

Presentación del proyecto juego "ChecoGame"

Jesus Rigoberto Estrada Ortíz

Alma Citlali Velez Gutierrez

Nestor Zuno Segura

*CENTRO UNIVERSITARIO DE CIENCIAS
EXACTAS E INGENIERÍAS, (CUCEI, UDG)*

jesus.estrada1635@alumnos.udg.mx

alma.velez4624@alumnos.udg.mx

nestor.zuno1291@alumnos.udg.mx

Abstract— Este proyecto se enfoca en desarrollar un videojuego de carreras de Fórmula 1 que utiliza el reconocimiento del movimiento de la cabeza para evadir obstáculos, con el objetivo de ofrecer una experiencia inclusiva y accesible para personas con diversas capacidades físicas. Se emplearon tecnologías como Pygame, junto con algoritmos de inteligencia artificial, para crear un juego dinámico. En conclusión, este proyecto no solo ofrece entretenimiento accesible, sino que también contribuye a generar conciencia sobre la importancia de la inclusión en la comunidad de los videojuegos.

Palabras claves:

- Videojuego de carreras
- Inclusión
- Discapacidades
- Rehabilitación
- Inteligencia artificial

Repositorio de código:

<https://github.com/citlali/Proyecto-de-algoritmos>

I. INTRODUCCIÓN

En un mercado de videojuegos donde la inclusión, la accesibilidad son cada vez más importantes, surge la necesidad de desarrollar experiencias que sean tanto entretenidas como accesibles para personas con diversas capacidades. Este proyecto se centra en crear un videojuego de carreras de Fórmula 1 que utiliza el reconocimiento del movimiento de la cabeza para evadir obstáculos.

II. TRABAJOS RELACIONADOS

Vivimos en una sociedad en la que la tecnología ha cobrado cada vez más relevancia en todos los ámbitos de nuestras vidas. En el ámbito de la salud se han logrado importantes progresos en la atención a los pacientes, incluyendo mejoras significativas en los métodos de rehabilitación.

Existen diversas razones que pueden llevar a las personas a tener discapacidades temporales o permanentes. Es por eso

que en el mercado de videojuegos está explorando realizar proyectos donde los videojuegos sean accesibles para personas con discapacidades y personas que necesiten rehabilitación. Algunos proyectos similares son los siguientes:

A. EyeMine

Utiliza tecnología de seguimiento ocular para permitir que personas con discapacidades motoras severas jueguen Minecraft.

B. Xbox Adaptive Controller

Microsoft ha desarrollado un controlador adaptable diseñado para permitir que personas con discapacidades físicas jueguen videojuegos en la consola Xbox.

C. SpecialEffect

Se dedica a ayudar a personas con discapacidades a disfrutar de los videojuegos. Ofrecen evaluaciones personalizadas y soluciones tecnológicas, como controles adaptativos, para que las personas con discapacidades puedan jugar cómodamente.

III. DESCRIPCIÓN DEL DESARROLLO DEL PROYECTO

Dividimos nuestro proyecto en tres secciones: backend, frontend e inteligencia artificial. Cada uno se encargó de un conjunto específico de tareas dentro de su área. Al final, integramos las tres partes para optimizar la eficiencia del equipo.

A. Backend

La parte del backend se encargó del desarrollo de objetos y movimientos también realizó un sistema de puntuación para que conforme fuera avanzando en el juego fuera obteniendo cierta cantidad de puntos.

B. Frontend

La parte del frontend fue responsable de la GUI, añadiendo la pista, los monoplazas y los sonidos de ambientación durante el juego. También implementó las delimitaciones de las

posiciones de los vehículos, para que los jugadores sigan el camino dentro de la pista.

C. Inteligencia Artificial

Se encargó del reconocimiento facial, el ajuste de grados para los movimientos de los monoplazas. Se utilizó una inteligencia artificial ya entrenada para el reconocimiento facial.

D. Requerimientos

Los requerimientos de este proyecto son:

1) Integración de inteligencia artificial:

1. Movimiento con rostro

- Detección del rostro.
- Cálculos de grados de inclinación .
- Definición de movimiento izquierda y derecha.
- Implementación del código final.

2) Integración de interfaz gráfica:

1. Desarrollo de GUI

- Introducción del juego.
- Página principal del juego.
- Desarrollo de pista.
- Delimitaciones.

3) Integración de frontend:

1. Creación de objetos

- Carro principal.
- Carros enemigos.
- Leaderboard.
- Hitboxs.

2. Movimiento con teclas (para pruebas de funcionamiento).

E. Tecnologías que se utilizaron

Integramos la IA Face Mesh para el reconocimiento facial en nuestro juego. Face Mesh es una tecnología de inteligencia artificial que utiliza modelos de aprendizaje profundo para detectar y seguir los puntos clave en el rostro humano, con Face Mesh, podemos ofrecer una experiencia de juego más accesible a las personas que utilicen nuestro juego.

F. Pruebas realizadas

La gráfica titulada "Tiempo sin errores" ilustra el progreso del jugador en términos del tiempo jugado sin que el juego contenga errores a lo largo de tres sesiones. En esta gráfica, el

eje horizontal (x) representa el número de juegos, mientras que el eje vertical (y) muestra el tiempo en minutos durante el cual no se presentaron errores en el juego.



Los datos específicos de cada sesión son los siguientes:

- *Juego 1:* Durante la primera sesión, el jugador consiguió mantener el juego sin errores durante 36 minutos.
- *Juego 2:* En la segunda sesión, hubo una mejora significativa, logrando 68 minutos sin errores.
- *Juego 3:* En la tercera sesión, el tiempo sin errores aumentó aún más, alcanzando los 87 minutos.

El análisis de la gráfica revela una clara tendencia ascendente, lo que indica un progreso constante en la capacidad del jugador para mantener el juego libre de errores. Este incremento progresivo sugiere que el juego no presenta errores conforme al tiempo, logrando aumentar considerablemente el tiempo de juego sin interrupciones debido a errores.

La mejora notable entre cada sesión evidencia un proceso de aprendizaje y adaptación eficiente, destacando la capacidad del jugador para analizar sus errores pasados y aplicar conocimientos adquiridos para optimizar su desempeño en las siguientes sesiones.

G. Proceso de implementación

Los requerimientos de este proyecto son:

- 1) *Definición de Objetivos Específicos:* Crear un videojuego accesible que utiliza el movimiento de la cabeza para el control del juego.
- 2) *Asignación de herramientas:* Pygame para desarrollo, bibliotecas de IA.
- 3) *Cronograma:*
 - Investigación y aprendizaje de Pygame e IA(1 semana).
 - Desarrollo de la GUI y monoplazas(1 semana).
 - Integración de IA y pruebas iniciales(3 días).
 - Optimización y pruebas finales (4 días).

4) Ejecución:

- Despliegue de actividades:

- Investigación: Aprender y documentar cómo usar Pygame y las bibliotecas de IA necesarias.
- Desarrollo de GUI: Crear una interfaz de usuario amigable y accesible.
- Desarrollo de Monoplazas: Programar los movimientos de los monoplazas de los enemigos.
- Integración de IA: Implementar IA para el movimiento del jugador y la detección del movimiento de la cabeza del jugador.
- Desarrollo de Mecánicas de Juego: Puntajes y condiciones de Game Over.

-Supervisión y monitoreo:

- Reuniones semanales para revisar el progreso.

5) Control y evaluación:

-Evaluación de Resultados:

- Pruebas de usabilidad con personas con discapacidades.
- Revisión del rendimiento y funcionalidad del juego.

-Identificación de Problemas:

- Documentar errores y áreas de mejora detectadas durante las pruebas.

-Ajustes y Correcciones:

- Solucionar bugs y optimizar el código para mejor rendimiento.
- Mejorar la accesibilidad según el feedback recibido.

6) Finalización y revisión:

-Documentación del Proceso:

- Registrar todos los procedimientos, decisiones tomadas y resultados obtenidos.

-Retroalimentación:

- Recoger opiniones y sugerencias de los usuarios finales para futuras mejoras.
- Evaluar el impacto del videojuego en la rehabilitación y entretenimiento de los usuarios.

Módulo I Justificación de Arquitectura y Programación de Sistemas

Optamos por Python debido a su legibilidad y sintaxis, lo que simplifica el desarrollo. Además, su amplio conjunto de

bibliotecas acelera el proceso de desarrollo y garantiza la calidad del producto. La versatilidad de Python nos permite abordar diferentes aspectos del proyecto, promoviendo la cohesión del equipo y la integración fluida de componentes.

Módulo II Justificación de Sistemas Inteligentes

Optamos por utilizar redes neuronales en conjunto con la IA Face Mesh por su capacidad para analizar y procesar de manera eficiente datos complejos, como las características faciales en tiempo real. Las redes neuronales son especialmente adecuadas para tareas de reconocimiento y seguimiento facial debido a su capacidad para aprender patrones y características únicas en conjuntos de datos extensos. Además, las redes neuronales permiten adaptar dinámicamente el modelo de reconocimiento facial a diferentes condiciones de iluminación, ángulos de visión y expresiones faciales.

Módulo III Justificación de Sistemas Distribuidos

El proyecto maneja información en tiempo real mediante sockets para asegurar la sincronización precisa del juego, lo que permite una experiencia de usuario fluida. Además, es escalable y puede alojarse en una plataforma en la nube, facilitando su descarga, actualizaciones continuas y soporte para un gran número de usuarios simultáneamente.

IV. RESULTADOS OBTENIDOS DEL PROYECTO

Uno de los resultados esperados de nuestro proyecto es que las personas con discapacidad puedan controlar el carro en el juego utilizando únicamente el movimiento de la cabeza. Esta funcionalidad busca hacer el juego más accesible e inclusivo, permitiendo que un grupo más amplio de personas disfrute de la experiencia de juego sin importar sus capacidades físicas.

V. CONCLUSIONES Y TRABAJO A FUTURO

En conclusión, este proyecto no solo proporciona entretenimiento, sino que también promueve la inclusión y la accesibilidad en la comunidad de videojuegos. Ha demostrado ser una herramienta valiosa para crear conciencia sobre la importancia de diseñar tecnologías que sean accesibles para todos, sin importar sus capacidades físicas.

Las pruebas realizadas, representadas en la gráfica de "Tiempo sin errores", indican una mejora progresiva en el desempeño del juego, mostrando que el sistema es robusto. Los resultados destacan que la implementación del reconocimiento facial es eficaz, proporcionando una experiencia de juego fluida y libre de errores prolongados.

Este proyecto ha demostrado ser un avance significativo en la creación de videojuegos inclusivos, específicamente diseñados para ser accesibles a personas con diversas capacidades físicas. A través del uso de tecnologías

avanzadas como Pygame y algoritmos de inteligencia artificial, se ha desarrollado un videojuego de carreras de Fórmula 1 que utiliza el reconocimiento del movimiento de la cabeza para evadir obstáculos.

A pesar de los logros alcanzados, hay varias áreas en las que este proyecto puede evolucionar y mejorar. Algunas de las direcciones futuras para el desarrollo incluyen:

Ampliación de Funcionalidades:

- Incorporar nuevas características en el juego, como distintos niveles de dificultad, más obstáculos y escenarios, para enriquecer la experiencia de juego y mantener el interés del usuario a largo plazo.

Optimización del Rendimiento:

- Optimizar el rendimiento del juego para que funcione de manera eficiente en una variedad de dispositivos, desde computadoras de alta gama hasta dispositivos móviles con recursos limitados.

Mantenimiento y Actualizaciones Continuas:

- Establecer un ciclo regular de mantenimiento y actualizaciones del software para corregir errores, mejorar la seguridad y añadir nuevas funcionalidades basadas en el feedback de los usuarios.

REFERENCIAS

- [1] "Pygame Front Page — pygame v2.6.0 documentation". pygame news. Accedido el 2 de mayo de 2024. [En línea]. Disponible: <https://www.pygame.org/docs/>
- [2] "Guía de detección de puntos de referencia faciales | Edge | Google for Developers". Google for Developers. Accedido el 2 de mayo de 2024. [En línea]. Disponible: https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker?hl=es-419
- [3] "Google Colab". Google Colab. Accedido el 2 de mayo de 2024. [En línea]. Disponible: [https://colab.research.google.com/github/googlesamples/mediapipe/blob/main/examples/face_landmarker/python/\[MediaPipe_Python_Tasks\]_Face_Landmarker.ipynb?hl=es-419](https://colab.research.google.com/github/googlesamples/mediapipe/blob/main/examples/face_landmarker/python/[MediaPipe_Python_Tasks]_Face_Landmarker.ipynb?hl=es-419)
- [4] "3.12.3 Documentation". 3.12.3 Documentation. Accedido el 2 de mayo de 2024. [En línea]. Disponible: <https://docs.python.org/3/>