

Desarrollo de humanos digitales con capacidades de aprendizaje mediante inteligencia artificial

Berenice L. Balich, Martin E. Medrano, Nestor A. Balich, Franco A. Balich,

CAETI - Centro de Altos Estudios en Tecnología Informática
Universidad Abierta Interamericana. Informática (UAI)
Montes de Oca 745. Ciudad Autónoma de Buenos Aires, Argentina

```
{nestor.balich, francoadrian.balich}@uai.edu.ar  
{berenicelourdes.balich,  
martinezequiel.medrano}@alumnos.uai.edu.ar
```

Resumen. Los avances de la tecnología en los motores gráficos utilizados para videojuegos, sumados a los avances en inteligencia artificial conversacionales con capacidad de aprendizaje como chat GPT, de acceso libre, nos permiten investigar y proponer el desarrollo de un avatar digital de un humano real, con sus características físicas, movimientos y gestos sincronizados con una voz digital que pueda aprender de la interacción con los alumnos o sobre un contenido determinado.

Este trabajo propone la creación de humanos digitales con herramientas libres que permitan crear un modelo 3D de una persona real, animarlo y crear un protocolo de comunicación entre el motor gráfico y la API de inteligencia artificial (IA). Entrenándolo con las clases de robótica dictadas en la universidad y que tenga la capacidad de responder preguntas en base a ello, a través de texto, voz y desde redes sociales como Discord o programas de comunicación como Telegram.

Palabras clave: inteligencia artificial, omniverse, realidad virtual, chatGPT, metahumano

1 Introducción

En este trabajo proponemos la creación de un humano digital, que permita integrar un chatbot de gestión educativa en Discord [1] articulando con el servicio API de chatGPT como herramienta en contextos educativos [2] afrontando nuevos retos para la educación superior a distancia [3]. Para ello evaluamos diferentes softwares y nos decidimos por realizar el desarrollo para el software Metahuman Creator de Epic Games, que a medida que fuimos utilizándolos confirmamos las posibilidades que nos brindaba en cuanto a calidad de imagen, procesamiento en tiempo real, así también nos encontramos con el inconveniente de poca documentación en referencia al control gestual e integración con Python y chatGPT, temas que pudimos resolver y comentaremos en el presente trabajo.

El presente trabajo está centrado en la utilización de las características gráficas y animación de tiempo real de motores de videojuegos, en este caso Unreal pues con enuncia Chayka [4] *“La infraestructura y las herramientas de los videojuegos se van a apoderar cada vez más de todas las plataformas digitales”* sumado a los entornos virtuales como metaverso o herramientas de base para el diseño de realidad virtual como Omniverse [5] y metahumano [6] optimizados para el trabajo en línea que nos permiten realizar entornos virtuales en tiempo

Para crear nuestro humano digital con este software primero tuvimos que acceder a su plataforma web y registrarnos de forma gratuita. En dicha plataforma nos encontramos con gran variedad de metahumanos, con amplias configuración y acciones de demostración de movimiento fluidas. Esta plataforma permite exportar el metahumano a el motor gráfico de Unreal, donde planteamos hacer el desarrollo principal.

2 Digitalización humana

El proceso de digitalización de una cabeza humana a un avatar 3D implica varias etapas y técnicas. A continuación, se detallan los pasos comunes en este proceso:

- Escaneo 3D: El primer paso es capturar la geometría de la cabeza humana en 3D utilizando un escáner 3D. Esto puede hacerse utilizando tecnología de escaneo láser, escaneo fotográfico o escaneo por tomografía computarizada (CT).
- Limpieza de datos: Una vez que se ha capturado la geometría de la cabeza humana, se realiza una limpieza de datos para eliminar cualquier error o imperfección en el modelo 3D.
- Texturización: Después de limpiar los datos, se aplica una textura al modelo 3D. Esto se hace proyectando imágenes de la textura en el modelo 3D para dar la apariencia de una piel humana realista.
- Rigging: Luego, se crea un esqueleto virtual para la cabeza humana, que se utiliza para animar el modelo 3D. El proceso de crear un esqueleto virtual se conoce como rigging.
- Animación: Finalmente, se anima la cabeza humana 3D utilizando herramientas de animación. Esto permite que el avatar 3D se mueva y exprese emociones.

Existen varias herramientas gratuitas que pueden acelerar este proceso, luego de evaluar varias de ellas seleccionamos dos de ellas, una basada en inteligencia artificial y la segunda basada en fotografías 360 grados.

2.1 Proceso con AvatarSDK

Con la intención de crear un avatar 3D en función de un humano real creamos una primera versión digital del director del laboratorio de robótica física e inteligencia artificial de nuestra universidad (ver Fig. 1), centrados en el modelado 3D de su rostro. Para ellos evaluamos diferentes softwares como plugins para Blender, software pagos y finalmente utilizamos el programa “AvatarSDK” [7] que nos permitía mediante inteligencia artificial con una fotografía generar un modelo 3D.

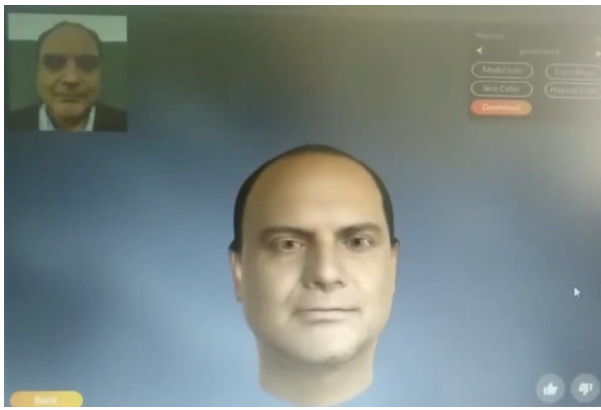


Fig. 1. Modelado 3D mediante IA con AvatarSDK.

Se exportó este modelo a Blender el cual nos permite modificar el avatar (ver Fig. 2) para luego convertirlo en un archivo compatible con Unreal Engine.

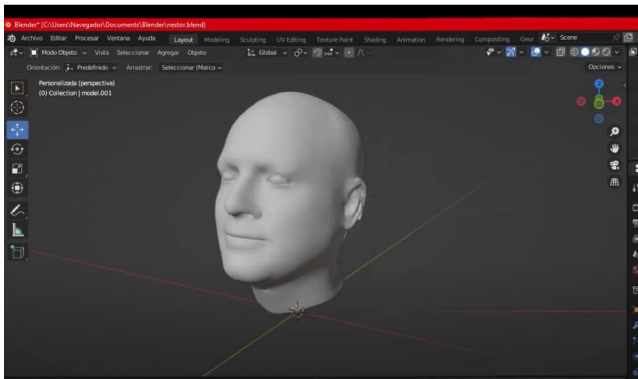


Fig. 2. Ajuste y exportación de formatos compatibles con Blender.

Una vez en Unreal seleccionamos las facciones faciales requeridas para la animación, como por ejemplo los ojos y la boca, para luego convertirlo en humano digital y poder cargarlo en nuestra cuenta de Metahuman Creator [9]. Desde la plataforma

mencionada empezamos a personalizar nuestro propio modelo 3D para que sea lo más parecido posible al humano real de base, modificando su textura física, su color de piel, de ojos, de cabello, y de vestimenta, también la textura de su piel, el calzado, y la ropa (ver Fig. 3) de esta forma crearemos dos modelos uno masculino profesor robot basado en un humano real y Elektra versión femenina basada en la herramienta digital.



Fig. 3. Importación a metahuman de modelo digital 3D.

2.2 Proceso con PolyCam

Como segunda prueba de digitalización utilizamos la aplicación “PolyCam” [10], que, mediante la toma de 60 fotografías a 360 grados, en promedio nos permite generar una modelado 3d de la cabeza. Tuvimos que repetir el proceso varias veces tomando debidos factores de iluminación, posición y calidad de imagen. Al tercer intento logramos un modelo 3D que cumpliera con nuestras expectativas, importamos el archivo a Blender y modificamos algunos polígonos para que quede más real en comparación al humano digital anterior (ver Fig. 4), y luego lo pasamos a Metahuman Creator en donde lo personalizamos, sin embargo, esta versión sigue en proceso de experimentación.



Fig. 4. Modelo mejorado de humano digital animado en tiempo real.

Por último, elegimos a María de Metrópolis 1927 como modelo de inspiración “ELEKTRA”, en principio hicimos una prueba inicial con Avatar SDK, pero el resultado no fue el óptimo (ver Fig. 5) siendo esta la base para empezar a diseñar el modelo en metahumano creator decidimos no utilizarlo. Con el buen resultado obtenido al importar el modelo de profesor robot (ver Fig. 4) desde la plataforma de Metahuman Creator a nuestro desarrollo en unreal decidimos crear una versión femenina ELEKTRA empleando las herramientas y modelos con que cuenta la herramienta. Esto nos permitió tener dos modelos 100% compatibles, un humano y humana digital centrándonos en el desarrollo y control de las expresiones, digitalización de voz y el modelo conversacional de IA.

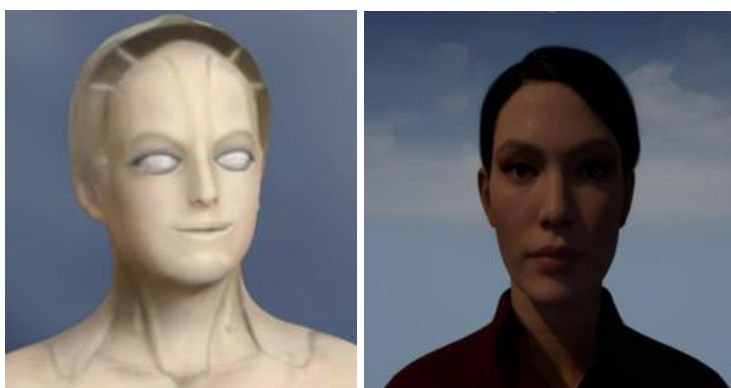


Fig. 5. Comparación de calidades de imagen de Elektra en las aplicaciones creadas con modelo 3D fijo (avatar 3d) descartado y el creado 100% con metahuman creator.

3 Animación de gestos faciales

Uno de los objetivos del proyecto fue utilizar herramientas Open Source o gratuitas. Para poder crear las animaciones en tiempo real, sincronizando con audio, utilizamos dos enfoques que nos permitieran utilizar las capacidades gráficas y de animación en tiempo real del motor de videojuegos Unreal, para ellos desarrollamos dos enfoques, uno basado en hardware “Omniverse” y el segundo basado en una arquitectura distribuida

3.1 Desarrollo en Omniverse

Realizamos una investigación de diferentes herramientas, en donde finalmente utilizamos NVIDIA Omniverse Audio2Face.

Omniverse Audio2Face es una aplicación que nos “genera de forma rápida y fácil animaciones faciales expresivas a partir de una fuente de audio con tecnología IA de Deep Learning de NVIDIA” NVIDIA Omniverse Audio2Face Main Page [1].

Antes de comenzar con las pruebas, verificamos que nuestro equipamiento cumpla con los requerimientos técnicos de la aplicación. (Tabla 1)

Tabla 1. Requerimientos técnicos NVIDIA Omniverse Audio2Face.

Elemento	Especificaciones mínimas
Compatibilidad con SO	Windows 10 de 64 bits
CPU	Intel I7, AMD Ryzen (2.5 Ghz o superior)
Núcleos de CPU	4 o más
RAM	16 GB o más
Almacenamiento	SSD de 800 Gb o más
GPU	Cualquier GPU RTX
VRAM	6 GB o más
Placa de video	RTX Geforce RTX 3070, NVIDIA RTX A4000, con 8 GB o más.

Para la creación y desarrollo de la animación facial expresiva, realizamos la grabación de audios en formato .WAV y la importamos en Audio2Face.

Luego importamos un Blendshape y lo utilizamos para generar la animación facial. Seguido exportamos un archivo de extensión .USD en donde se copia toda la animación facial expresiva que nos generó gracias a la IA utilizada en el Software. El archivo exportado nos sirve para el siguiente paso, el cual es aplicarlo a nuestro metahumano en Unreal Engine 5 (UE5).

3.2 Desarrollo en Unreal con conexión a Python

En este modelo, planteamos la creación de un modelo distribuido que permitiera, la conexión en tiempo real a la API de chatGPT, Discord y la interacción de voz y audio a través de un programa cliente Python que controlara las acciones del metahumano en Unreal (ver Fig. 6).

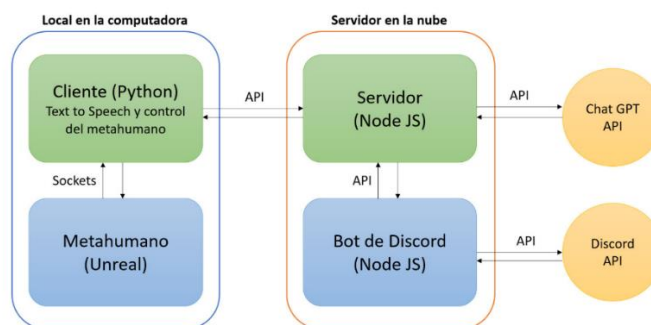


Fig. 6. Arquitectura cliente-servidor desarrollada.

Para ello plantemos una arquitectura de software separada en 4 procesos diferentes los cuales estarán divididos de forma tal de tener dos procesos en la parte del cliente y otros dos en la parte del servidor.

En la parte del cliente utilizamos un software cliente desarrollado con Python que realiza consultas a una API provista por el servidor web en donde se obtendrá un diálogo conformado por una pregunta y una respuesta. Luego usara dicho diálogo para generar el audio de la respuesta que se reproducirá en la computadora y se enviaran por socket la secuencia de los movimientos faciales que el metahumano deberá realizar sincronizados con el audio que se está reproduciendo.

Por otro lado, se creó un servidor con Node JS en donde se manejará una API con varios endpoints de los cuales uno se usara para generar respuestas usando la API de chatGPT y otro para comunicarse con otro proceso el cual mantendrá en ejecución un bot de Discord para que se pueda realizar preguntas por medio de dicha plataforma, se genere una respuesta con chatGPT, se muestre la respuesta por Discord y que tanto la pregunta como la respuesta se guarden en un archivo JSON haciendo una cola de diálogos que van a ser solicitados de a uno por medio de una API.

4 Líneas De Investigación Y Desarrollo

Los ejes principales del trabajo son:

- Implementar un modelo de IA que aprenda los contenidos de la materia robótica.
- Digitalizar un humano real en un avatar 3d.
- Crear una arquitectura que permita unir los diferentes sistemas con Unreal y Discord en tiempo real.
- Implementar un sistema de voz a texto que permita interactuar de forma verbal.

5 Resultados Obtenidos/Esperados

Las pruebas realizadas con Omniverse, mostraron una gran calidad en procesamiento de voz y sincronización gestual. Pero el alto costo de las placas graficas hace que sea poco accesible al resto de los laboratorios de la universidad y de aquellos alumnos que deseen realizar la experiencia.

Las pruebas realizadas con el modelo distribuido fueron satisfactorias, con un menor nivel de sincronización que definimos de manera autónoma concordando solo con las principales vocales, tiempos de silencios y gestualidades. El retardo de comunicación por red creo algunos problemas de sincronismo con expresiones complejas.

En cuanto al sistema de IA encontramos que chatGPT nos permitía entrenarlos con el contenido de los videos de la clase de robótica pasados a texto, y mediante ingeniería de prompts logramos un alto grado de respuesta en concordancia con dicho contenido.

La utilización de Discord como plataforma de consulta, resultado muy conveniente pues los alumnos utilizaron rápidamente el profesor digital, pues estaban familiarizados con dicha plataforma.

Queda por mejorar los problemas de sincronismo en voz y animación digital, para lo cual se planea realizar la implementación del cliente dentro de Unreal y agregar las funcionalidades de movimiento de cuerpo.

6 Formación De Recursos Humanos

El equipo se conformó por 2 docentes de maestría de la Universidad Abierta Interamericana (UAI), estudiantes del doctorado en informática UAI e investigadores en laboratorio de robótica física e inteligencia artificial (LRFIA) y un investigador colaborador. Se capacito al equipo en técnicas de modelado digital, Blender, API rest, inteligencia artificial, Omniverse y programación de sistemas en Python y Blueprint para Unreal.

7 Bibliografía

1. R. Ai, D. Kohli, L. Maina, E. Manelski, N. T. Dhar and C. Brzycki, "Programming an Educational Chatbot to Support Virtual Learning," 2020 IEEE MIT Undergraduate Research Technology Conference (URTC), Cambridge, MA, USA, 2020, pp. 1-5, doi: 10.1109/URTC51696.2020.9668855.
2. García-Peñalvo, F. J. (2023). La percepción de la Inteligencia Artificial en contextos educativos tras el lanzamiento de ChatGPT: disrupción o pánico. *Education in the Knowledge Society (EKS)*, 24, e31279. <https://doi.org/10.14201/eks.31279>
3. RAZO, Omar Terrazas. ChatGPT y los retos de la educación media superior a distancia en México. *Revista Mexicana de Bachillerato a Distancia*, 2023, vol. 15, no 29.
4. Chayka, Kyle. 2022. "video game infrastructure and tools are increasingly going to take over all digital platforms". 18 January. Twitter. <https://twitter.com/chaykak/status/1483490290719764483>
5. I. L. Chamusca, F. V. De Jesus Santos, C. V. Ferreira, T. B. Murari, A. L. Apolinário Junior and I. Winkler, "Evaluation of design guidelines for the development of intuitive virtual reality authoring tools: a case study with NVIDIA Omniverse," 2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), Singapore, Singapore, 2022, pp. 357-362, doi: 10.1109/ISMAR-Adjunct57072.2022.00078.
6. Kemal Gökhan Nalbant, Şevval Uyanik. (2022) A Look At The New Humanity: Metaverse and Metahuman. *International Journal of Computers*, 7, 7-13
7. <https://avatarsdk.com/>
8. NVIDIA Omniverse Audio2Face "NVIDIA Omniverse Audio2Face Main Page", <https://www.nvidia.com/es-la/omniverse/apps/audio2face/>
9. <https://metahuman.unrealengine.com/>
10. <https://poly.cam/>