y experiencia sabemos que la trayectoria del proyectil debe generar una forma casi parabólica la cual se genera en las simulaciones realizadas como se muestran en las figuras 6.1 y 6.2. Así mismo, las distancias totales recorridas cumplen con el comportamiento esperado debido a la velocidad con la que sale de la boca del cañón.

Las simulaciones realizadas fueron con viento (ver figura 6.1) y sin viento (ver figura 6.2), tomando en cuenta la velocidad inicial de $852 \frac{m}{s}$ correspondiente a la carabina M4, con una munición $5,56 \times 45 mm$, el cual se encuentra estandarizado por la OTAN (La Organización del Tratado del Atlántico Norte), que es la organización mundial encargada de definir el calibre que deben utilizar los cuerpos militares y policiales para generar heridas que protejan los derechos humanos. Estas simulaciones se realizaron con un ángulo de disparo de 45 grados, a una altura inicial de 1.8 metros y disparados con una carabina M4, obteniendo los siguientes resultados:

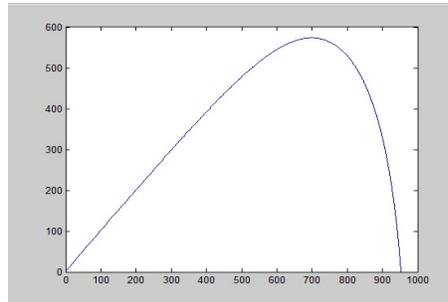


Figura 6.1: Simulación que describe la trayectoria de un proyectil, sin efecto del viento.

$$X_{max} = 940,90 \text{ metros.}$$

$$Y_{max} = 561,82 \text{ metros}$$

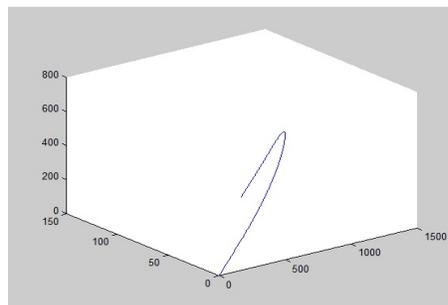


Figura 6.2: Simulación que describe la trayectoria de un proyectil, con efecto del viento.

$$X_{max} = 940,90 \text{ metros.}$$

$$Y_{max} = 561,82 \text{ metros}$$

$$Z_{max} = 96,90 \text{ metros}$$

Así mismo se realizaron simulaciones para comparar los resultados obtenidos en el libro Trayectoria y efectos de los proyectiles en armas cortas [13], dando como resultado los siguientes valores para disparos a una altura de 1.47 metros con un ángulo de 0° en la figura 6.3 y 15° en la figura 6.4, ambos disparados con una pistola 9mm.

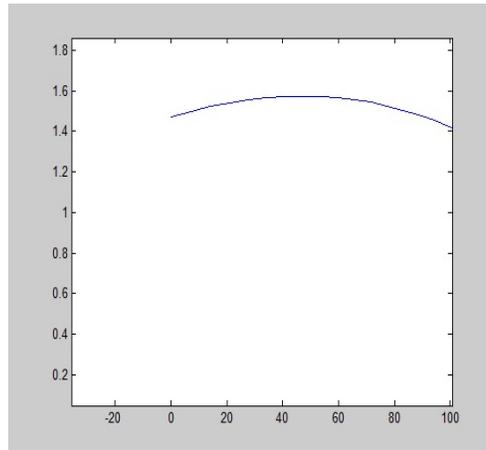


Figura 6.3: Simulación del modelo matemático con un disparo a 0 grados con pistola 9mm.

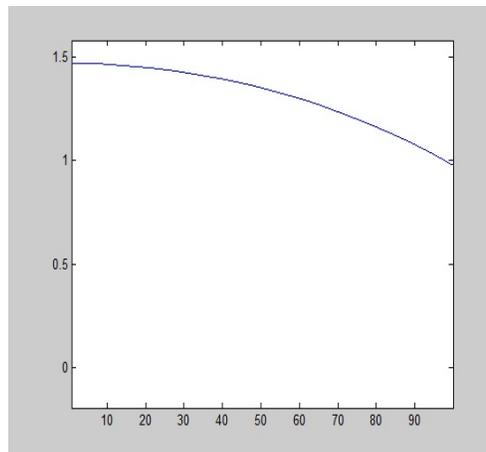


Figura 6.4: Simulación del modelo matemático con un disparo a 15 grados con pistola 9mm.

6.1.2. Conclusiones

En este trabajo de tesis se logró el objetivo de realizar un simulador de tiro mediante la captura de un punto láser, concluyendo con lo siguiente:

- Se logró la lectura de un punto láser mediante la captura de imágenes con una cámara Optitrack V120, estas lecturas fueron enviadas hacia la computadora segmentándolas por el método de umbralización y mapeando las coordenadas del punto láser en el mundo real a un ambiente virtual, lo que permite generar disparos al instante en dicho ambiente.
- Se desarrollaron dos ambientes virtuales, uno simulando un ambiente para cacería en donde aparecían venados aleatoriamente. El segundo ambiente que se desarrolló fue una casa de tiro, en donde se les dispara a patos inmóviles colocados sobre dos repisas. Estos ambientes fueron integrados con sonidos, lo que permite una inmersión al usuario que atrae su atención para que crea estar disparando a blancos reales.
- Se instrumentó un arma de plástico a la cual se le integró un láser que emite un haz de luz que es detectado por la cámara Optitrack e interpretado como un disparo en la aplicación, mejorando así la interacción del usuario con la aplicación al oprimir el gatillo de dicha arma.
- Por ultimo se integró el modelo matemático que describe la trayectoria de un proyectil, generando así disparos con trayectorias reales dentro del ambiente virtual. Comprobando resultados con disparos reales, los cuales fueron analizados en el capítulo 1 en base al libro Trayectoria y efectos de los proyectiles en armas cortas [13], mostrándose en las figuras 6.3 y 6.4 dando como resultado final que el modelo genera una trayectoria muy aproximada a la real. En caso de que el modelo no calculara valores cercanos a la realidad se debería obtener una nueva solución numérica, revalidar el modelo y modificarlo. Así también, al integrar las ecuaciones matemáticas a Unity 3D y realizar las simulaciones en modo edición diferenciaron del software matemático por menos de 4 metros, mientras que las simulaciones realizadas en modo PLAY, diferenciaron por 10 metros debido a consumo de recursos en la aplicación.

6.1.3. Trabajos a futuro

Como trabajo a futuro se considera la mejora gráfica de los ambientes virtuales generados, así como la creación de nuevos escenarios en donde se practique la puntería y capacidad de reacción. Otra modificación que mejoraría la interacción del usuario con la aplicación sería añadir una forma de navegación dentro de los escenarios existentes.

Referencias

- [1] I. Reid A. Criminisi y A. Zisserman. “A Plane Measuring Device”. En: *University of Oxford*. (1997).
- [2] VICENTE ABOITES. *EL LÁSER*. D.R. © 1991, FONDO DE CULTURA ECONÓMICA, S. A. DE C. V., Primera edición, 1991. ISBN: 968-16-3578-7.
- [3] Yusuf Ismail Nakhoda Aryuanto Soetedjo Eko Nurcahyo. “Development of a Cost-Effective Shooting Simulator Using Laser Pointer”. En: *International Conference on Electrical Engineering and Informatics* July (2011), págs. 17-19.
- [4] Patrick Diskin. *Nintendo Entertainment System Documentation*. Nintendo, Version 1.0, Agosto 2004, pág. 47.
- [5] *Empresa Española: lasersniper*. Fecha de consulta 2014. URL: <http://www.lasersniper.es>.
- [6] *Empresa Estadonunidense: Dynamic Animation Systems, Inc*. Fecha de consulta 2014. URL: <http://www.d-a-s.com/VICE>.
- [7] *Empresa Estadounidense: Ti Training*. Fecha de consulta 2014. URL: <http://www.titraining.com>.
- [8] *Empresa Estadounidense VirTra*. Fecha de consulta 2014. URL: <http://www.virtra.com/>.
- [9] *Empresa Francesa: Meggitt Training Systems*. Fecha de consulta 2014. URL: <http://www.meggitttrainingsystems.com>.
- [10] *Empresa global: Laser shot*. Fecha de consulta 2014. URL: <http://www.lasershot.com>.
- [11] Philippe Coiffel Grigore C. Burdea. *Virtual Reality Technology*. Second edition. AWiley Interscience Publication, 2003. ISBN: 047136089-9.
- [12] Rudolf Scitovski Josip Cumin Branko Grizelj. “NUMERICAL SOLVING OF BALLISTIC FLIGHT EQUATIONS FOR BIG BORE AIR RIFLE.” En: *ehni ki vjesnik* č 16. (2009).
- [13] Roberto Jorge Locles. *Trayectoria y efectos de los proyectiles en armas cortas*. La Roca, Argentina, 2006, pág. 190. ISBN: 9789875170971.
- [14] José Daniel Cortijo Martínez. *Curso de balística*. URL: www.josedanielcortijo.com.
- [15] EURAMET Technical Committee for Mass y Related Quantities. “Guidelines on the calibration of non-automatic weighing instruments”. En: *European Association of National Metrology Institutes* (2009).

-
- [16] Ulises Román Concha Nora La Serna Palomino. *Técnicas de segmentación en procesamiento digital de imágenes*. Universidad Nacional Mayor de San Marco, 2007, pág. 16.
- [17] Giacomo P. “Equation for the determination of the density of moist air (1981)”. En: *Metrologia* 18 (1982), págs. 33-40.
- [18] laboratorio de criminalística Peritos balísticos. *Manual de balística forense*. Managua, Nicaragua, 2013, pág. 88.
- [19] Souza Gerardo P.F.A. *Manual de Tiro*. 2011.
- [20] Paula Tristan Rubén Wainschenker José María Massa. *Procesamiento digital de imágenes*. Área procesamiento de señales, UNAM, 2011, pág. 97.
- [21] *Soporte para play station move*. Fecha de consulta 2014. URL: <http://es.playstation.com/ps3>.
- [22] SUPPORT.NINTENDO.COM. *Wii Console Manual wii Operations Manual System Setup*. Nintendo, 2009, pág. 40.
- [23] Zisserman A. Torr P. “Robust parameterisation and computation of the trifocal tensor”. En: *In Proc. British Machine Vision Conference* (1996).
- [24] José Guillermo Hincapié Zuluaga. *Manual de balística*. I. Ediciones librería del profesional(ES), 2000, pág. 482. ISBN: 9586353613.