



**Universidad
Europea**

UNIVERSIDAD EUROPEA DE MADRID
ESCUELA DE ARQUITECTURA INGENIERIA Y DISEÑO

Estudio del voxel art como medio artístico.

Arturo Rodríguez Severino

TRABAJO FINAL DEL GRADO DE ANIMACIÓN

Dirigido por María Teresa Barranco Crespo

Convocatoria de Julio de 2021

Deja esta página en blanco. Es la “guarda”, sirve para proteger el documento y que el tribunal haga anotaciones.

UNIVERSIDAD EUROPEA DE MADRID
ESCUELA DE ARQUITECTURA INGENIERIA Y DISEÑO

Estudio del voxel art como medio artístico.

Arturo Rodríguez Severino

PROYECTO FINAL

DEL GRADO DE ANIMACIÓN

Dirigido por María Teresa Barranco Crespo

Convocatoria de Julio de 2021

Agradecimientos

Agradecimientos a mis dos tutores durante este proyecto, María Teresa Barranco y Guillermo Castilla.

Índice

Capítulo 1. Introducción, descripción y objetivos.	9
1.1 Introducción	9
1.2 Descripción general del proyecto	9
1.3 Objetivo general	9
1.4 Objetivos específicos	10
Capítulo 2. Metodología.	10
2.1 Análisis de estudio como método de investigación	10
2.2 Errores tempranos en el aprendizaje	15
2.3 <i>Pixel art</i> , comparación directa	17
Capítulo 3. Estado de la cuestión, referentes.	19
3.1 Artistas referentes del medio	19
3.2 <i>Voxel art</i> en videojuegos	29
3.3 <i>Voxel art</i> en el cine	36
Capítulo 4. Estudio sobre el voxel art.	38
4.1 Modelado de <i>voxel art</i>	38
4.2 Importancia del texturizado en <i>voxel art</i>	40
4.3 Importancia de la resolución de voxels	41
4.4 Rigging de personajes con <i>voxels</i>	46

4.5 Diferentes técnicas de animación en <i>voxel art</i>	50
4.6 Comentarios sobre iluminación en <i>voxel art</i>	51
4.7 Estilos de <i>voxel art</i>	53
4.8 Editores de <i>voxels</i> más importantes	58
4.9 Voxel art no digital	60
Capítulo 5. Diseño artístico de Voxel Knights.	62
Capítulo 6. Presentación del producto.	64
Capítulo 7. Conclusiones.	69
Glosario	70
Bibliografía	72
Anexo I. Índice de figuras.	77
Anexo II. Utilidad de los <i>voxels</i> fuera del <i>voxel art</i> .	77
Anexo III. Requerimientos para licencias de música.	82

Resumen

El documento presentado trata de solventar los problemas que sufre el *voxel art*¹ como consecuencia de ser un medio artístico relativamente poco explorado: la falta y dispersión de la información al respecto.

Para ello, se ha realizado una investigación exhaustiva, escogiendo la información más relevante y unificándola en este trabajo, para así crear un texto de referencia para aquellos que quieran aprender del tema.

En este texto, se presentarán tres piezas de información: el progreso en el aprendizaje del medio por parte del autor, la búsqueda y unificación de información en el medio, y el diseño y creación de *assets*² para ejemplificar la información dada.

Se tratarán aquellos procesos del desarrollo 3D que diferencian al *voxel art* de otros medios, y aquellos procesos fundamentales en la creación de *assets* utilizando *voxels*. Adicionalmente, se hablarán de los usos del *voxel*³ más allá de los artísticos.

Para el desarrollo del proyecto, se han utilizado los siguientes *software*: el editor de *voxels Qubicle* (Minddesk Software GmbH, 2016) para el modelado, el programa de desarrollo de gráficos 3D *Autodesk Maya* (Alias Systems Corporation, 1998) para las animaciones, y el software de render *Arnold* (Solid Angle, 1997) para la presentación del render final de los *assets*.

Palabras clave

Voxel, animación, Qubicle.

¹ Voxel art: Medio artístico en el que se utiliza el *voxel* como elemento base.

² *Asset*: En este contexto, *asset* hace referencia a todo elemento resultante del trabajo de desarrollo 3D del proyecto: modelos, animaciones, *rigs*, etc.

³ Voxel: Unidad cúbica tridimensional, equivalente a lo que representa un píxel en 2D.

Abstract

The document presented tries to solve the problems that the voxel art suffers as a consequence of being a relatively unexplored artistic medium: the lack and dispersion of information about it.

For this, an exhaustive investigation has been carried out, choosing the most relevant information and unifying it in this work, in order to create a reference text for those who want to learn about the subject.

In this text, three pieces of information will be presented: the progress in the learning of the medium by the author, the search and unification of information in the medium, and the design and creation of assets to exemplify the information given.

Those 3D development processes that differentiate voxel art from other media, and those fundamental processes in the creation of assets using voxels, will be discussed. Additionally, the uses of voxel will be discussed beyond the artistic ones.

For the development of the project, the following software has been used: the Qubicle voxel editor (Minddesk Software GmbH, 2016) for modeling, the Autodesk Maya 3D graphics development program (Alias Systems Corporation, 1998), for the animations, and the Arnold rendering software (Solid Angle, 1997), for the presentation of the final render of the assets.

Keywords

Voxel, animation, Qubicle.

Capítulo 1. Introducción, descripción y objetivos.

1.1 Introducción

En la búsqueda de un proyecto que me resultase de interés y que tuviera cierto nivel de originalidad, me fue propuesto realizar una investigación sobre el *voxel art*. Este es un medio que no me resultaba desconocido, pero al que nunca le había dedicado una especial consideración.

Tras una pequeña toma de contacto con el tema, descubrí que había una escasez de información al respecto. La gran mayoría de esta información se reducía a tutoriales para el aprendizaje de las herramientas, sin énfasis en las decisiones que se toman en el diseño de las piezas.

Esta falta de información teórica, junto con mi creciente interés en el medio, fueron las razones que me llevaron a realizar este proyecto.

1.2 Descripción general del proyecto

El proyecto busca realizar un documento de referencia sobre el *voxel art*, un medio artístico que ha resurgido en popularidad en los últimos años.

Además, se mostrará el aprendizaje del autor como consecuencia de la investigación realizada, para finalmente desarrollar los *assets* que acompañarán la explicación de los conceptos mostrados.

1.3 Objetivo general

Reunir y reflexionar sobre la información existente, para así poder crear un estudio que sirva de guía a aquellos interesados en el *voxel art*.

Desarrollo personal en el medio del *voxel art*, para la familiarización con todos los procesos necesarios para crear *assets* de calidad, como consecuencia de la investigación anterior.

1.4 Objetivos específicos

La búsqueda y estudio de la información disponible al respecto, a partir de documentos académicos, artículos, videojuegos, películas, o a partir de la obra de los propios artistas del medio. De esta manera se busca recoger información en todo el espectro del conocimiento en el medio.

Selección y cribado de la información de manera objetiva, mostrando las diferentes posibilidades que ofrece el *voxel art* en cada uno de los procesos de creación del mismo.

Desarrollo de *assets* para la explicación de los conceptos más relevantes, y creación de una presentación visual de los mismos. Esto lleva consigo el desarrollo del propio autor, y el aprendizaje a través de la práctica de los conceptos expuestos.

Capítulo 2. Metodología.

2.1 Análisis de estudio como método de investigación

Para establecer una base sólida en el aprendizaje del tema, se tomó la decisión de comenzar con un curso por parte de uno de los artistas más relevantes del medio: Zach Soares (Soares, Voxels, s.f.).

Su curso, llamado “*Introducción al voxel art para el diseño de personajes*” (Soares, Domestika, 2020), comprende no sólo el diseño de personajes en sí, si no todas las bases del diseño en general en *voxel art*.

Incluye, además, una introducción a uno de los principales editores de *voxels*: *Qubicle*, y toda la metodología de trabajo necesaria entre *Qubicle*, *Autodesk Maya*,

y el motor de desarrollo de videojuegos *Unity* (Unity Technologies, 2005) para poder utilizar personajes *voxel art* animados en videojuegos.

En el inicio del curso, se introduce al alumno a algunos conceptos muy importantes, como pueden ser las diferencias entre el diseño en alta y en baja resolución, y el sombreado, temas en los que se profundizará más adelante.

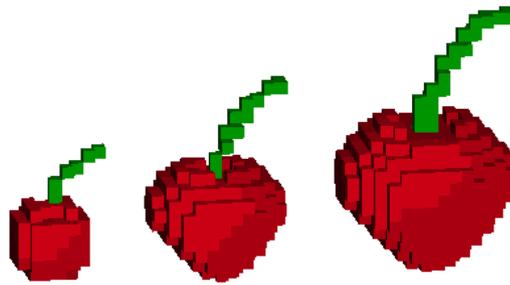


Figura 1. Comparativa de resoluciones utilizando cerezas.

Tras unas primeras lecciones, se empezaron a desarrollar proyectos de manera paralela al curso, dando pie a los primeros modelos propios, inspirados en el videojuego *Crossy Road* (Hipster Whale, 2014).

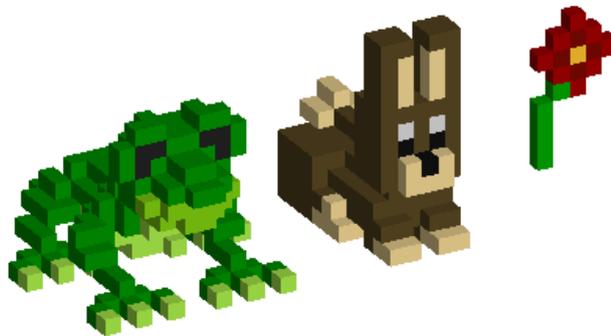


Figura 2. Primeros modelos originales del autor.

Una vez terminado el establecimiento de conceptos base del *voxel art* y la introducción a *Qubicle*, el curso pasa directamente a la realización del proyecto final: la creación de una escena animada en *Unity*.

En este proceso se modela un personaje de cero y los elementos del escenario que lo acompañarán. Tras esto, se envía el personaje a *Autodesk Maya* donde se crea el *rig*⁴ y se anima, para posteriormente realizar la composición final en *Unity*.

En este caso, siendo el proyecto una animación sin ninguna interacción por parte del receptor, se hubiera obtenido un mejor resultado utilizando el motor de renderizado *Arnold*, integrado en *Autodesk Maya*. Aun así, el realizar la escena en *Unity* ayuda a conocer el proceso que se sigue para utilizar *assets voxel art* en el desarrollo de videojuegos.



Figura 3. Proyecto final en Qubicle.

Durante la realización de esta parte final del curso, se comenzó a buscar más información sobre el tema y otros métodos de aprendizaje. Esto llevó al descubrimiento de “@VoxelDailies” (Soares, @VoxelDailies, 2014), una cuenta de

⁴ En desarrollo de gráficos 3D, el *rigging* hace referencia al proceso a través del cual se crean una serie de controladores digitales para agregárselos a un modelo 3D, lo cual permite animarlo y modificarlo fácilmente.

Twitter (Twitter Inc., 2006), actualmente abandonada, la cual proponía diariamente un tema sobre el que desarrollar una obra en *voxel art*.

Esto sirvió como motivación para crear obras en temas específicos que quizá no se hubieran explorado de otra manera, y para profundizar y mejorar en el uso de las herramientas. De todo este proceso se muestran aquí algunas de las obras resultantes:

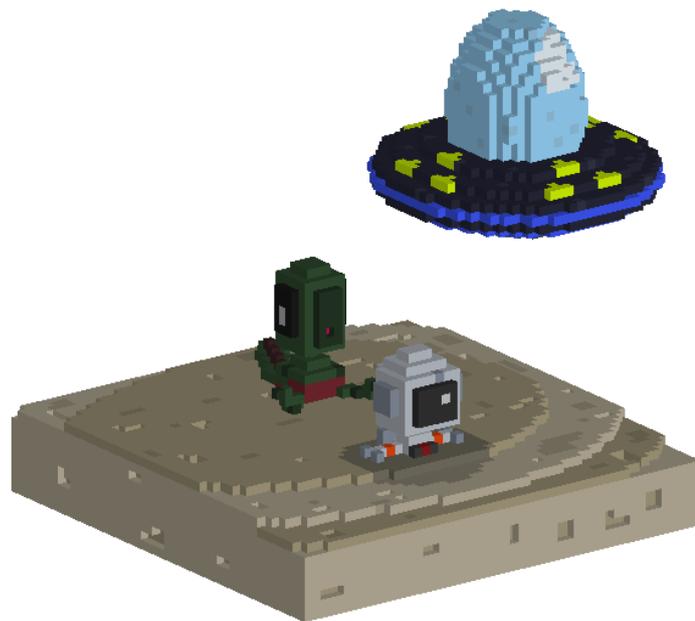


Figura 4. Diorama con la temática "moon".



Figura 5. Diorama con temática "lantern".



Figura 6. Modelo de la temática "haunted house".

Al terminar el curso, se abandonó el desarrollo de estos modelos y se pasó a la investigación en profundidad de la información disponible

Esta es la parte más importante del proyecto, además de ser de las más complejas: el *voxel art* es un medio increíblemente poco explorado a nivel académico u oficial.

Debido a esto, toda la documentación académica en relación con la temática *voxel* que se referencia en este trabajo, lo hace desde puntos de vista distintos al artístico. También se han buscado documentos oficiales en temáticas similares o comparables, como puede ser el *pixel art*⁵.

⁵ Pixel art: Medio artístico en el que se utiliza el pixel como elemento base.

La falta de información académica, o sobre la que se haya llegado a un consenso de forma oficial, obliga a una búsqueda de información que irremediamente será más sesgada, por lo que este documento intentará aportar información de manera objetiva y no absoluta.

Para intentar solventar este sesgo en cuanto a la información, se han buscado artículos y entrevistas de los que parecen ser los mayores referentes en este medio, tanto en el desarrollo de videojuegos como artistas independientes.

Estos “referentes” se han seleccionado teniendo en cuenta su popularidad, o directamente por recomendación de artistas, como los que se pueden encontrar en la entrevista a Zach Soares: “¿Qué es el Voxel art?” (Criscuolo, 2020), donde comenta algunos de los artistas de los que personalmente toma inspiración.

Otros lugares donde se han buscado referentes son páginas como *VoxelMade* (Simar, 2017) o *MegaVoxels* (Mega Voxels, s.f.), dedicadas específicamente a la muestra de arte de la comunidad online y a la educación en el medio.

Sitios web como *Artstation* (Teo, 2014) o *Twitter* también se han utilizado como lugar de búsqueda, ya que artistas *voxel* tienden a compartir o seguir obras de otros miembros de la comunidad, dando lugar a nuevos descubrimientos.

2.2 Errores tempranos en el aprendizaje

Durante el aprendizaje práctico en el medio, como es natural, se cometieron numerosos fallos en el diseño de diferentes piezas. Aquí se propondrán algunas de estas piezas para explicar brevemente dos de los errores más comunes.

2.2.1 Elección errónea de resolución de *voxels*

El fallo más común que se cometió, es no elegir la resolución correcta de *voxels* para algunas piezas, tanto por exceso, como por falta de ellos.

En la figura que se ve a continuación, se utilizó una resolución demasiado baja: Se introdujo un gran número de elementos en un espacio de 30x30x30 *voxels*. Esto provoca que la mayoría de partes sean vagamente reconocibles, en especial la araña, y el esqueleto del suelo. Ni si quiera la contextualización concedida por el resto de elementos puede suplir la falta de detalle.



Figura 7. Diorama con resolución escasa.

En la siguiente figura, el error es el contrario, un uso excesivo de *voxels*. Pese a que usar muchos *voxels* no es un problema de serie, el uso excesivo de detalle en esta pieza hizo que su silueta fuese difícilmente reconocible, y un poco confusa visualmente. Además, la postura tampoco es la ideal, ya que también empeora aún más la silueta.

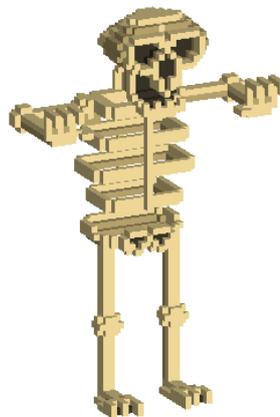


Figura 8. Esqueleto con demasiado nivel de detalle.

2.2.2 Personajes difícilmente “*riggeables*”

Otro error que se cometió, fue crear un personaje que no diera demasiadas opciones en lo que al *rig* se refiere. Con esto nos referimos concretamente a las articulaciones, en las cuales se añadió demasiado detalle.

Esto, sumado al bajo grosor de las mismas, provocaría deformaciones extrañas en caso de intentar doblarlas. Obviamente esto no sería un problema si se quiere utilizar un estilo de animaciones en las que no se doblen las articulaciones, pero esa no fue la intención a la hora de diseñar el modelo.



Figura 6. Personaje con temática “zombi”.

2.3 Pixel art, comparación directa

El *pixel art* es el medio a partir del cual se desarrolló el *voxel art*, esto se hace bastante obvio cuando se toma en consideración que un *voxel* es, al fin y al cabo, un *pixel* tridimensional. La relación de ambos medios es tal, que en muchos ámbitos aún se llama a los *voxels*, “*pixels*”, o “*pixels 3D*”.

En este apartado, se comentarán las principales disciplinas que ambos medios tienen en común, y las que no, y por qué esas disciplinas no funcionan necesariamente al introducir el factor tridimensional.

La mayor parte de la información de este apartado ha sido sacada del trabajo “*Pixel art: Estética de la necesidad o elogio del medio*” (Maravall Llagaria & Martín Martínez, 2015), y de la página web de uno de los artistas de videojuegos con estética *pixel art* más relevantes, Pedro Medeiros (Medeiros, 2020).

Lo primero en común: la existencia de ambos medios nació por necesidades técnicas, en su momento de origen eran las únicas maneras de crear imágenes digitales y objetos tridimensionales respectivamente.

La preocupación por las resoluciones es el segundo punto donde coinciden, en ambos es un factor mandatorio a la hora de diseñar y plantear los modelos, de hecho, es lo primero que hay que contemplar antes de comenzar una pieza.

Por otro lado, la manera en la que la “construcción” de las formas y el texturizado se intercalan para poder crear imágenes claras es muy similar. Sin ir más lejos, algunos artistas optan por mezclar ambos medios, como es el caso del artista Henry Dan (Dan, 2019)

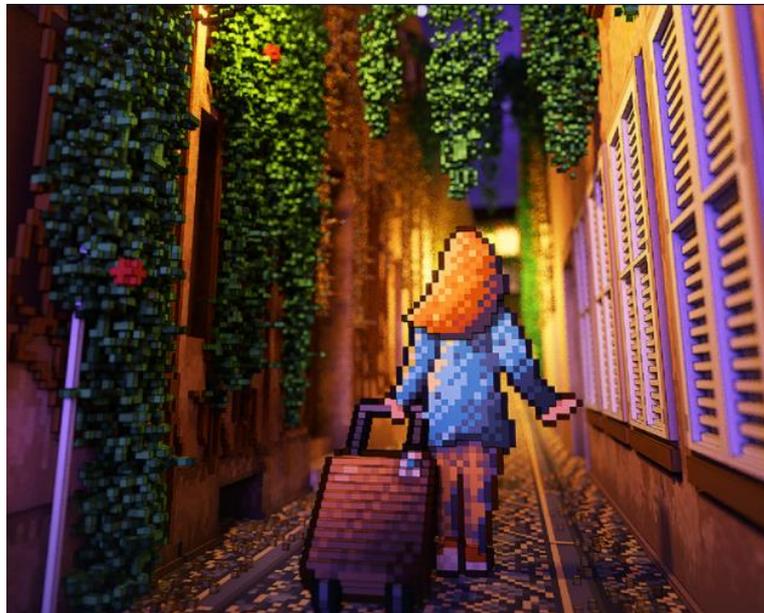


Figura 9. “Summer in a Strange City”, por Henry Dan.

El otro punto que tienen en común, es su irremediable conexión con el estilo “retro”, al que ambos medios evocan en muchos casos. Existen múltiples estilos dentro de ambos que no recuerdan al “retro”, pero sí que se suelen asociar, especialmente por la gente que no tiene mucho conocimiento del *pixel art* o del *voxel art*.

Por esta asociación con el “retro”, ambos medios están teniendo mucho éxito en el mundo de los *Non Fungible Tokens*⁶(*NFTs*) y las criptomonedas. En este mercado, que ha crecido enormemente desde principios de 2021, estos dos medios proliferan más que el resto de obras artísticas digitales.

Actualmente, uno de los mayores mercados de compra-venta de *NFTs voxel art* es *The Sandbox* (Pixowl, 2021), el cual es un videojuego con un editor para crear juegos dentro de él, y en el que se pueden utilizar los *assets* comprados en la tienda virtual que él mismo incluye.

Capítulo 3. Estado de la cuestión, referentes.

3.1 Artistas referentes del medio

A continuación, se hablará de los que se ha considerado que son los artistas más relevantes en el *voxel art*, y el porqué de ello. Algunos de ellos no utilizan su nombre propio en las redes así que se utilizará el nombre de usuario que utilicen con mayor frecuencia.

Se han buscado artistas que destaquen en algún campo en particular, para tratar las diferentes maneras que existen de afrontar el diseño con *voxels*, y las metodologías de trabajo de estos artistas en caso de estar disponibles.

⁶ *Non Fungible Token (NFT)*: es un elemento criptográfico único y que no puede ser copiado, por lo que solo puede existir un dueño para un NFT determinado.

Mari K.

Mari K. (K, s.f.) es una artista especializada en el modelado de dioramas de gran tamaño y con un inmenso nivel de detalle.



Figura 10. "Sea Tale", por Mari K.

El tamaño de sus obras es tal, que vistas a cierta distancia pueden confundirse con modelos 3D tradicionales en algunos casos. Otra de las virtudes de esta artista es su capacidad para generar formas que realmente parecen orgánicas y naturales con *voxels*, lo cual es universalmente una de las mayores debilidades del medio.

Esto lo consigue en parte creando obras con cantidades ingentes de *voxels*, pero también haciendo uso del contraste entre formas rígidas, como piezas mecánicas o rocas esculpidas, frente a elementos con mucha más variabilidad, como vegetación o arena.



Figura 11. "Happy place II", por Mari K.

Para reducir la carga de trabajo, y así poder producir de una manera más eficiente, se sirve de una biblioteca de formas ya preconstruidas propia, y de herramientas para generar “ruido⁷”, las cuales además agregan mucha rugosidad y “textura” a sus obras.

Esto se puede observar en vídeos en los que muestra su proceso de trabajo, como por ejemplo en “*Secret Hiding Place - Voxel Art Timelapse*” (K., 2020).

⁷ Ruido: en este contexto hace referencia grandes cantidades de voxels situados de manera aleatoria, en general para simular partículas o rugosidades.

Antoine Lendrevie

Antoine Lendrevie (Lendrevie, @Sir_carma, 2011) es otro artista que trabaja con escenas de gran tamaño, pero a diferencia de Mari K., no las lleva hasta el mismo nivel de detalle, sino que plantea escenas más estilizadas que recuerdan más a lo que alguien esperaría al ver *voxel art*.

El mayor factor diferenciador de Antoine, es el uso de animaciones para dar vida a sus escenas, y a buscar posiciones de cámara que otorguen cierto dinamismo a la imagen. Esto se puede ver, por ejemplo, en su obra “*Cyberpunk City*” (Lendrevie, Behance, 2020)



Figura 12. "SciFi City", por Antoine Lendrevie.

Otra línea de obras diferentes desarrolladas por este autor, es la llamada “*Carved Cities*” (Lendrevie, Behance, 2019), en el que realiza una serie de modelos mucho menos orientados a crear un escenario “realista”, dando prioridad a generar una estética “limpia” y estilizada.

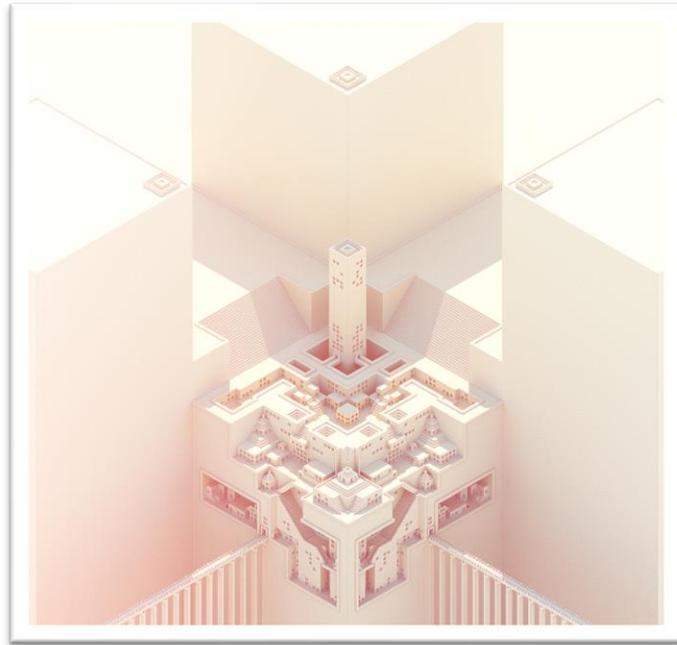


Figura 13. Pieza sacada de la línea "Carved Cities", por Antoine Lendrevie.

La obra hace uso de la existencia exclusiva de ángulos rectos, como consecuencia del trabajo con *voxels*, para generar ese nivel de estilización extremo.

Marco Peschiera

El trabajo de Marco Peschiera (Peschiera, s.f.) destaca en dos ámbitos, el primero siendo sus modelos de criaturas. Con una resolución no muy alta, usando apropiadamente el color, y reconociendo las formas más importantes de cada animal, crea modelos fácilmente legibles.



Figura 14. Pez espada por Marco Peschiera.

El segundo ámbito en el que Peschiera destaca, y por el cual es más conocido, es por la capacidad de dar vida a dichas criaturas a través de animaciones. Estas animaciones son similares a las que se realizan en la animación tradicional, modelando al personaje en cada posición al completo.



Figura 15. Animación de lobo corriendo por Marco Peschiera.

La animación de la figura anterior, y muchas otras, se pueden ver en el videojuego *Fugl* (Team Fugl, 2017), del cual es director artístico, y que se tratará en un apartado posterior.

Se puede ver en profundidad el proceso detallado que este autor lleva para todas sus animaciones en el artículo “*Art Design Deep Dive: The gorgeous voxel creatures of Fugl*” (Peschiera, Gamasutra, 2019). En este artículo, Peschiera trata temas como escala, forma, referencias, o cómo determinar las poses de cada animación.

William Santacruz

William Santacruz (Santacruz, s.f.) es un artista 3D especializado en el modelado *low poly*⁸ y en el *voxel art*. En este último, destaca por sus obras homenaje a videojuegos.

⁸ Low poly: Estilo de modelado 3D en el que se utilizan muy pocos polígonos para crear modelos muy estilizados.



Figura 16. "Bomberman" por William Santacruz.

La relevancia del trabajo de Santacruz viene de su capacidad para “traducir” al *voxel art*, de una manera acertada, las imágenes o modelos de las obras a las que hace homenaje.

Cuando se intenta hacer un trabajo de estas características, es posible caer en la tentación de intentar recrear con gran precisión aquello que se está imitando, lo cual, debido a las limitaciones del *voxel art*, en muchos casos solo es posible con resoluciones muy altas.

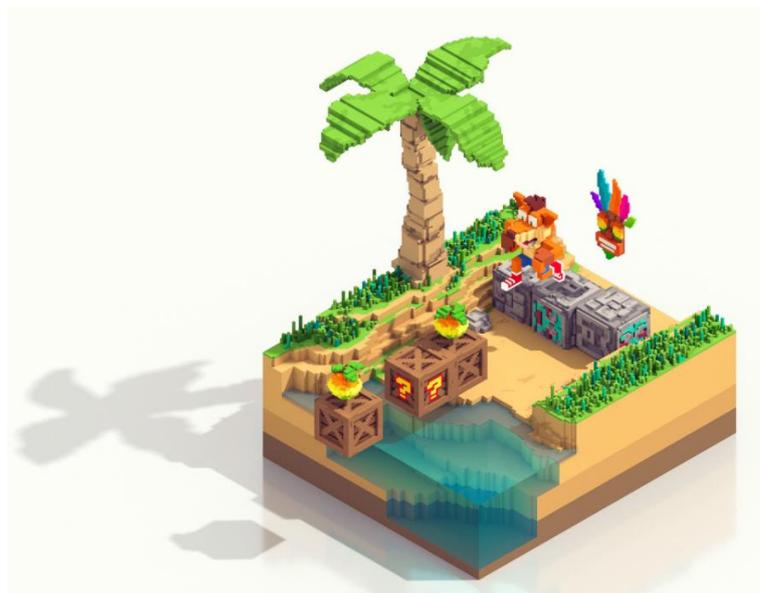


Figura 17. Crash Bandicoot, por William Santacruz

Para poder obtener resultados satisfactorios con resoluciones no excesivamente altas, Santacruz, aparte de utilizar unas formas apropiadas, hace uso del color para aumentar la legibilidad de sus modelos, resultando en obras como las que se han expuesto.

Studio Picko

Studio Picko (Studio Picko, 2020) es el único nombre de esta lista que no pertenece a un artista individual, si no a un estudio. Además, pese a trabajar en digital inicialmente, las obras finalizadas del estudio no son renders o animaciones 3D digitales, sino animaciones *stop motion*⁹ a partir de objetos 3D impresos.



Figura 18. Modificación del escenario de manera manual en Studio Picko

Para la realización de sus proyectos, desarrollan los modelos en el editor de voxels *MagicaVoxel* (Ephtracy, 2015), a continuación, los convierten a un formato apto para una impresora 3D, y finalmente los imprimen y montan los escenarios.

⁹ Stop motion: Técnica de animación en la que cada fotograma se establece y genera individualmente, ya sea mediante fotos, dibujos o renders.

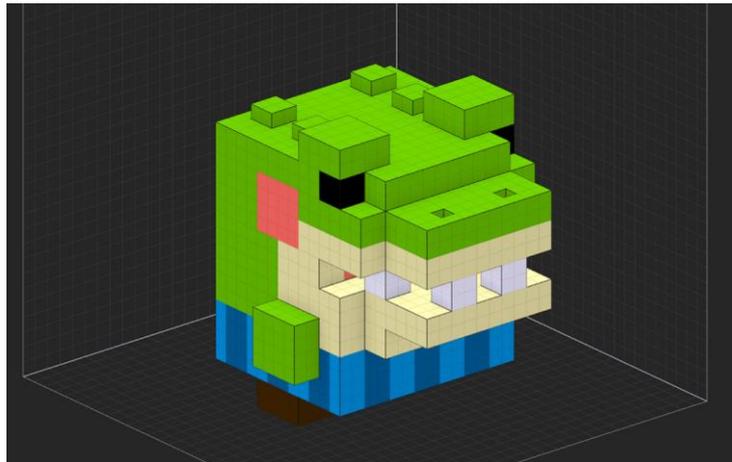


Figura 19. Modelo de personaje en MagickaVoxel, por Studio Picko.

La obra más importante del estudio, y de la que se han estado mostrando imágenes, es una serie de vídeos que giran en torno al personaje “*Akku*” (Studio Picko, 2020), en la que se puede ver el nivel del estudio y la cantidad de trabajo que vuelcan en este tipo de obras.

Estudiando el diseño de “*Akku*”, que se puede ver en la *Figura 16*, se podría hacer la observación que se están usando ocho veces más *voxels* de los necesarios (dos por cada dimensión). Esto viene de que aparentemente el propio personaje está formado por cubos de $2 \times 2 \times 2$ *voxels*.

La realidad es que este aumento de la resolución existe únicamente para proporcionar la opción de añadir una serie de detalles pequeños, pero esenciales, al diseño de “*Akku*”: los agujeros de la nariz y las escamas de su cabeza, que rompen este “molde” de $2 \times 2 \times 2$ *voxels*.

Se comenta esto para anticipar, una vez más, lo que se verá en el apartado de resoluciones, sobre la importancia de escoger la apropiada para cada obra.

Zach Soares

Zach Soares (Soares, Voxels, s.f.) es de las personas más reconocidas de la comunidad *voxel*, entre otras cosas por la cantidad de contenido educativo que ha ofrecido al respecto.

De hecho, es de las pocas personas que tiene contenido publicado que trata a un nivel teórico el *voxel art*, publicando varios artículos en la página *Gamasutra* (UBM plc, 1997). Todos ellos han resultado de gran valor para la investigación de este estudio.

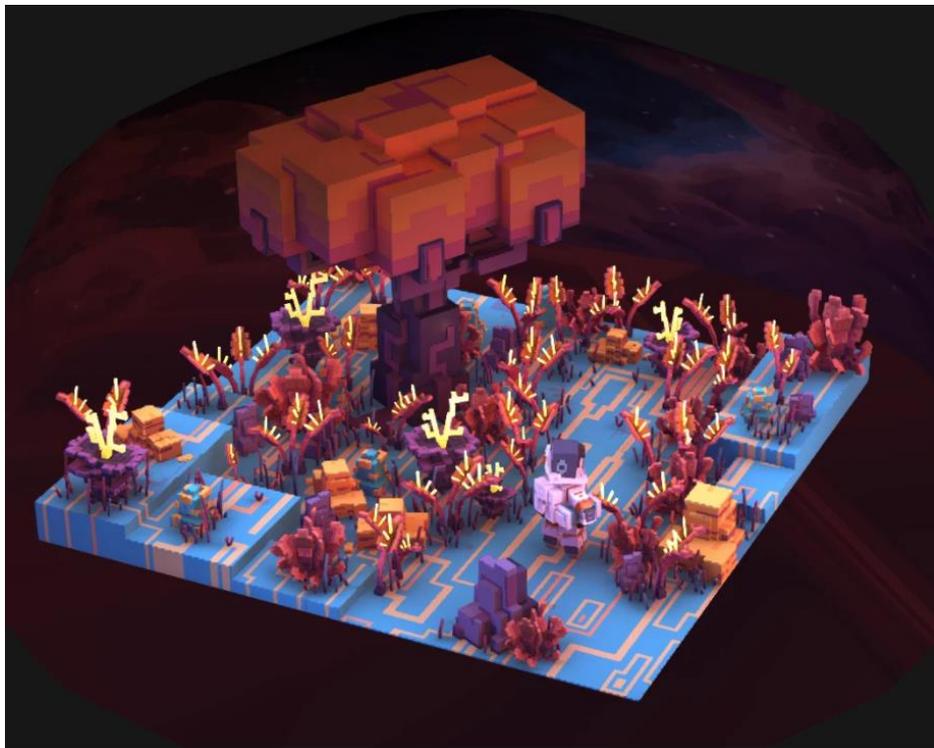


Figura 20. "Astro", por Zach Soares.

El estilo de Soares, el cual él mismo trata en el artículo “*Voxel art: Reducing the Grebbles*” (Soares, Voxel Art: Reducing the Grebbles, 2017), es un estilo muy plano y angulado, en el que existe muy poco “escalonamiento” o intento de generar curvas en las formas. Para complementar esta simpleza en las formas, que otorga un aspecto similar al del estilo *low poly* del desarrollo 3D tradicional, Soares utiliza

diferentes tonalidades de los colores del modelo para contornear y ayudar a la legibilidad de la imagen.



Figura 21. Búfalo por Zach Soares.

Esto surge como consecuencia de sus primeros trabajos en *voxel art*, en torno a 2013, una época durante la cual existían muchas más limitaciones técnicas (aquel que quisiera utilizar un editor de *voxels* tenía que programarlo él mismo), donde no tener unos modelos estilizados visualmente suponía multitud de problemas.

3.2 Voxel art en videojuegos

De la misma manera que se han buscado los artistas independientes más relevantes del medio, se comentarán aquí algunos de los videojuegos más populares que utilizan *voxel art*, y cómo hacen uso de él para enriquecer la experiencia de juego.

Minecraft

Minecraft (Mojang Studios, 2011) es uno de los principales responsables de popularizar el medio, tanto, que a veces el *voxel art* es llamado “estilo *Minecraft*”.

En una primera impresión, la decisión de utilizar este medio artístico para el juego puede parecer plenamente por preferencia personal, o debida a limitaciones artísticas del autor. Pese a que esto puede ser en parte cierto, hay que resaltar cómo el uso de *voxels* es fundamental para su funcionamiento.



Figura 22. Minecraft, por Mojang Studios.

Minecraft se puede catalogar de muchas maneras, pero por encima de todo es un juego *sandbox*¹⁰, esto se justifica dando al jugador la capacidad de modificar todo el mundo a su alrededor: eliminando trozos del terreno, utilizando los materiales que se encuentra para crear herramientas con las que interactuar con el entorno, o construyendo.

¹⁰ Los juegos del género *sandbox* son aquellos en los que se permite gran libertad al jugador, modificando hasta el propio entorno del juego para completar sus objetivos.

Aquí es donde se hace tan relevante el uso de *voxels*: el uso de unidades de construcción rígidas, ayuda enormemente a la construcción, ya que todo se ajusta perfectamente, como las piezas de un puzzle. En un juego 3D tradicional, es complicado que el jugador modifique, por ejemplo, el suelo, ya que habría que modelar específicamente el suelo, sin el trozo eliminado, para cada porción que se elimine.

En *Minecraft*, esto deja de ser un problema porque todo el mundo está creado a partir de una unidad de tamaño rígida: el “bloque”, el cual es el tamaño fijo de cualquier pieza de terreno: el mundo entero está ajustado en una matriz de tamaño infinito.



Figura 23. Las posibilidades de construcción en Minecraft son infinitas. Imagen sacada de Youtube (Rake, 2021)

Existe aún otro factor que hace de este juego el mayor referente de *voxel art* en el mundo de los videojuegos: la posibilidad de construir libremente en el modo “creativo”¹¹.

¹¹ El “godmode” (modo dios) conocido como modo creativo en *Minecraft*, es una modalidad en la que se da al jugador poder absoluto en el juego, eliminando los retos que normalmente se presentan. En *Minecraft*, es un modo en el que el jugador tiene recursos infinitos, puede volar, y es invulnerable.

Este modo, entre otras cosas, se comporta como una versión simplificada de un editor de *voxels*, sin la mayor parte de las prestaciones que simplifican el trabajo, pero para alguien desconocedor del medio les permite introducirse en él.

A raíz de esto, y con la existencia de bloques de un solo color sólido en el juego, muchos usuarios han podido crear obras de *voxel art* con los mismos resultados que se obtendrían en un editor tradicional.

Teardown

Teardown (Tuxedo Labs, 2020) es otro juego de *sandbox*, pero radicalmente distinto a *Minecraft*. Mientras que en éste se manejan los *voxels* a través de grandes bloques, que a su vez se componen de miles de cubos, en *Teardown* se manipulan individualmente con físicas, dando lugar a una precisa capacidad de destrucción.



Figura 24. Teardown, por Tuxedo Labs.

En este juego, cada nivel tiene una serie de objetivos que deben de ser completados en un minuto. Antes de iniciar el nivel el jugador se encuentra en una fase de preparación, en la que tiene acceso a todo el nivel y puede modificarlo a su antojo (rompiendo paredes, creando puentes), para facilitar el cumplimiento de dichos objetivos en el tiempo deseado.



Figura 25. Ejemplo de destrucción en Teardown.

Aquí es donde el uso de *voxels* hace su aparición, ya que, al comportarse como piezas, se puede modificar el entorno de una manera mucho más precisa que en *Minecraft*, llevando la capacidad de interacción con el espacio que rodea al jugador al extremo.

En una entrevista (GDC, 2020) para la *Game Developers Conference* (UBM TechWeb, 1989), uno de los coautores, Dennis Gustafsson, hablaba de cómo los *voxels* son un elemento con el que es mucho más sencillo de trabajar que los polígonos, al menos cuando se trabaja con físicas de destrucción.

Esto permite que esta capacidad de modificación del terreno no sea una un añadido más al *gameplay*, sino el eje fundamental respecto al que el juego gira, gracias al uso de *voxels*.

Crossy Road

Crossy Road (Hipster Whale, 2014) es el precursor de los juegos *voxel* modernos, dando gran popularidad al medio.



Figura 26. Crossy Road de Hipster Whale.

Crossy Road mostró una de las fuerzas del *voxel art* para el desarrollo de juegos de móvil: las formas tan rígidas y definidas permiten mucha legibilidad en bajas resoluciones de pantalla. Esto, sumado al estilo propio del juego, que lleva al mínimo la cantidad de *voxels* por modelo, permitía muchos elementos en pantalla, legibles, y a costa de muy poco rendimiento.

Este uso tan estilizado no se utiliza únicamente por su claridad: los editores de *voxels* generalmente los exportan en un solo bloque de seis caras cuando es posible, por lo que no tener *voxels* en disposiciones “escalonadas” disminuye el número de polígonos drásticamente.

Fugl

El *voxel art* en *Fugl* (Team Fugl, 2017), solo sirve como elemento artístico. Sin embargo, cuando se está hablando de un juego centrado en la atmósfera e inmersión del jugador, y en transmitir emociones por medio de imágenes, el *voxel art* se convierte en un factor clave del juego.



Figura 27. Fugl, por Team Fugl.

Este videojuego consiste en manejar distintos tipos de aves en tercera persona, y moverte a través de un mundo generado aleatoriamente encontrándote con otros animales.

En el artículo “*How a new wave of developers are using voxels to create jaw-dropping worlds*” (Yarwood, 2020), el desarrollador líder de *Fugl*, Johan Gjestland, habla de cómo los *voxels* son especialmente buenos a la hora de dar una sensación de escala al mundo, lo cual a su vez ayuda a dar una sensación apropiada de velocidad.

Como se puede entender, en un juego en el que se es un ave volando a través del mundo, éste es un factor fundamental. Además, comenta sobre cómo los *voxels* generan terreno (como cuevas o arcos de piedra) de manera más atractiva que un mapa de altura, lo cual es subjetivo, pero es lo que influyó en su decisión.

3.3 Voxel art en el cine

En cuanto al mundo del cine, los *voxels* no han realizado prácticamente ninguna aparición en la cual sean un elemento principal, a excepción de una (al menos que haya tenido gran repercusión pública), aquí se comentará este ejemplo y otro donde se pueden trazar similitudes.

Pixels

La película “erróneamente” llamada *Pixels* (Columbus, 2015), es una película donde unos alienígenas envían videojuegos retro, hechos tridimensionales, a la Tierra para destruirla, y una serie de jugadores profesionales de videojuegos se encargan de impedirlo.



Figura 28. Diferentes carteles promocionales de la película *Pixels*.

Se alude a que el nombre es “erróneo”, porque lo que realmente compone a estos personajes de videojuegos son *voxels*, aunque se entiende que la película fuese nombrada así por razones de publicidad, ya que el término “*pixel*” está significativamente más extendido.

En esta película, aparte de los personajes y elementos de videojuegos, toda la destrucción ocasionada por ellos también está compuesta por *voxels*, todas estas decisiones aportan a la hora de dar una atmósfera “retro”.

Lego

Existen varias películas ambientadas en un universo de *Lego* (Kirk, 1932), aquí solo se referenciará a *The Lego Movie* (Lord & Miller, 2014), ya que los principios que se tratarán en este apartado afectan a todas de la misma manera.



Figura 29. Cartel de *The Lego Movie*.

Se pueden trazar ciertas similitudes entre esta película y el *voxel art*. En primer lugar, ambos están compuestos por “piezas”, por lo que los modelos, especialmente los que se ven de manera lejana, son bastante similares.

La otra gran similitud que se puede ver es entre la animación utilizada para los personajes de la película, y los de *Minecraft*, en ambos casos son torsos rígidos con piernas y brazos que giran alrededor de un solo eje, también rígidos.

Esto es extremadamente limitante a la hora de ser animado, pero a la vez le añade un grado de originalidad y unicidad que diferencia a ambas obras del resto en sus respectivos medios.

Capítulo 4. Estudio sobre el *voxel art*.

La creación de *voxel art* es, en su nivel más básico, mucho más simple que la creación de arte 3D en otros medios. Con esto, lo que se intenta expresar, es que es sencillo crear obras de cierta calidad sin tener mucha experiencia, en comparación, por ejemplo, a si se estuviera trabajando con modelado geométrico.

Esto ocurre en parte por la propia naturaleza del medio: Al estar compuesto por “piezas” individuales, es muy sencillo tomar de referencia el arte ya existente para comenzar a crear. Si se puede ver físicamente donde va cada uno de los cubos en la obra de un artista experimentado, los podemos colocar de una manera similar para tomar como referencia.

A su vez, esta ventaja puede convertirse en un problema, especialmente a bajas resoluciones: Si el número de *voxels* es muy reducido (por ejemplo, 15x15x15 *voxels*), hay un limitado número de maneras de colocar los *voxels* para que parezcan, por ejemplo, una gallina, haciendo que irremediamente cualquier gallina modelada a baja resolución recuerde a la de *Crossy Road*, o *Minecraft*.

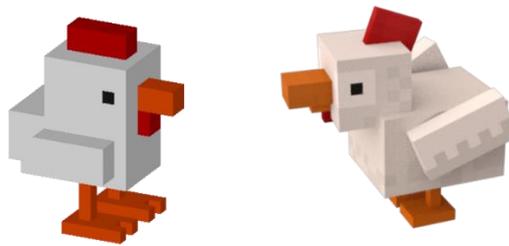


Figura 30. Gallina de Crossy Road comparada con realizada para este proyecto.

4.1 Modelado de *voxel art*

La metodología de modelado en *voxel art* es más similar al modelado orgánico o escultura 3D que al modelado geométrico tradicional, ya que se basa casi totalmente en la adición o sustracción de material, en este caso, *voxels*.

Al igual que en el modelado orgánico tradicional, se recomienda empezar por formas base a las que se añaden o sustraen *voxels*, primero en grandes cantidades, generando las formas básicas, para luego ir añadiendo más detalle progresivamente.

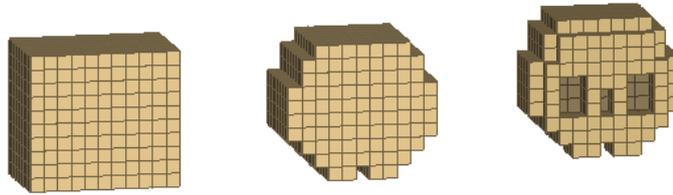


Figura 31. Fases de la creación de una calavera.

Algo importante a tener en cuenta, es el tamaño del *voxel*, en el editor serán todos del mismo tamaño, pero a la hora de pasar varios modelos a otro software, se puede cambiar el tamaño de los mismos, modificando el tamaño de los *voxels* que componen el objeto como consecuencia de ello.

Se debe tratar esto con precaución, ya que, pese a que no es algo necesariamente malo, crear una discrepancia de tamaños entre los *voxels* de una misma escena puede provocar efectos no deseados a nivel visual.

4.1.1 Modelado de personajes

La creación de personajes es la parte más compleja del modelado *voxel art*, debido principalmente a que es un apartado en el que se requiere cierta plasticidad, en concreto en la parte de las articulaciones.

Existen dos planteamientos para enfrentarse a este problema: Evadir por completo el uso de articulaciones, o lidiar con este problema y crear las extremidades de una manera determinada, lo cual se tratará en profundidad en el apartado de *rigging*.

En cuanto a la ignorar el uso de articulaciones, hay dos planteamientos posibles, que se nombrarán de acuerdo a los dos juegos más populares que los usan:

El estilo *Minecraft*, en el que se utilizan extremidades rígidas, sin doblar las articulaciones. O el estilo *Rayman* (Ancel, 1995), en el que se eliminan las extremidades, dejando exclusivamente el fin de las mismas.

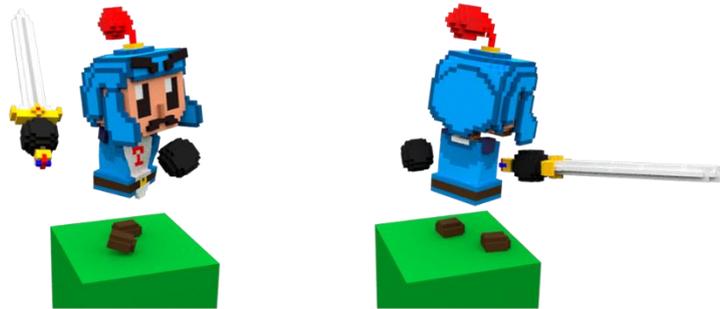


Figura 32. Personaje animado utilizando estilo Rayman sin brazos ni piernas.

4.2 Importancia del texturizado en *voxel art*

El texturizado en el *voxel art* se suele hacer de una manera diferente a otros medios, tanto por las necesidades que surgen, tanto por las posibilidades que se ofrecen: gran parte del texturizado se suele realizar de manera paralela al modelado.

Esto hace referencia a que en los editores de *voxels* se puede cambiar fácilmente el color con el que se está trabajando, frente a otros medios donde el cambiar el color del modelo es bastante menos conveniente.

La necesidad que surge, especialmente a bajas resoluciones, es la necesidad del color para ayudar a la comprensión del modelo. En algunos casos el color de cada *voxel* es decisivo a la hora de obtener un modelo legible.

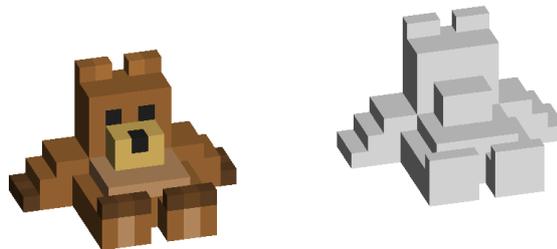


Figura 33. En bajas resoluciones, el uso del color es fundamental para la comprensión del modelo.

Además, la falta de definición inherente a los *voxels* hace que en muchos objetos el color sea lo que permite diferenciar a un objeto de otro, o las expresiones de un personaje cuyas facciones solo se marcan a través de color, muy típico en este medio.



Figura 34. Cambio de expresiones a partir del texturizado.

4.3 Importancia de la resolución de voxels

Cuando se habla de alta y baja resolución de *voxels*, no se trata de la calidad de las imágenes que estemos mostrando, si no de la cantidad de *voxels* por elemento que se estén utilizando.

Hay dos cosas que definirán la resolución de una obra en *voxel art*: el estilo buscado, dado que ciertos estilos requieren resoluciones más altas, y el nivel de detalle, ya que el elemento de menor tamaño de la escena influirá en el número de *voxels* del resto.

Por ejemplo, si se quisiera poner una mosca en la misma escena que un rascacielos, aun utilizando un diseño muy minimalista para la mosca (solamente un cubo negro), el rascacielos ya tendría que estar formado por miles de millones de *voxels*.

Esto no es cierto cuando trabajemos con distintos tamaños de *voxel*, pero al no ser una técnica muy usual, o al menos no cambiando drásticamente el tamaño de cada cubo, no se entrará en mayor profundidad.

También, ha de tenerse en cuenta, que esto sólo es un problema cuando estemos trabajando con ambos elementos a la vez en el editor de *voxels*, una vez exportado el rascacielos, si no tiene mucho detalle, tendrá un número bastante bajo de caras.

Esta diferencia de escalas tan extrema se puede observar en el juego *Cloudpunk* (ION LANDS, 2020), en el que se maneja un personaje en un mundo ciberpunk¹², rodeado de edificios entre los que se puede volar y apreciar sus dimensiones.



Figura 35. Imagen del juego *Cloudpunk*, por ION LANDS.

Por ello, mientras que en otros medios artísticos la diferencia de resolución no es un elemento demasiado restrictivo o definatorio en la forma de trabajar, en *voxel art* sí puede llegar a ser así.

Esta problemática la comparte con otro medio artístico con el que comúnmente es comparado: el *pixel art*, y por ello gran parte de las consideraciones a tener en cuenta en él, se traducen de manera bastante natural a los *voxels*.

¹² Ciberpunk: subgénero de la ciencia ficción, conocido por reflejar visiones distópicas del futuro en las cuales se combinan la tecnología avanzada con un bajo nivel de vida.

4.3.1 Baja resolución

La complejidad de trabajar en baja resolución reside en la escasez de elementos con la que se está trabajando. Al tratar con un número tan limitado de *voxels*, el diseño de los elementos se convierte en una especie de puzle, en el que el mal posicionamiento de un solo voxel puede marcar la diferencia entre una imagen ilegible y una clara.

Por esto la mayor dificultad de la baja resolución es obtener un diseño limpio y comprensible.

A continuación, se muestran los modelos originales de dos de los personajes de *Crossy Road* (Hipster Whale, 2014), y encima de ellos unas versiones donde se han reestructurado ligeramente los *voxels*. Esto se ha hecho para demostrar que, pese a que los modelos “nuevos” no son necesariamente erróneos, sí que son bastante diferentes de los originales.

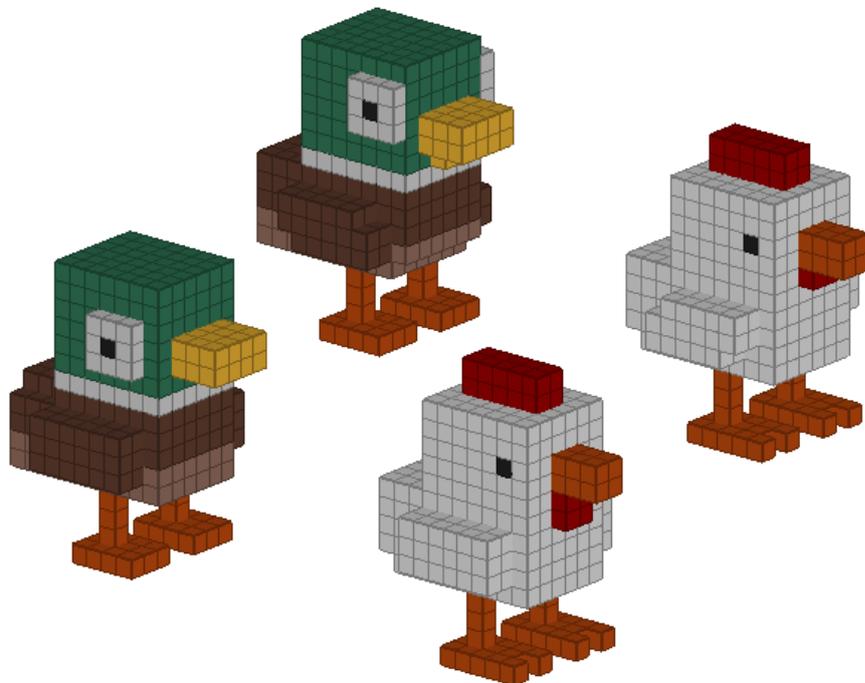


Figura 36. Modelos de *Crossy Road* con modificaciones.

Además, es extremadamente difícil trabajar con elementos que varíen en tamaño, ya que se depende mucho de la capacidad del espectador de inferir la forma de los objetos a partir de uno o dos *voxels*. Obviamente esto se puede facilitar mediante la contextualización, pero siempre se va a depender en parte de un elemento subjetivo que varía de un espectador a otro.

Por ejemplo, esto se puede observar en la siguiente figura, en la que, sin la imagen de la colmena y el árbol, sería casi imposible entender la manzana o la abeja.



Figura 37. Ejemplo baja resolución.

4.3.2 Alta resolución

En alta resolución, las dificultades aparecen en otros aspectos: mientras que ahora, un *voxel* “mal puesto” suele pasar desapercibido al haber tal cantidad de ellos, hay que trabajar mucho más cada elemento, sobre todo en aquellos con una gran cantidad de detalle.

Con esto nos referimos a que, al haber tanta cantidad de detalle, en caso de no ser utilizado correctamente, se podría crear mucho “ruido” visual, haciendo que la pieza en cuestión sea mucho más difícil de comprender.

Además, como se comentaba antes, trabajar con un nivel de polígonos muy alto puede dar problemas de rendimiento, causando bajadas de rendimiento en el editor de *voxels*.

Este problema es menos usual en el resto de programas del proceso 3D, a no ser que exportemos modelos con mucho detalle a nivel físico (en contraposición a detalle en texturizado únicamente).

Por ejemplo, el rascacielos de millones de *voxels* que se mencionaba antes podría llegar a ser incluso un solo hexaedro de seis caras si se ha querido dejar con una forma muy básica.

En el siguiente ejemplo se ha querido replicar la escena del apartado anterior, pero sirviéndose de las ventajas del trabajo a alta resolución de *voxels*, para que se observen las diferencias en el diseño de los elementos.



Figura 38: Ejemplo alta resolución.

4.4 Rigging de personajes con *voxels*

Uno de los factores más importantes a la hora hacer el *rig* un personaje usando *voxels*, es que es muy común “*riggear*” piezas enteras asignando todas las influencias a un solo hueso.

A esto, Zach Soares, en su curso “*Introducción al Voxel Art para el diseño de personajes*”, (Soares, Domestika, 2020) lo llama “*hard rigging*” (rígido) en el ámbito de los *voxels*, porque se elimina el aspecto de plasticidad al deformar la malla.

“*Soft rigging*” haría referencia al estilo tradicional en 3D de distribución de influencias sobre la malla de los huesos cercanos.

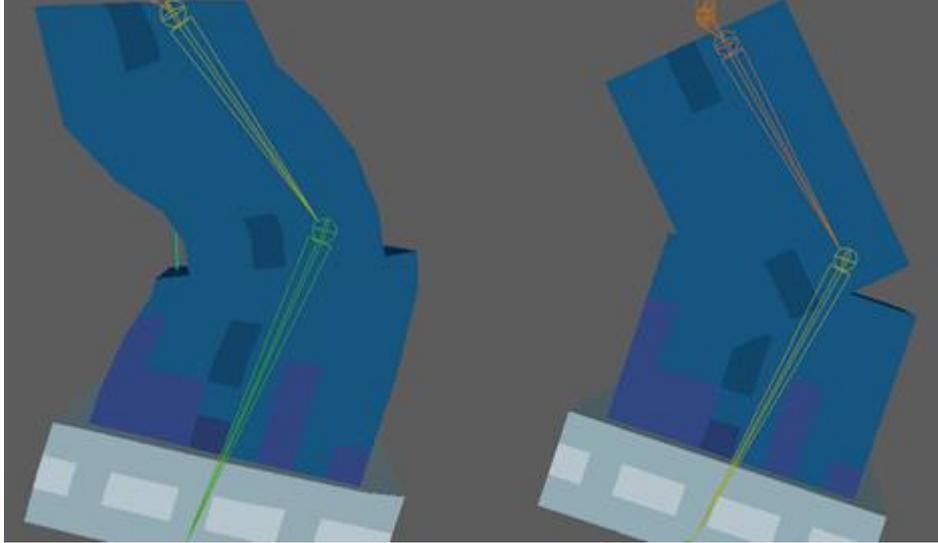


Figura 39. Comparación "*soft rigging*" vs "*hard rigging*".

En “*Hard rigging*” se elimina totalmente la deformación de las piezas, dando una sensación de movimiento “en bloque”. Este método de *rigging* es particularmente efectivo en *voxel art*, donde la rigidez natural del medio permite que este tipo de movimientos no desentonen.

Existen una serie de factores a tener en cuenta dependiendo de que estilo de *rigging* llevemos a cabo:

A la hora de usar “*soft rigging*”, hay que asegurarse de limpiar bien la malla, ya que algunos de los programas de manipulación de voxels exportan mallas con muchos triángulos (*Qubicle* en este caso).

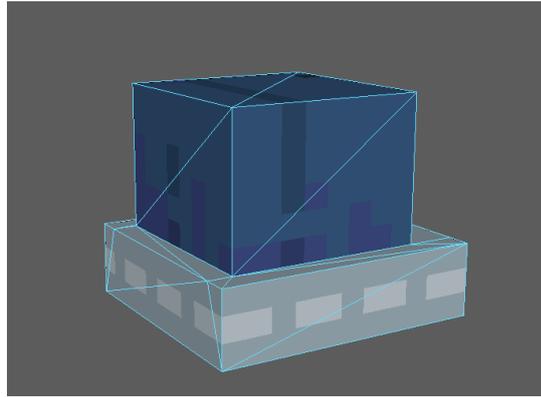


Figura 40. Ejemplo de maya recién exportada sin limpiar.

También, se recomienda utilizar extremidades proporcionalmente alargadas, y lisas, ya que tienden a dar mejores resultados en la animación, y son mucho más sencillas de “riggear”.

Con “hard rigging” existen dos pautas para mejorar la calidad del rig: La primera, es crear piezas que se superpongan unas con otras en cierta medida en cuanto a longitud, para que así no haya una ruptura al doblar por el punto que las une.

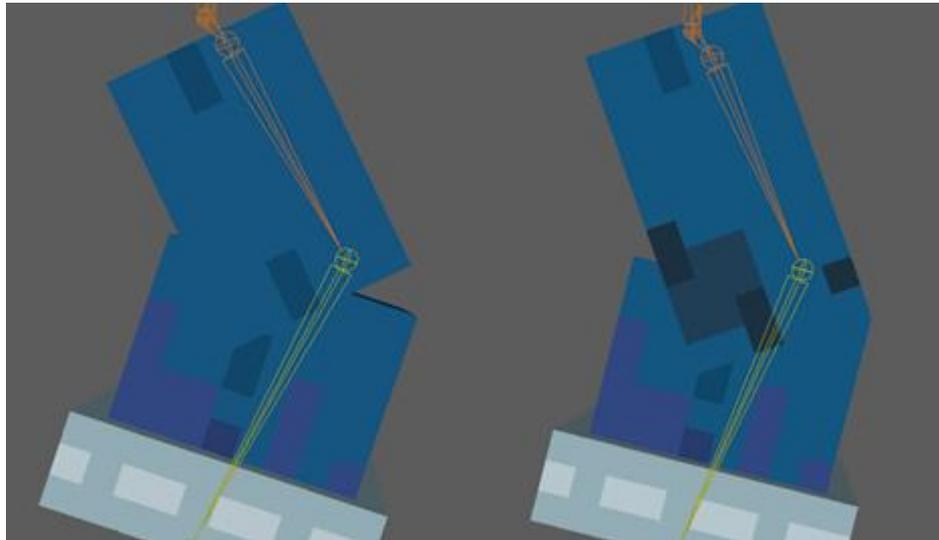


Figura 41. En "hard rigging" es recomendable evitar rupturas en la maya superponiendo elementos.

La segunda pauta es, si las piezas tienen exactamente el mismo grosor, aumentar sutilmente el grosor de una de las piezas para evitar el choque de las normales. Esto habría de hacerse en el programa de desarrollo 3D, en el caso de este proyecto, en *Autodesk Maya* (Alias Systems Corporation, 1998).

De la misma manera, en vez de aumentar ligeramente el tamaño de una de las piezas, se pueden diseñar una deliberadamente más grande que la otra, aun siendo parte de la misma extremidad.

Por ejemplo, en la siguiente figura, los pantalones de campana del personaje se utilizan a favor del diseño, para generar ese grosor extra en la conexión de la rodilla.



Figura 42. Ejemplo de ensanche entre piezas de una articulación.

Dicho esto, otra opción es combinar ambas técnicas en el mismo personaje dependiendo de la zona del mismo:

“*Hard rigging*” para las zonas físicamente diferenciadas (de ropa a piel, entre diferentes prendas, etc.), además, obviamente, de piezas que en *voxel art* no tiene sentido que se deformen nada (cabeza, palma de la mano, gafas, etc.)

“*Soft rigging*” para el resto del cuerpo (codos, cuello, rodillas, etc.).

Estos ejemplos hacen referencia a los casos más usuales, pero el estilo que se busque será el que en última instancia defina qué tipo de *rigging* se utilice y dónde.

4.5 Diferentes técnicas de animación en *voxel art*

La animación en este medio y los procesos para realizarla es similar a otros, tanto 3D como 2D, y podría considerarse en muchos casos más simple.

En cuanto a animación 3D tradicional, el *voxel art* no destaca respecto a otros medios, pero hay que tener en cuenta que, en general, la rigidez de los modelos impide que haya mucha deformación.

La ventaja del *voxel art* es que, debido a que todos los modelos están formados por múltiples piezas individuales, la utilización de animación fotograma a fotograma (en la que se cambia la forma fundamental del modelo) se puede utilizar de una manera mucho más natural que en modelos creados a base de polígonos.

Es mucho más sencillo modificar físicamente un modelo *voxel* que un modelo hecho de manera 3D tradicional. Esto se veía claramente representado en el trabajo de Marco Peschiera (Peschiera, s.f.) que se vio al principio del texto.

En la siguiente figura, se pueden observar pequeños cambios de forma y color, los cuales permiten ver la evolución de la bomba y la mecha a medida que se va aproximando a su explosión.



Figura 43. Bomba y mecha cambiando totalmente de forma y color para la animación.

Y, por último, y más importante, la capacidad de combinar ambos estilos es uno de los puntos fuertes del *voxel art*. Piezas que se deforman en conjunción con piezas que cambian en cada fotograma por completo.



Figura 44. Combinación de ambos estilos de animación entre el personaje y la espada.

4.6 Comentarios sobre iluminación en *voxel art*

En el apartado de la iluminación, el *voxel art* no se diferencia demasiado de otros medios, con la excepción de dos pequeños detalles.

Las sombras que se proyectan son muy estilizadas y marcadas cuando se utiliza luz directa, como consecuencia de la cantidad de líneas y ángulos rectos que predominan más que en ningún otro medio.

El otro detalle tiene que ver con la técnica de iluminación *flat shading*¹³. Generalmente lo que hace esta manera de iluminar es calcular el valor de luz para cada cara, haciendo que cada una aparezca con el mismo color a lo largo de su superficie.

¹³ El *flat shading* es una técnica de iluminación en la cual el valor de la iluminación se calcula una vez por cada cara del objeto.

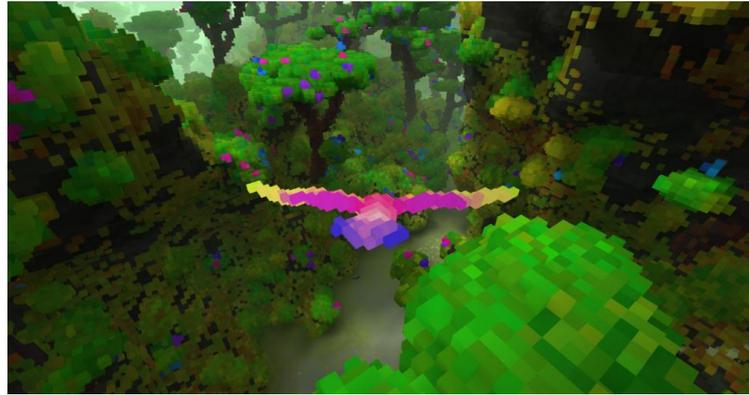


Figura 45. Iluminación flat shading en el videojuego Fugl.

Esto, combinado con un modelo en el que se hayan exportado todos los *voxels* de manera individual, da lugar a un tipo de imágenes muy características de este medio, similares a las del *pixel art*. Este estilo visual se puede alcanzar de dos maneras:

La primera es una técnica significativamente más costosa en cuanto a computación, que requiere que exportemos nuestros elementos con todos y cada uno de sus *voxels*, para luego ser iluminados uno a uno (cuando generalmente se simplifican los modelos al pasar a los motores de videojuegos o renderizado).

La otra, siendo mucho menos costosa en cuanto a computación: Consiste en iluminar manualmente, por medio de color, como se haría en *pixel art*, y luego se añaden luces ambientales, eliminando las sombras en el render. El problema de este método es que está limitado a escenas estáticas.



Figura 46: Ejemplo de estilo pixelado (iluminación realizada a mano).

4.7 Estilos de *voxel art*

Dentro del *voxel art*, podemos diferenciar numerosos estilos artísticos, pero dada a la escasez de información con respecto al medio, no existe como tal un consenso con respecto a cómo identificarlos.

Zach Soares, en su artículo “*Voxelart Styles in Video Games*” (Soares, Gamasutra, 2019), menciona cuatro de ellos. Ya se habla del estilo “*flat shaded*” en el apartado de iluminación, así que se tratarán aquí los tres restantes.

4.7.1 *Vector voxel art*

También llamado estilo vectorial, es el estilo visualmente más simple de los cuatro. Es un estilo extremadamente minimalista utilizado principalmente en el mercado de videojuegos móvil, como en el caso de *Crossy Road* (Hipster Whale, 2014).

En estos juegos, en los que se prioriza la optimización a través de reducir el número de polígonos, benefician estilos de modelado con poco detalle, tanto en forma como en textura.

El punto fuerte de este estilo es la limpieza y claridad de los elementos representados, donde otros estilos pueden resultar confusos y llenos de ruido, éste crea imágenes fácilmente legibles con muy poco esfuerzo.

Por otro lado, la mayor debilidad de este estilo es la falta de flexibilidad en cuanto a detalles en los objetos: No es posible representar objetos con ningún grado de detalle si existen otros mucho más grandes en escena.

Por ejemplo, en la figura de abajo, el tamaño del *voxel* es el del pomo de las puertas del edificio, por lo que cualquier objeto más pequeño que se quisiera representar habría de ser, al menos, tan grande como dicho pomo.



Figura 47: Ejemplo estilo vectorial

A esto hay que añadir, que, en caso de hacer un objeto del tamaño del pomo, que además se tendría que hacer con un único *voxel*.

No se debe confundir este estilo con trabajar en baja resolución de *voxels*: el estilo vectorial es un estilo de baja resolución, pero no todos los estilos de baja resolución se podrían considerar vectorial, ya que se puede trabajar añadiendo más detalle y escalonamiento en los *voxels*.

4.7.2 Greeble voxel art

Su nombre, traducido a “gravilla”, viene dado por la gran “rugosidad” o textura que se aplica en los elementos de este estilo, lo cual, sumado a la forma rígida y angulada de los *voxels*, da lugar a un estilo con mucho más ruido, sobre todo al implementar elementos curvos o que forman diagonales.



Figura 48. Ejemplo estilo gravilla

Por motivos obvios, este es un estilo que requiere resoluciones muy altas para poder existir. Pese a esto, no es necesariamente mucho más trabajoso que el vectorial, ya que los programas de edición de *voxels* suelen facilitar el trabajo a la hora de crear estas formas.

Por ejemplo, *Qubicle* y *MagicaVoxel* cuentan con la posibilidad de crear esferas compuestas de *voxels*, conos, anillos, y otros tipos de formas curvas.

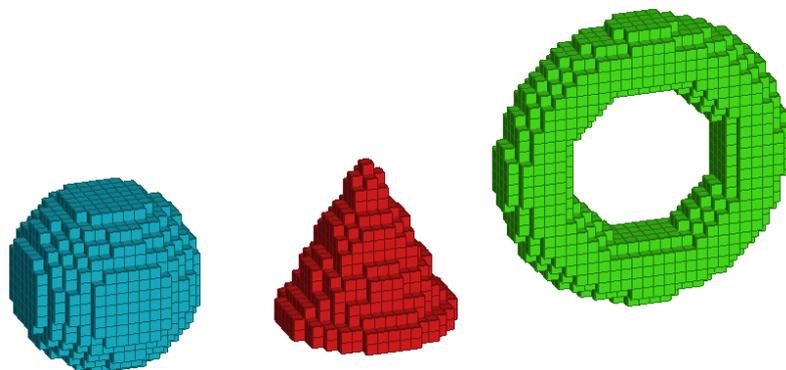


Figura 49. Figuras curvas creadas automáticamente en Qubicle.

El principal reto de este estilo es, que, si no se trata con cuidado, es fácil generar imágenes difícilmente legibles. Esto ocurre por la existencia excesiva de ruido, tanto en las partes “curvas”, al pretender hacerlas demasiado orgánicas, como en las partes planas, al abusar de los modificadores “noise”.

Por esto mismo se aconseja no abusar de los modificadores “noise”, ni a la hora de texturizar, ni a la hora de agregar textura.

4.7.3 *Blocky voxel art*

Este es posiblemente el estilo más conocido de *voxel art*, puesto que es el estilo que podemos encontrar en *Minecraft*. Este estilo se caracteriza por generar sus elementos a base de combinar formas hexaédricas grandes, y luego añadir detalle por medio de texturas de, comparativamente, bastante más resolución.



Figura 50. Animales de granja estilo “blocky”.

Por esto, este es posiblemente el estilo más simple en el que trabajar, incluso más que el vectorial. Esto se debe a que este estilo tiene las ventajas de trabajar tanto a alta como a baja resolución:

El uso de muy pocas formas básicas para generar cada elemento simplifica el modelado al mínimo, teniendo que centrarse solamente en las formas esenciales de cada modelo.

A la vez, el texturizado de alta resolución hace que sea mucho más difícil realizar fallos destacables. De hecho, en general, estos modelos suelen tener un nivel alto de ruido en las texturas, para compensar esa falta de detalle físico.

Este estilo suele combinarse muchas veces con objetos formados por una textura sobre un plano para dar detalles extra (como vegetación o pelo), a las cuales se les aplican transparencias.

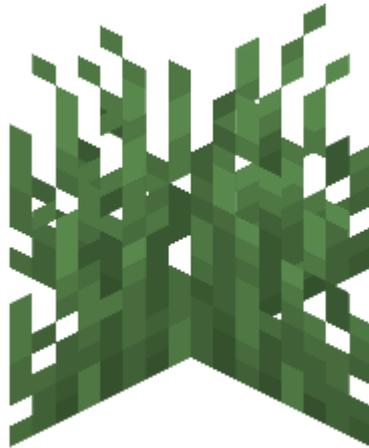


Figura 51. El césped en Minecraft está compuesto por dos planos cruzados.

Por factores como este último, se considera un estilo menos tradicional de *voxel art*, siendo incluso no considerado parte del medio por una parte de la comunidad.

Si se quieren ver otros ejemplos de este mismo estilo, el videojuego *Hytale* (Hypixel Studios, 2018) implementa este estilo con un grado mucho mayor de detalle y resolución, además de utilizar composiciones de bloques más complejas dentro de un mismo modelo.



Figura 52. Imagen promocional de Hytale, por Hypixel Studios.

4.8 Editores de voxels más relevantes

Existen múltiples editores de *voxels*, aquí se mencionarán los dos más extendidos, sus ventajas, sus desventajas, y el por qué se ha utilizado uno y no otro para la realización de este proyecto.

4.8.1 *Qubicle Voxel Editor*

Qubicle fue el primer editor en ganar popularidad en el mundo de los videojuegos, ya que fue el primer editor en facilitar la creación de modelos para poder exportarse correctamente a estos.

Con esto nos referimos a que permite exportar en los formatos .OBJ, .FBX y .DAE, esenciales en el desarrollo de juegos, mientras que *MagicaVoxel* solamente permite OBJ.

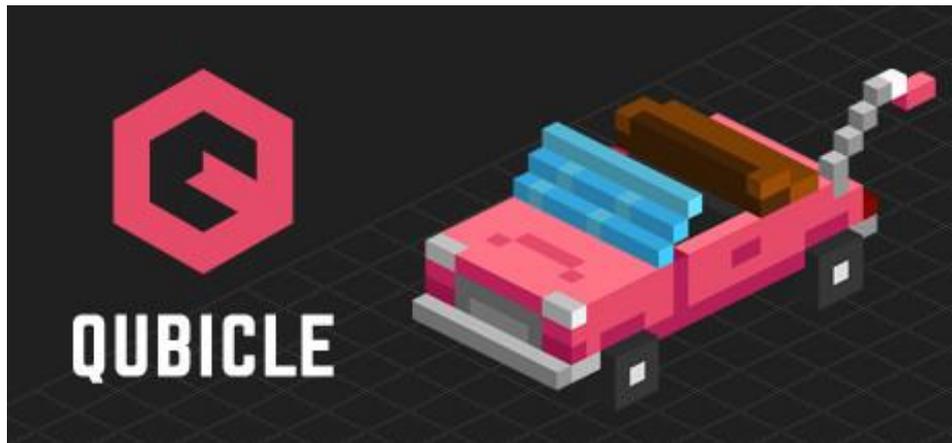


Figura 53. Imagen presentación de Qubicle

La otra ventaja de *Qubicle* es que permite trabajar con matrices de tamaño ilimitado, mientras que *MagicaVoxel* limita hasta tamaños de 126x126x126 *voxels*.

Por esto, *Qubicle* es superior para el desarrollo de videojuegos, o al menos permite una metodología de trabajo más sencilla.

4.8.2 *MagicaVoxel*

MagicaVoxel presenta varias y claras ventajas sobre *Qubicle*, la primera, y la que ha llevado a su masiva popularidad en el medio, es que es gratuito, y no requiere ningún tipo de licencia especial para monetizar obras realizadas en él.

Como consecuencia de esto, viene su segunda ventaja: la comunidad y documentación al respecto es muy extensa. Al estar mucho más extendido dado su coste nulo, existen muchos más recursos de la comunidad para el aprendizaje y la resolución de problemas.

Qubicle cuenta con documentación propia y de calidad, pero la ingente cantidad de contenido y de actividad en la comunidad de *MagicaVoxel* hacen que este último sea mucho más accesible.

MagicaVoxel es un programa de código abierto, lo cual da a la comunidad la posibilidad de crear *plug-ins* y extensiones para aumentar aún más sus posibilidades.

Por último, otra gran ventaja de este programa es que está en permanente desarrollo, recibiendo actualizaciones regulares, mientras que *Qubicle* no ha recibido ninguna actualización desde 2017. Esto implica la existencia de correcciones constantes a problemas, o la introducción de prestaciones nuevas con cierta frecuencia.

4.8.3 Conclusión y uso personal

Tras este estudio comparativo, podría plantearse el por qué se ha decidido utilizar *Qubicle* en vez de *MagicaVoxel*. La respuesta es simple: La formación que se recibió inicialmente fue en ese programa. Por ello se ha continuado trabajando en él, además, una vez pagado, *Qubicle* es un programa perfectamente capaz y versátil para crear *voxel art*.

De todas maneras, para nuevos usuarios se recomendaría empezar por *MagicaVoxel*, ya que es gratuito y de gran calidad, y si se empieza a trabajar en videojuegos de manera continuada, quizá ya debería plantearse el cambio, pero en principio no es necesario.

4.9 Voxel art no digital

Pese a las facilidades que existen para hacer *voxel art* de manera digital, algunos artistas deciden llevarlo al mundo real. Ya hablamos al principio de este texto del trabajo del Studio Picko (Studio Picko, 2020), pero no son los únicos en hacerlo.

Podemos encontrar otro ejemplo de esto en la obra del artista Kentaro Yamada, (Yamada, 2019) el cual realiza obras de madera, a base de *voxels*, imitando cabezas de taxidermia.



Figura 54. Ejemplos de la obra de Kentaro Miara.

Por otro lado, respecto a recursos para crear *voxel art*, existe un producto que permite hacerlo de manera sencilla, sin necesidad de recursos adicionales: estos son los cubos magnéticos de la marca PIXIO (PIXIO, 2017).

CREATE YOUR OWN
JURASSIC WORLD



Figura 55. Imagen promocional de PIXIO.

Estos cubos magnéticos permiten crear *voxel art* en físico, con la única limitación del peso. A pesar de esto, son bastante potentes y hacen falta piezas realmente grandes para llegar a un punto que los imanes no puedan soportar.

Capítulo 5. Diseño artístico de Voxel Knights.

De manera paralela a este proyecto de fin de grado, se desarrolló otro: *Voxel Knights* (Rodríguez, 2021), en el cual se desarrollaba un videojuego utilizando *voxel art* como medio artístico. Aquí se mencionarán los apartados esenciales de su diseño.

En el estilo visual del juego se pueden ver influencias claras de Zach Soares, tanto en el predominio de formas planas (a excepción de los personajes), tanto en el texturizado de algunos objetos, donde se usan los mismos colores con exposición diferente para marcar el contorno o para dar personalidad a las figuras.



Figura 56. Objetos decorativos del juego *Voxel Knights*.

Para este proyecto se realizaron multitud de *assets*, entre ellos el personaje soldado que ya se ha visto en anteriores figuras, y un esqueleto, para el que se utilizaron las mismas animaciones y *rig*.

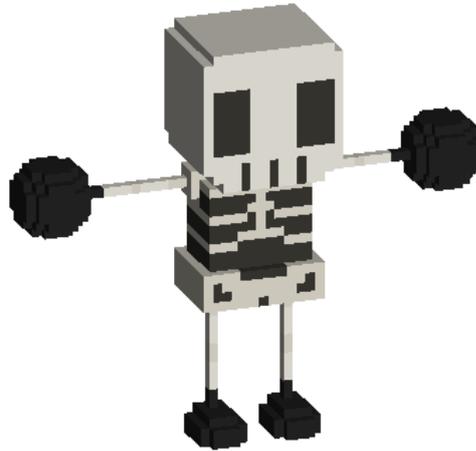


Figura 57. Esqueleto de voxel knights.

Se ha querido mencionar este proyecto en particular, porque es el primer uso del *voxel art* por parte del autor en videojuegos, y para ver de manera general los resultados que obtuvo.

Para ambos personajes se mantuvieron visibles los brazos y piernas, ya que el juego utiliza una vista isométrica lejana, y se quiere mantener la claridad visual durante las animaciones: no conectar los pies y las manos con el torso hacía que fueran más difíciles de percibir.

El otro apartado importante en el desarrollo en este juego en cuanto a *voxel art*, fue la creación de la interfaz utilizando este medio, modelando los diferentes botones y paneles y realizando renders de los mismos, para mantener una estética consistente.

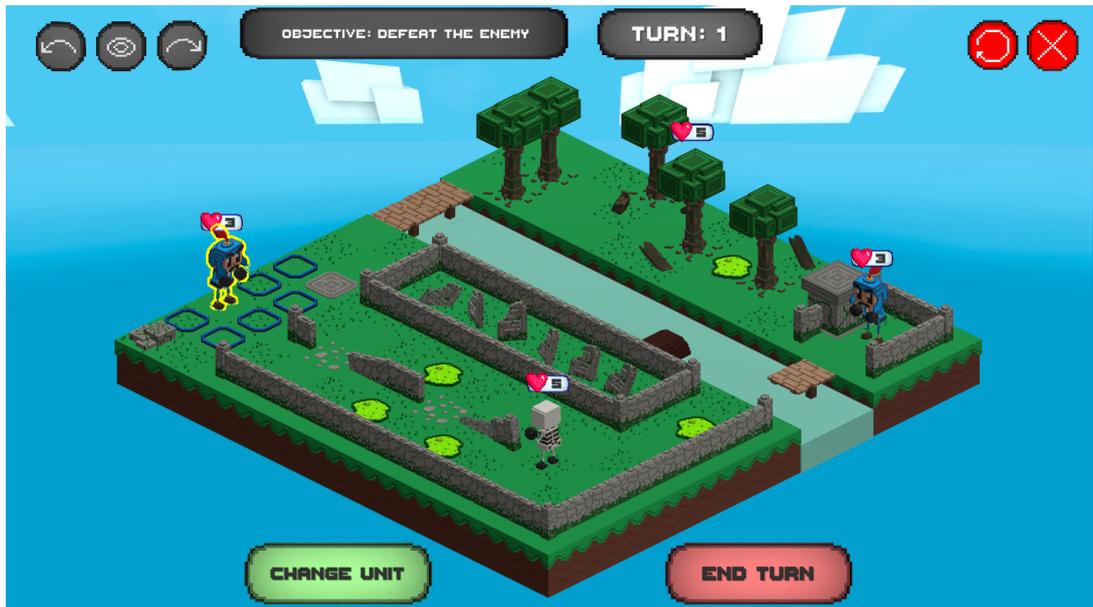


Figura 58. Pantalla de juego de *Voxel Knights*, donde se aprecian los distintos elementos.

Capítulo 6. Presentación del producto.

Para la presentación en vídeo que acompaña al proyecto, se han seleccionado las piezas que se han considerado más importantes de las que se han desarrollado durante todo el proceso, para posteriormente ser animadas, renderizadas, y finalmente compiladas en una vídeo-presentación de las mismas.

Todas estas piezas han sido maquetadas y animadas en *Autodesk Maya* (Alias Systems Corporation, 1998), para posteriormente ser renderizadas utilizando el motor de renderizado *Arnold* (Solid Angle, 1997). Finalmente han sido juntadas en el editor de vídeo *Adobe Premiere Pro CC* (Adobe, 2003).

La iluminación de las escenas es bastante simple, con luces de área de gran tamaño para crear sombras difusas o, en algunos casos como la escena de la mazmorra, prácticamente inexistentes.

A continuación, se da una breve explicación del proceso llevado a cabo en cada una de las seis piezas:

Habitación de la mazmorra: El objetivo principal de esta pieza era para ilustrar un ejemplo de *flat shading* hecho de manera manual, aunque algunos de sus elementos, como la calavera, se han utilizado para ilustrar conceptos adicionales.

Para la vídeo-presentación, la escena empieza sin la calavera y con el cáliz de vino sin volcar. En medio de esta secuencia la calavera cae del cielo volcando el cáliz, y vuelve a salir de escena, para momentos después aparecer de nuevo, dando lugar a al render que se muestra en el apartado de iluminación de este documento.

En esta escena se han combinado los dos tipos de animación de los que se hablaron en el proyecto: el fuego de la vela y el vino de la copa al volcar siguen una animación fotograma a fotograma, mientras que la calavera, y la propia copa volcando están animadas con el método 3D tradicional.



Figura 59. Diferentes fotogramas de la mesa manchándose.

Animales de granja: Estos personajes no han sido modelados únicamente al estilo “*blocky*”, sino que, además, se han animado imitando las animaciones de *Minecraft*.

Esta imitación consiste en dos partes: el movimiento rígido de las piernas y en sincronía 2 a 2 (en el cerdo); y en la manera en la que los animales miran a cámara, para posteriormente perder el interés, algo bastante frecuente en el juego.

Escenario de calle: Esta escena comprende elementos modelados en dos estilos diferentes: el “*grebble*”, que se puede encontrar en las partes central y derecha; y el vectorial, que compone la parte izquierda de la maqueta.

La animación realizada es simple en cuanto a concepto, aunque requiere considerable trabajo y planificación previa. Esta animación consiste en múltiples piezas de la escena cayendo del cielo a medida que se va montando poco a poco, mientras que la cámara gira alrededor.

El trabajo viene de la correcta selección de cortes en las piezas de las que se compone cada modelo, y de desincronizar cada pieza una a una para que no caigan ni a la misma velocidad, ni al mismo tiempo.

Se quería utilizar una animación de este tipo, en el que la escena se construyese así, ya que el *voxel art* es un medio en el que concuerda perfectamente, dadas las formas cúbicas que irremediabilmente encajan entre ellas.



Figura 60. Escenario durante el proceso de montaje.

Oso descansando: El objetivo de estas piezas es bastante obvio, ambas se crearon con el objetivo de representar una misma escena en dos resoluciones totalmente opuestas.

Para mostrar esto de una manera aún más clara, se ha utilizado una cámara que gira alrededor de la maqueta montada, fundiéndose una con la otra en mitad del proceso.

Bucle bomba: Esta escena es la que ha requerido un mayor nivel de animación y planificación. Esta pieza, aparte de combinar de nuevo ambos tipos de animación mencionados en este proyecto, pretende romper las expectativas del espectador.



Figura 61. Modelos necesarios para la animación por fotogramas.

Para esto se presenta una escena en la que la bomba cae y va enrojeciéndose hasta explotar, reiniciando todo. Después la bomba vuelve a caer, y tras la finalización de la mecha, sale un payaso de juguete de ésta, en vez de explotar de nuevo.



Figura 62. Payaso de juguete tras salir de la bomba.

El resto de objetos en la secuencia se han elegido similares a los de una oficina, por el mero hecho de utilizar una temática diferente del resto de escenas.

Animaciones Voxel Knight: Se muestra un “turnaround¹⁴” de las tres animaciones utilizadas en el videojuego *Voxel Knights* (Rodríguez, 2021), se presentan como principal animación de personajes en el proyecto.

Letras: Adicionalmente, se han añadido letras de inicio y final de la presentación, modeladas en *voxel art*, y renderizadas en formato .png.

Música: Se ha utilizado la canción “*Inspired*” (MacLeod, 2016) de Kevin McLeod para acompañar el vídeo.

¹⁴ Un *turnaround* es un tipo de vídeo en el que se muestra un objeto en todos sus ángulos a través de una cámara que gira alrededor de este.

Capítulo 7. Conclusiones.

Como conclusión: se ha desarrollado un documento que comprende extensivamente los conceptos necesarios para entender el *voxel art*, cómo trabajar con él, y poder comprender la gran variedad de posibilidades que ofrece.

Además, se ha conseguido la suficiente habilidad en el medio para crear *assets* que complementen algunos de los conceptos explicados en este estudio, y se han aprendido todos los procesos que forman parte del desarrollo de *voxel art* para videojuegos.

Esto se puede ver tanto en este proyecto, como en *Voxel Knights* (Rodríguez, 2021).

Por lo tanto, se ha desarrollado una memoria que condensa la información más relevante sobre el *voxel art*, pudiendo servir de guía a futuros interesados y que, además, muestra mi proceso de aprendizaje para ver ejemplos de los principales errores que se pueden cometer, y la velocidad de desarrollo que se puede esperar de alguien nuevo en el medio.

Glosario

asset

En este contexto, asset hace referencia a todo elemento resultante del trabajo de desarrollo 3D del proyecto

modelos, animaciones, rigs, etc., 7

ciberpunk

subgénero de la ciencia ficción, conocido por reflejar visiones distópicas del futuro en las cuales se combinan la tecnología avanzada con un bajo nivel de vida., 42

low poly

Estilo de modelado 3D en el que se utilizan muy pocos polígonos para crear modelos muy estilizados., 24

modo “creativo”

El “godmode”(modo dios) conocido como modo creativo en Minecraft, es una modalidad en la que se da al jugador poder absoluto en el juego, eliminando los retos que normalmente se presentan. En Minecraft, es un modo en el que el jugador tiene recursos infinitos, puede volar, y es invulnerable., 31

Non Fungible Tokens

es un elemento criptográfico único y que no puede ser copiado, por lo que solo puede existir un dueño para un NFT determinado., 19

pixel art

Medio artístico en el que se utiliza el pixel como elemento base., 14

rig

Proceso a través del cual se crean una serie de controladores digitales para agregárselos a un modelo 3D, lo cual permite animarlo y modificarlo fácilmente., 12

stop motion

Técnica de animación en la que cada fotograma se establece y genera individualmente, ya sea mediante fotos, dibujos o renders., 26

turnaround

tipo de vídeo en el que se muestra un objeto en todos sus ángulos a través de una cámara que gira alrededor de este., 68

voxel

Unidad cúbica tridimensional, equivalente a lo que representa un píxel en 2D., 7

Voxel art

Medio artístico en el que se utiliza el voxel como elemento base., 7

Bibliografía

Adobe. (31 de Agosto de 2003). Adobe Premiere Pro CC. San José, California, EEUU.

Alias Systems Corporation. (Febrero de 1998). Autodesk Maya.

Ancel, M. (9 de Septiembre de 1995). Rayman. París, Francia.

Coeurjolly, D., Gueth, P., & Lachaud, J.-O. (Agosto de 2018). Regularization of Voxel Art.

Columbus, C. (Dirección). (2015). *Pixels* [Película].

Crassin, C., Neyret, F., Lefebvre, S., & Eisemann, E. (27 de Febrero de 2009).

GigaVoxels: Ray-Guided Streaming for Efficient and Detailed Voxel Rendering.

Criscuolo, I. (16 de Noviembre de 2020). *What Is Voxel Art?* Obtenido de Domestika:

<https://www.domestika.org/es/blog/5698-what-is-voxel-art>

Dan, H. (2019). Obtenido de <https://art.henrydan.me/>

Ephtracy. (24 de Mayo de 2015). MagicaVoxel.

GDC. (24 de Noviembre de 2020). *GDC Plays Teardown with Dennis Gustafsson*.

Obtenido de Youtube: <https://www.youtube.com/watch?v=Z8QbY-xmbUQ>

Hipster Whale. (19 de Noviembre de 2014). Crossy Road. Melbourne, Australia.

Hypixel Studios. (2018). Hytale.

ION LANDS. (23 de Abril de 2020). Cloudpunk. Berlin, Germany.

K, M. (s.f.). *Mari.K*. Obtenido de ArtStation: <https://madmaraca.art/>

K., M. (26 de Noviembre de 2020). *Secret Hiding Place - Voxel Art Timelapse*.

Obtenido de Youtube: <https://www.youtube.com/watch?v=BAOc2qleqwc>

Kirk, O. (10 de Agosto de 1932). LEGO. Billund, Dinamarca.

Lendrevie, A. (Marzo de 2011). @Sir_carma. Obtenido de Twitter:

https://twitter.com/sir_carma?lang=es

Lendrevie, A. (6 de Abril de 2019). Obtenido de Behance:

<https://www.behance.net/gallery/78578059/Carved-Cities>

Lendrevie, A. (28 de Noviembre de 2020). Obtenido de Behance:

[https://www.behance.net/gallery/108661331/Cyberpunk-city-Voxelart-
animated-scene](https://www.behance.net/gallery/108661331/Cyberpunk-city-Voxelart-animated-scene)

Lord, P., & Miller, C. (Dirección). (2014). *The Lego Movie* [Película].

MacLeod, K. (2016). Inspired [Grabado por K. MacLeod]. Green Bay, Wisconsin, EEUU.

Maravall Llagaria, J. L., & Martín Martínez, J. V. (02 de Febrero de 2015). Pixel art:

Estética de la necesidad o elogio del medio. Murcia, Murcia, España.

Medeiros, P. (2020). Obtenido de Saint11: <https://saint11.org/>

Mega Voxels. (s.f.). *Mega Voxels*. Obtenido de

<https://www.megavoxels.com/p/home.html>

Minddesk Software GmbH. (21 de Abril de 2016). Qubicle Voxel Editor. Kiel, Alemania.

Mojang Studios. (2011). Minecraft. Estocolmo, Suecia.

Peschiera, M. (s.f.). Obtenido de https://twitter.com/marcoapc_art?s=20

Peschiera, M. (15 de Mayo de 2019). Obtenido de Gamasutra:

[https://www.gamasutra.com/view/news/342524/Art_Design_Deep_Dive_The_g
orgeous_voxel_creatures_of_Fugl.php](https://www.gamasutra.com/view/news/342524/Art_Design_Deep_Dive_The_gorgeous_voxel_creatures_of_Fugl.php)

PIXIO. (2017). *PIXIO*. Obtenido de <https://pixio.site/>

Pixowl. (15 de Mayo de 2021). The Sandbox. San Francisco, California, EEUU.

Rake. (10 de Febrero de 2021). *Minecraft - How to Build a Floating Island - Floating Island Tutorial*. Obtenido de Youtube:
<https://www.youtube.com/watch?v=gsTSpJP1lvc>

Rodríguez, A. (14 de Junio de 2021). Voxel Knights. Madrid, Comunidad de Madrid, España.

Santacruz, W. (s.f.). Obtenido de Artstation:
<https://www.artstation.com/williamsantacruz>

Simar, T. (2017). *VoxelMade*. Obtenido de <https://www.voxelmade.com/>

Soares, Z. (s.f.). Obtenido de Voxels: <https://voxels.ca/>

Soares, Z. (Diciembre de 2014). *@VoxelDailies*. Obtenido de Twitter:
<https://twitter.com/VoxelDailies>

Soares, Z. (6 de Mayo de 2017). *Voxel Art: Reducing the Greebles*. Obtenido de Gamasutra:
https://www.gamasutra.com/blogs/ZachSoares/20170605/299333/Voxel_Art_Reducing_the_Greebles.php

Soares, Z. (15 de febrero de 2019). Obtenido de Gamasutra:
https://www.gamasutra.com/blogs/ZachSoares/20190215/336661/Voxelart_Styles_in_Video_Games.php

Soares, Z. (22 de Septiembre de 2020). Obtenido de Domestika:
<https://www.domestika.org/es/courses/1404-introduccion-al-voxel-art-para-el-diseno-de-personajes>

Solid Angle. (1997). Arnold. Madrid, Comunidad de Madrid, España.

Studio Picko. (20 de Junio de 2020). *Behance*. Obtenido de https://www.behance.net/gallery/107575829/Picko-Studio?tracking_source=search_projects_recommended%7Cstopmotion%20animation

Studio Picko. (14 de Agosto de 2020). *EP01 악어가 당근을 먹는다고?! | 악구는 당근이지* 🥕 | *Farmboy Akku | Stopmotion*. Obtenido de Youtube: <https://www.youtube.com/watch?v=L4seKgo2vas&t=>

Team Fugl. (14 de Septiembre de 2017). *Fugl*.

Teo, L. (2014). *ArtStation*. Obtenido de https://www.artstation.com/?sort_by=trending&dimension=2d

Tuxedo Labs. (29 de Octubre de 2020). *Teardown*. Malmö, Suecia.

Twitter Inc. (21 de Marzo de 2006). *Twitter*. Obtenido de <https://twitter.com/>

UBM plc. (1997). *Gamasutra*. Obtenido de <https://www.gamasutra.com/>

UBM TechWeb. (1989). *Game Developers Conference*. EEUU.

Unity Technologies. (30 de Mayo de 2005). *Unity*. San Francisco, California, EEUU.

V.Kniaz, V., A. Knyaz, V., Remondino, F., Bordodymov, A., & Moshkantsev, P. (2020). *Image-to-Voxel Model Translation for 3D Scene*. Moscú, Rusia.

Xiang, Y., Choi, W., Yuanqing, L., & Savarese, S. (7 de Junio de 2015). *Data-Driven 3D Voxel Patterns for Object Category Recognition*. Boston, Massachusetts, EEUU.

Yamada, K. (2019). *Pixidermy*. Obtenido de Etsy: <https://www.etsy.com/shop/pixidermy>

Yarwood, J. (21 de Febrero de 2020). *How a new wave of developers are using voxels to create jaw-dropping worlds*. Obtenido de Gamasutra:
https://www.gamasutra.com/view/news/357225/How_a_new_wave_of_developers_are_using_voxels_to_create_jawdropping_worlds.php

Anexo I. Índice de figuras.

Figura 1. Comparativa de resoluciones utilizando cerezas.	11
Figura 2. Primeros modelos originales del autor.	11
Figura 3. Proyecto final en Qubicle.	12
Figura 4. Diorama con la temática "moon"	13
Figura 5. Diorama con temática "lantern".	13
Figura 6. Modelo de la temática "haunted house".	14
Figura 7. Diorama con resolución escasa.....	16
Figura 8. Esqueleto con demasiado nivel de detalle.	16
Figura 9. "Summer in a Strange City", por Henry Dan.	18
Figura 10. "Sea Tale", por Mari K.	20
Figura 11. "Happy place II", por Mari K.	21
Figura 12. "SciFi City", por Antoine Lendrevie.	22
Figura 13. Pieza sacada de la línea "Carved Cities", por Antoine Lendrevie.....	23
Figura 14. Pez espada por Marco Peschiera.	23
Figura 15. Animación de lobo corriendo por Marco Peschiera.	24
Figura 16. "Bomberman" por William Santacruz.	25
Figura 17. Crash Bandicoot, por William Santacruz	25
Figura 18. Modificación del escenario de manera manual en Studio Picko	26

Figura 19. Modelo de personaje en MagickaVoxel, por Studio Picko.	27
Figura 20. "Astro", por Zach Soares.	28
Figura 21. Búfalo por Zach Soares.	29
Figura 22. Minecraft, por Mojang Studios.....	30
Figura 23. Las posibilidades de construcción en Minecraft son infinitas. Imagen sacada de Youtube (Rake, 2021)	31
Figura 24. Teardown, por Tuxedo Labs.....	32
Figura 25. Ejemplo de destrucción en Teardown.	33
Figura 26. Crossy Road de Hipster Whale.....	34
Figura 27. Fugl, por Team Fugl.	35
Figura 28. Diferentes carteles promocionales de la película Pixels.....	36
Figura 29. Cartel de The Lego Movie.....	37
Figura 30. Gallina de Crossy Road comparada con realizada para este proyecto. .	38
Figura 31. Fases de la creación de una calavera.	39
Figura 32. Personaje animado utilizando estilo Rayman sin brazos ni piernas.	40
Figura 33. En bajas resoluciones, el uso del color es fundamental para la comprensión del modelo.	40
Figura 34. Cambio de expresiones a partir del texturizado.	41
Figura 35. Imagen del juego Cloudpunk, por ION LANDS.....	42
Figura 36. Modelos de Crossy Road con modificaciones.....	43

Figura 37. Ejemplo baja resolución.	44
Figura 38: Ejemplo alta resolución.	46
Figura 39. Comparación "soft rigging" vs "hard rigging".	47
Figura 40. Ejemplo de maya recién exportada sin limpiar.	48
Figura 41. En "hard rigging" es recomendable evitar rupturas en la maya superponiendo elementos.	48
Figura 42. Ejemplo de ensanche entre piezas de una articulación.	49
Figura 43. Bomba y mecha cambiando totalmente de forma y color para la animación.	50
Figura 44. Combinación de ambos estilos de animación entre el personaje y la espada.	51
Figura 45. Iluminación flat shading en el videojuego Fugl.	52
Figura 46: Ejemplo de estilo pixelado (iluminación realizada a mano).	52
Figura 47: Ejemplo estilo vectorial.	54
Figura 48. Ejemplo estilo gravilla.	55
Figura 49. Figuras curvas creadas automáticamente en Qubicle.	55
Figura 50. Animales de granja estilo "blocky".	56
Figura 51. El césped en Minecraft está compuesto por dos planos cruzados.	57
Figura 52. Imagen promocional de Hytale, por Hypixel Studios.	58
Figura 53. Imagen presentación de Qubicle.	59

Figura 54. Ejemplos de la obra de Kentaro Miara.	61
Figura 55. Imagen promocional de PIXIO.....	61
Figura 56. Objetos decorativos del juego Voxel Knights.	62
Figura 57. Esqueleto de voxel knights.	63
Figura 58. Pantalla de juego de Voxel Knights, donde se aprecian los distintos elementos.	64
Figura 59. Diferentes fotogramas de la mesa manchándose.	65
Figura 60. Escenario durante el proceso de montaje.	66
Figura 61. Modelos necesarios para la animación por fotogramas.....	67
Figura 62. Payaso de juguete tras salir de la bomba.	68

Anexo II. Utilidad de los *voxels* fuera del *voxel art*.

Durante la investigación para este proyecto, se encontraron numerosos documentos académicos dedicados a los *voxels*, pero no desde un punto de vista artístico necesariamente.

Pese a que no es el objetivo de este proyecto, se ha considerado interesante comentar tangencialmente algunos de estos estudios y su temática. Por un lado, ayudan a enfatizar la profundidad de la investigación realizada, y por otro lado es importante ver que los *voxels* tienen utilidad más allá del arte.

Los dos primeros documentos “*Image-to-Voxel Model Translation for 3D Scene Reconstruction and Segmentation*” (V.Kniaz, A. Knyaz, Remondino, Bordodymov, & Moshkantsev, 2020) y “*Data-Driven 3D Voxel Patterns for Object Category Recognition*” (Xiang, Choi, Yuanqing, & Savarese, 2015), tratan de diferentes maneras el uso de *voxels* para crear imágenes tridimensionales a partir de imágenes.

En la tesis “*Regularization of Voxel Art*” (Coeurjolly, Gueth, & Lachaud, 2018), se propone un sistema que podría ser de gran utilidad en conjunto con los del párrafo anterior, permitiendo convertir modelos compuestos de *voxels* en figuras lisas, simplificando la malla para convertirlos en modelos similares a los del modelado 3D tradicional.

“*GigaVoxels: Ray-Guided Streaming for Efficient and Detailed Voxel Rendering*” (Crassin, Neyret, Lefebvre, & Eisemann, 2009), propone un sistema de renderizado, el cual utiliza un algoritmo de *ray tracing* para renderizar a tiempo real inmensas cantidades de *voxels*.

Anexo III. Requerimientos para licencias de música.

Inspired by Kevin MacLeod

Link: <https://incompetech.filmmusic.io/song/3918-inspired>

License: <https://filmmusic.io/standard-license>