

Modelado de personaje en 3D para su control a través de un dispositivo kinect

ANTONIO BENITEZ RUIZ¹, MARÍA AUXILIO MEDINA NIETO¹, JORGE DE LA CALLEJA MORA¹, XOCHITL MANZANO SALINAS¹, AND KARLA CECILIA APAN ARAUJO²

¹Maestría en Ingeniería en Sistemas y Cómputo Inteligente, Universidad Politécnica de Puebla, 3er. Carril del ejido serrano s/n, San Mateo Cuanalá, Puebla, México. C.P. 72640, xochitl.manzano4303, antonio.benitez@up Puebla.edu.mx

²Universidad Politécnica de Amozoc, Av. Ampliación Luis Oropeza No. 5202, San Andrés las Vegas 1ra. Secc. CP 72980, Amozoc de Mota, Puebla, México, karla.apan@upamozoc.edu.mx

En este artículo se presenta el diseño de un personaje en 3D utilizando herramientas de cómputo gráfico pasando desde el modelado hasta el aparejo o mejor conocido como rigging, que consiste en la incorporación de una estructura tipo esqueleto a la cual se dota de la capacidad de movilidad. Una vez, desarrollado el personaje se aprovecha el entorno de videojuegos Unity para establecer una conexión entre el editor de Unity, el personaje y el dispositivo Kinect para que la captura del movimiento del cuerpo humano se vea reflejada en la replica del movimiento del personaje en 3D.

© Universidad Politécnica de Puebla

OCIS codes: (140.3490) Lasers, distributed feedback; (060.2420) Fibers, polarization-maintaining; (060.3735) Fiber Bragg gratings.

<http://dx.doi.org/10.1364/ao.XX.XXXXXX>

1. INTRODUCCIÓN

El crecimiento de la industria de los videojuegos se debe en parte a que la tecnología para su desarrollo ha ido en aumento, el video, la simulación, la animación, la realidad virtual (VR), la realidad aumentada (AR), la inteligencia artificial (AI), los motores de videojuegos; además de la expansión de los géneros, la capacidad del hardware utilizando métodos, técnicas y algoritmos para soportarlos y por último de las técnicas propias de la mercadotecnia utilizadas en este medio han dado la posibilidad de crear un área de oportunidad para los desarrolladores y distintas disciplinas que intervienen en dicho proceso [1] [2].

En el proceso de desarrollo a los responsables del contenido audio-visual se les denomina artistas, estos se encargan del desarrollo de personajes u objetos de ambientación y de las animaciones para los personajes. Los artistas se pueden especializar en diferentes aspectos como: artista conceptual, modeladores, artistas de texturización, artistas de iluminación, animadores, actores de captura de movimiento, diseñadores de sonido, entre otros [3].

Independiente del software, el hardware también ha ido mejorando y permitiendo una diversidad de desarrollos,

por ejemplo el dispositivo Kinect que permite capturar los movimientos que realiza una persona situada delante de él, ya sea sólo sus extremidades o con el conjunto del cuerpo, haciendo que la persona controle el objeto o el personaje en 3D [4].

Estas herramientas y dispositivos pueden ser utilizados para el desarrollo de aplicaciones que permitan interactuar con las personas en diferentes tipos de procesos, específicamente, en aquellos que tienen que ver con rehabilitación física. Este trabajo tiene como objetivo diseñar un personaje en tres dimensiones a través de herramientas de cómputo gráfico al que se le pueda integrar un esqueleto (utilizando una cadena cinemática), que permita darle movilidad como un actor digital con el propósito de conectarlo al dispositivo kinect, el cual permitirá mover el personaje desarrollado.

A continuación se describe la organización de este artículo. En la sección 2, se presentan brevemente algunas herramientas de desarrollo gráfico, posteriormente, en la sección 3, se describe el dispositivo de captura de movimiento utilizado en el proyecto para replicar el movimiento. En la sección 4, se describe el desarrollo del personaje y su ligado a un esqueleto para darle movilidad, finalmente, en la sección 5 se presentan los resultados obtenidos.

2. CÓMPUTO GRÁFICO

La producción de manera rápida y económica de imágenes ha ido mejorando gracias al cómputo gráfico y puede ser encontrado en muchas disciplinas ya que proporcionan una ventaja en todas las tareas en la que la representación gráfica puede ser utilizada [5].

Por otra parte, las herramientas de desarrollo gráficos en el área de videojuegos ha creado una área de oportunidad a los desarrolladores muy fructuosa, incrementando la demanda de desarrolladores para áreas como animación, simulación, realidad aumentada, interfaces humano computadora, inmersión, entre otras [6].

El desarrollo de este trabajo, se enfoca en un campo de la

informática conocido como computación gráfica, donde se generan imágenes visuales sintéticas. Esta rama de la computación se divide en varias áreas como: la animaciones 3D en tiempo real que es usada en los videojuegos, la edición de efectos especiales utilizada para las películas y la televisión, la edición de imágenes, modelado de objetos 3D, entre otros [7].

A. Herramientas de modelado en 3D

El diseño aprovecha cada vez más las herramientas de software y como resultante, el diseño asistido por computadora se hace cada vez más fácil conforme pasa el tiempo, esto de acuerdo con John F. Wakerly y varios diseñadores digitales con veinte años de experiencia. De las herramientas de modelado de objetos en 3D que se utilizan actualmente se pueden mencionar las siguientes [8].

- Adobe Fuse.[9]
- Mixamo. [10]
- Blender. [11]
- MakeHuman.[12]

B. Motores de videojuegos

Un motor de videojuegos es a un conjunto de herramientas que permiten el diseño, la creación, la representación y visualización en tiempo real de un videojuego. Los motores de videojuegos ofrecen las funciones como: administración de memoria, animaciones, renderización, gestión de físicas, gestión del sonido, colisiones y la posibilidad de crear scripts para poder manipular la inteligencia artificial del videojuego [13]. Algunos casos de herramientas de desarrollo de videojuegos son las siguientes:

- Unreal Engine.[14]
- Unity 3D. [15].

3. CAPTURA DE MOVIMIENTO

La captura de movimiento es el proceso de registrar la posición y orientación de un evento de movimiento en vivo y convertirlo en términos matemáticos utilizables mediante el seguimiento de una serie de puntos clave en el espacio a lo largo del tiempo y combinarlos para obtener una representación tridimensional única. [16].

En el área del entretenimiento, la captura de movimientos comenzó con la rotoscopía, esta técnica es utilizada en la actualidad por algunos estudios de animación tradicionales para copiar el movimiento real de imágenes a dibujos animados; es decir, redibujar o calcar una imagen real a una imagen animada.[16].

A. Kinect

Kinect es un dispositivo de Microsoft para las consolas Xbox 360 y Xbox One, que permite interactuar y controlar con la consola sin tener contacto físico con un mando. El dispositivo cuenta un proyector de infrarrojos y con dos sensores de profundidad 3D que capturan los datos del esqueleto de las personas que tiene frente a él, para coordinar los movimientos de la estructura que el Kinect tiene establecido [4].

Para poder utilizar las herramientas que ofrece Kinect es necesario instalar el SDK de Kinect para Windows, la versión

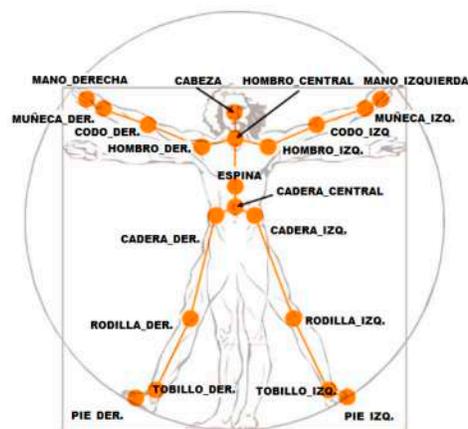


Fig. 1. Skeletal tracking: es el esqueleto que kinect provee para la detección de los movimientos de las personas.

que se utiliza en este trabajo es la 1.8, que contiene las librerías para la manipulación de los datos obtenidos del sensor de Kinect y la herramienta skeletal tracking.

La versión de Kinect que se utiliza en este trabajo es Kinect Xbox 360, esta versión sólo puede identificar 20 articulaciones como se puede ver en la figura 1, en cambio, la versión Kinect Xbox One puede identificar hasta 25 articulaciones y algunos rasgos faciales.

4. DISEÑO DE PERSONAJE EN 3D

La herramienta de MakeHuman, utiliza un método de animación 3D llamado animación por vértice también conocido por varios nombres como: per-vertex animation, shape interpolation, blend shapes o morph target animation. Unos años atrás se utilizaba este método de animación para dar expresiones faciales a rostros en 3D. El método consiste en tomar varios vértices de una malla para controlar la posición de cada vértice, ese segmento de vértices en la malla se almacenan como una serie de posiciones de vértices. En base a este método MakeHuman proporciona una malla humanoide en 3D llamada malla universal para ser manipulada por medio de parámetros, los cuales accionan el método descrito y así poder representar características anatómicas y fenotipos humanos (grupo de pertenencia según las cualidades físicas y observables) de acuerdo a las necesidades del desarrollador; de esta manera con un solo modelo 3D se puede pasar de crear un hombre a una mujer, de un bebé a un anciano, entre otros [17].

La caracterización del personaje en 3D se realiza por medio de barras de deslizamiento (sliders) que manipulan los parámetros de las posiciones de cada vértice, según las características que se este modelando, en la sección principal de MakeHuman proporciona las características género, edad, musculatura, peso, altura, complexión y tres fenotipos: africano, asiático y caucásico. Además se personaliza detalladamente con la ayuda de las siguientes secciones: sexo, cara, torso, brazos y piernas, y por último la sección de medidas; cada una de estas secciones con sus respectivas categorías como se aprecia en la figura 2.

Con lo anterior, se puede decir que la manera de modelar

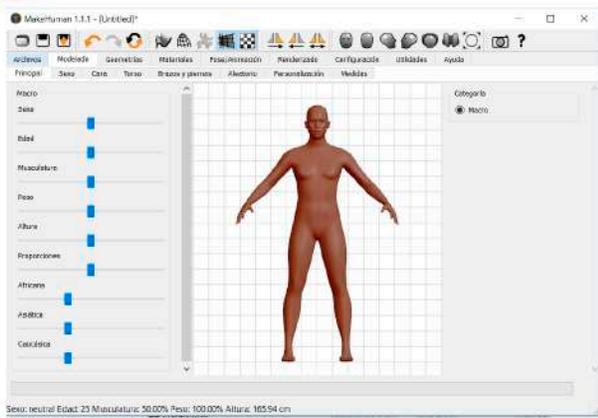


Fig. 2. Modelado de personaje en 3D con el editor de MakeHuman donde se aprecian las barras de desplazamiento para modelar la malla universal.

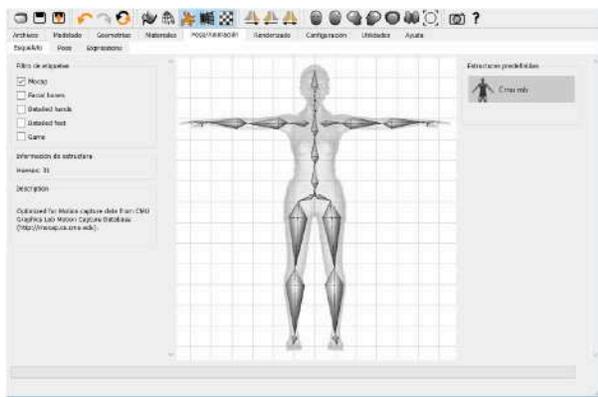


Fig. 3. Aparejo del modelo en 3D y la estructura con el editor de MakeHuman donde se aprecia la estructura para mocap.

el personaje utilizando MakeHuman se desarrolla sin tener que trasladar manualmente los vértices de la malla universal, por lo tanto implica que el desarrollo es de manera rápida. Posteriormente, para lograr la manipulación o el control del personaje en 3D es necesario el proceso de aparejo o rigging, el cual consiste en integrarle una estructura al personaje para su manipulación. MakeHuman cuenta con cuatro estructuras predefinidas que se diferencian de acuerdo al número de eslabones de cada una y a su propósito según su forma de buscarlas por etiquetas; el personaje modelado tiene la estructura para captura de movimiento (mocap, motion capture) que contiene 31 eslabones que se pueden ver en la figura 3.

5. CONTROL DE PERSONAJE EN 3D A TRAVÉS DE KINECT

Para poder manipular el personaje en 3D es necesario importarlo a una plataforma de desarrollo de videojuegos que a su vez pueda hacer uso de las librerías de Kinect, en este caso se utiliza Unity y scripts (un archivo con instrucciones en un determinado lenguaje) en lenguaje C para hacer la conexión entre el Kinect y el personaje en 3D.

El primer paso para establecer la conexión es crear un script donde se importan las librerías del dispositivo Kinect y se

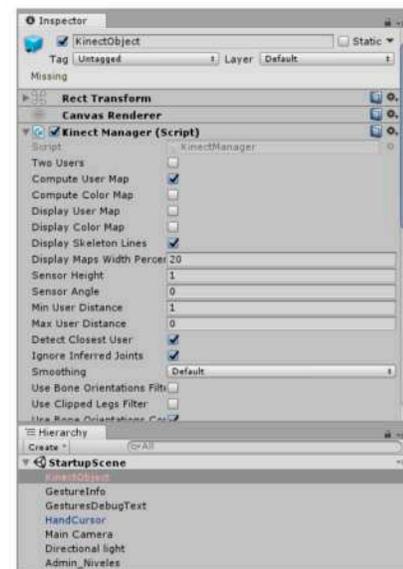


Fig. 4. Variables del script que permiten configurar la conexión de Unity con Kinect

definan los parámetros principales de las propiedades que se desean utilizar de Kinect como se aprecia en la figura 4. El script de conexión permite manipular las propiedades de la cámara RGB de Kinect, establecer si se requiere que Kinect detecte dos personas o sólo una, también si los usuarios tienen permitido controlar el mouse y qué gestos (secuencia de movimientos predeterminados, que el Kinect detecta y los convierte en una orden) son los que podría usar cada usuario, como deslizar a la izquierda, deslizar a la derecha, cursor con la mano derecha, cursor con la mano izquierda y el clic.

Debido a que Unity se maneja por escenarios dentro de un proyecto, se crea un proyecto donde se genera un escenario principal y se crea un objeto vacío para poder agregar un componente, el script de conexión a Kinect; aun si esta escena muestra un menú o no, este objeto de conexión creado en Unity heredará la conexión a las escenas relacionadas con la escena principal sólo si se agregan las demás escenas a la configuración de la estructura del proyecto, por lo tanto, las escenas siguientes o las que contenga el personaje en 3D tendrán que agregarse a la estructura del proyecto para poder establecer la conexión con Kinect.

Lo siguiente es importar el personaje en 3D a la escena del proyecto tomando en cuenta que este personaje tiene un esqueleto, se ve como en la figura 5 y se tiene que configurar el aparejo con el botón que se muestra en la figura 6. Cuando se configura por primera vez el personaje en Unity el aparejo se hace automáticamente, pero esto no garantiza que la configuración sea correcta, por lo tanto, lo recomendable es verificar que cada eslabón del personaje en 3D esté relacionado con la estructura que Unity proporciona para la manipulación de objetos tipo humanoides, es decir que el personaje este mapeado correctamente, el mapeo se puede ver en la figura 7.

Para lograr el control del personaje a través de Kinect es necesario agregarle como componente un script de control, el que se muestra en la figura 8; en este script se trabaja directamente con el esqueleto que el Kinect proporciona por defecto para

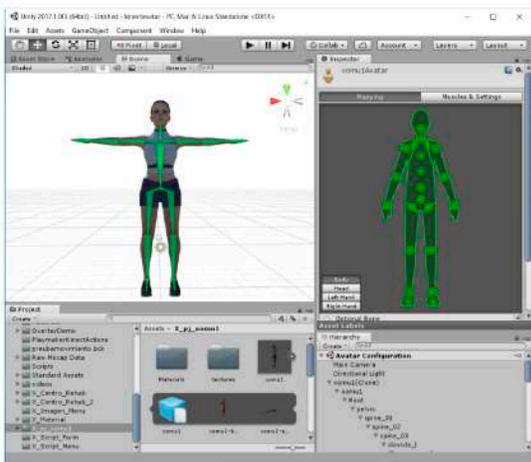


Fig. 5. Importación y visualización del esqueleto del personaje en 3D en la plataforma Unity.

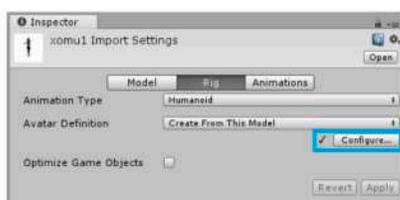


Fig. 6. Después de importar el personaje a Unity se configura el esqueleto.

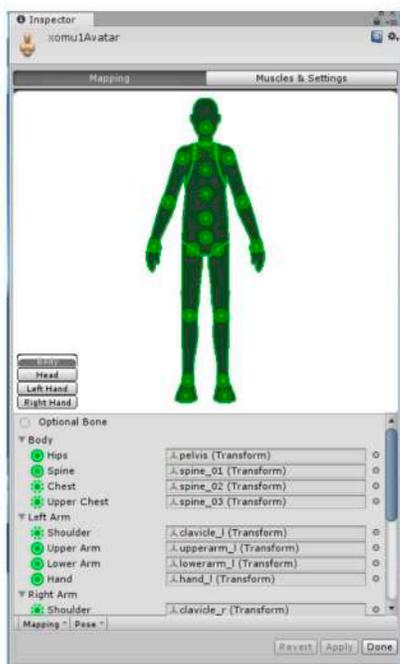


Fig. 7. Interfaz de mapeo que proporciona Unity para asociar manualmente los eslabones del personaje en 3D.

crear variables que permitan asignarle el nombre del eslabón del esqueleto del personaje en 3D al eslabón que se asociará del esqueleto de Kinect y así establecer la conexión. Además de la asociación de los eslabones de ambos esqueletos también se puede configurar si el movimiento del personaje será en espejo

al movimiento que realiza la persona frente al Kinect.

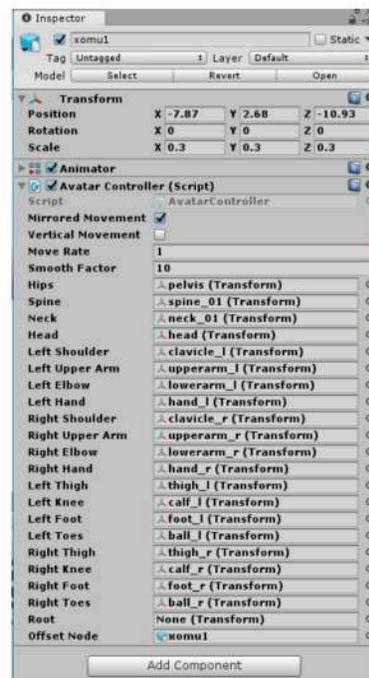


Fig. 8. Variables del script de control del personaje donde se ingresa el nombre del eslabón del personaje en 3D.

6. RESULTADOS

El desarrollo de este trabajo dio como resultado el modelado de un personaje en 3D utilizando múltiples herramientas de software dedicadas al diseño y entorno de videojuegos y su control por medio del dispositivo Kinect y sus librerías correspondientes. El personaje desarrollado es un humanoide femenino que se puede apreciar en la figura 9.

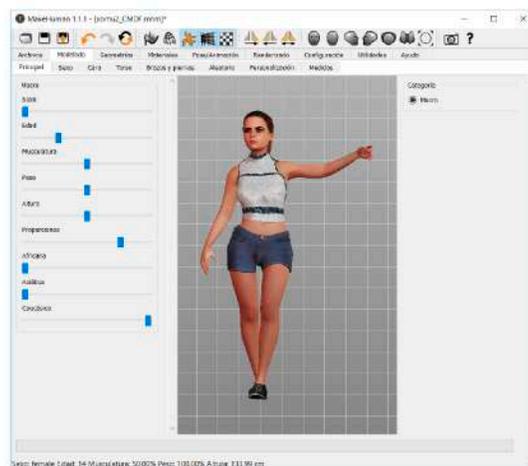


Fig. 9. Personaje en 3D modelado en MakeHuman que incluye una estructura esquelética de 31 eslabones y una pose.

Este personaje fue vestido con atuendos deportivos que fueron obtenidos desde la pagina oficial de MakeHuman. En la misma herramienta, se acopló al personaje una estructura

esquelética que consta de 31 eslabones para su manipulación como se puede observar en la figura 2 y también se puede incorporar una pose al personaje como se ven en la figura 9.

```

37 public Transform Hips;
38 public Transform Spine;
39 public Transform Neck;
40 public Transform Head;
41
42 public Transform LeftShoulder;
43 public Transform LeftUpperArm;
44 public Transform LeftElbow;
45 //public Transform LeftWrist;
46 public Transform LeftHand;
47 //public Transform LeftFingers;
48
49 public Transform RightShoulder;
50 public Transform RightUpperArm;
51 public Transform RightElbow;
52 //public Transform RightWrist;
53 public Transform RightHand;
54 //public Transform RightFingers;
55
56 public Transform LeftThigh;
57 public Transform LeftKnee;
58 public Transform LeftFoot;
59 public Transform LeftToes;
60
61 public Transform RightThigh;
62 public Transform RightKnee;
63 public Transform RightFoot;
64 public Transform RightToes;
65
66 public Transform Root;
67

```

Fig. 10. Relación de variables que relacionan las articulaciones del personaje con las articulaciones del esqueleto con el que se liga.

Posteriormente se importó el personaje a Unity y se configuró con un script dedicado a enlazar y controlar cada uno de los eslabones del personaje en 3D y los eslabones del esqueleto que el Kinet proporciona, en la figura 10 se muestra las variables donde se ingresa el nombre de los eslabones del personaje en 3D y dentro de ese mismo script se realiza la conexión con los eslabones del esqueleto de kinect y estableciéndose de esta manera la conexión a Unity.

7. CONCLUSIONES

El aumento en el número de herramientas de desarrollo gráfico para la creación de personajes virtuales ha aumentado y se cuenta con muchas opciones de software de libre para construir estos personajes. Al mismo tiempo, las herramientas de desarrollo de videojuegos que permiten asociar a personajes virtuales esqueletos o cadenas cinemáticas para dotarlos de movimiento también se han incrementado, tal es el caso de Unity.

Estas herramientas de desarrollo rápido de ambientes en tres dimensiones y los dispositivos de captura de movimiento como kinect han permitido que el proyecto descrito se halla llevado a cabo de forma satisfactoria, logrando alcanzar el objetivo propuesto.

Se desarrolló una primera versión de interfaz entre el dispositivo kinect y el personaje en 3D a través de Unity, permitiendo dotar de movimiento al personaje desarrollado.

Este trabajo servirá como plataforma de soporte para desarrollar aplicaciones más completas. La expectativa es aplicar este trabajo para obtener un sistema de apoyo a la rehabilitación motriz.

8. AGRADECIMIENTOS

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por la beca otorgada a Xochitl Manzano Salinas para la realización de la Maestría en Ingeniería en Sistemas y Cómputo Inteligente en la Universidad Politécnica de Puebla.

REFERENCIAS

1. "Cnbc: Video game industry is booming with continued revenue," <https://www.cnn.com/2018/07/18/video-game-industry-is-booming-with-continued-revenue.html>. Accessed: 2018-08-28.
2. "Forbes: 3 reasons the video game industry is bound for the blockchain," <https://www.forbes.com/sites/andrewrossow/2018/07/16/3-reasons-the-video-game-industry-is-bound-for-blockchain/#344be4dc7810>. Accessed: 2018-08-28.
3. D. Vallejo F. and C. Martín A., *Desarrollo de Videojuegos: Técnicas Avanzadas* (Bubok, 2013).
4. M. Martos and J. Angel, *Microsoft. Aprende de la compañía gigante del software* (Profit, 2012).
5. M. P. B. Hearn Donald, *Computer Graphics: Principles and Practice* (Addison-Wesley Professional, 1996).
6. M. N. Miguel, *Visión Politécnica* 10, 28 (2015).
7. C. Cruz, "Sistema de rehabilitación matriz para niños con síndrome de down utilizando tecnología kinect," Master's thesis, Universidad Politécnica de Puebla (2013).
8. J. F. Wakerly, *Diseño digital: principios y prácticas. Área universitarios* (Pearson Educación, 2001).
9. "Create 3d characters with fuse cc," <https://helpx.adobe.com/creative-cloud/how-to/create-3d-character-adobe-fuse.html>. Accessed: 2018-08-28.
10. "Animated 3d characters. no 3d knowledge required." <https://www.mixamo.com/#/>. Accessed: 2018-08-28.
11. "Sketchup," <https://www.sketchup.com/es>. Accessed: 2018-08-28.
12. "Documentation:what is makehuman?" http://www.makehumancommunity.org/wiki/Documentation:What_is_MakeHuman%3F. Accessed: 2018-04-26.
13. F. Moya, C. Gonzalez, D. Villa, and EspaCursos, *Desarrollo de Videojuegos: Arquitectura del Motor de Videojuegos* (Bubok).
14. "Unreal engine: Make something unreal," <https://www.unrealengine.com/en-US/what-is-unreal-engine-4>. Accessed: 2018-08-28.
15. "Unity. el motor de creación de contenido líder en el mundo," https://unity3d.com/es/unity?_ga=2.247112908.992394982.1502656586-594130821.1501907202. Accessed: 2017-10-26.
16. A. Menache, *Understanding Motion Capture for Computer Animation and Video Games* (Morgan Kaufmann, 2000).
17. Y. Zhuang, Y. Pan, and J. Xiao, *A Modern Approach to Intelligent Animation: Theory and Practice.*, 11 (Springer Science 2008).