Tabla de contenido

[1 Modelado de Simulador de Vuelo 1](#_Toc239679899)

[1.1 Introducción 1](#_Toc239679900)

[1.2 Cambio de perspectiva de diseño 2](#_Toc239679901)

[1.3 Descripción del proceso 2](#_Toc239679902)

[1.4 Modelado tridimensional 3](#_Toc239679903)

[1.4.1 Modelado del plano de vuelo 3](#_Toc239679904)

[1.4.2 Texturización del plano de vuelo 5](#_Toc239679905)

[1.4.3 Modelo del avión 3D 11](#_Toc239679906)

[1.5 Ubicación de los elementos 12](#_Toc239679907)

[1.5.1 Ubicación del Plano 13](#_Toc239679908)

[1.5.2 Ubicación del Avión 15](#_Toc239679909)

[1.6 Exportación a VRML 16](#_Toc239679910)

[1.7 Integración y animación 17](#_Toc239679911)

# Modelado de Simulador de Vuelo

## Introducción

En el siguiente apartado se define el proceso de diseño e implementación del mundo virtual que representa un Simulador de Vuelo, o más bien, un video juego 3D en el que se gobierna un avión con el objetivo de sobrevolar la ciudad de Málaga.

Para este mundo virtual se cambia la perspectiva de diseño implementada en los dos mundos anteriores, aportando un nuevo enfoque en la creación de mundos virtuales con el objetivo de aproximarse, en la mayor medida posible, a la realidad más cercana que intentan emular.

Por último el mundo virtual creado y modelado se ha integrado con el interfaz BCI de navegación diseñado en el departamento de Tecnología Electrónica de la ETSIT, por lo que ha sido necesario realizar modificaciones necesarias en el software del interfaz (código Matlab), precisamente debido a la nueva perspectiva utilizada en el diseño de este entorno virtual.

## Cambio de perspectiva de diseño

Hasta ahora el diseño de mundos virtuales se ha limitado a la recreación, más o menos cercana, de recintos reales habituales. Desarrollándose en este aspecto mundos virtuales que pueden ir desde un simple laberinto de ladrillo, pasando por un parque diáfano, una calle de una ciudad, recintos cerrados como habitaciones, viviendas, o incluso entornos más conocidos como la propia ETS de Telecomunicaciones de Málaga, como se ha visto en los capítulos anteriores.

En todos estos entornos virtuales existe un elemento común a la hora de la recreación de entornos y de la propia navegación a través de los mismos:

1. *En cuanto a la recreación de entornos*: se ha utilizado una composición de formas 3D (cajas, planos, esferas, cilindros, formas más complicadas), que una vez modeladas y agrupadas, proporcionan la sensación tridimensional de encontrarnos en un entorno real.
2. *En cuanto a la sensación de mivimiento*: se utiliza un punto de vista (cámara o view point), que a través de sus avances, giros y traslaciones dentro del mundo virtual emula el movimiento que en la realidad realizaría la persona física en su exploración del entorno.

Por tanto, el objetivo de la elaboración de este mundo virtual, además de acercar el diseño e implementación de entornos virtuales más diversos, no limitándose a recintos cerrados, urbanísticos, viviendas, entornos conocidos (ETSIT), etc., muestra como una simple plano correctamente texturizado y la idónea colocación de los elementos en el espacio y de un punto de vista o cámara, puede recrear la experiencia de sobrevolar toda una ciudad, en este caso Málaga capital.

En los siguientes apartados se profundizará en esta idea mientras se explica el proceso de diseño y modelado, junto con los elementos 3D utilizados, para el desarrollo de este mundo virtual.

## Descripción del proceso

Antes de comenzar con la modelado del Simulador Virtual de Vuelo propiamente dicho, se listan a continuación las fases necesarias para su implementación.

1. Diseño e implementación del entorno virtual tridimensional.

* Modelado del plano de sobrevuelo
* Creación de la textura del terreno sobrevolado.
* Texturización del plano de sobrevuelo.
* Optimización de modelos externos utilizados.

1. Ubicación estratégica de los elementos.
2. Exportación a VRLM e integración con el interfaz de navegación existente.
3. Desarrollo software para proporcionar a la escena de movimiento.

## Modelado tridimensional

En esta fase del proceso de implementación de la escena virtual se va a crear el entorno propiamente dicho, que como se mostrará más adelante solo trata de un plano, de dimensiones considerables, estratégicamente ubicado en el espacio.

Es muy probable que no se entienda que se pretende, pero poco a poco, conforme se avance en la implementación del mundo virtual, se irá viendo el efecto que se quiere obtener.

### Modelado del plano de vuelo

El plano principal de vuelo es el elemento más importante de este mundo virtual, de hecho es el propio escenario del mundo virtual, y ésta es una de las características más importantes que diferencian a este mundo virtual de los anteriores.

Los entornos virtuales previamente descritos, a lo largo de este proyecto, constaban de multitud de formas tridimensionales creadas en el proceso de modelado que en su conjunto formaban el escenario y recreaban los entornos reales que intentan virtualizar. El proceso de modelado era la piedra angular de estos mundos virtuales, en cambio, el mundo virtual que se describe en este capítulo consta únicamente de un elemento que incluso ni siquiera se puede incluir en el apartado de elemento 3D, ya que como se ha dicho, se trata de un simple plano.

Más tarde se verá, como este elemento bidimensional puede, perfectamente, recrear una experiencia virtual tridimensional, pero esto se describirá en el apartado XXXX de este capítulo.

Pues bien, el plano es creado en 3ds otorgándole grandes dimensiones, que una vez se texturiza adecuadamente con una imagen de alta resolución, hará las veces de terreno sobre el que el avión realiza el vuelo.

La forma del plano no tiene que ser cuadrada, de hecho y dadas las características de la textura que más tarde se le aplica, su forma es rectangular. Las dimensiones que se han escogido para el modelado de este plano son de 100 unidades de ancho y 200 de largo. Como se muestra en la figura XXX, el plano se sitúa sobre el plano YX, correspondiendo la dirección del lado más largo del rectángulo a la dirección ‘y’ del eje de coordenadas.

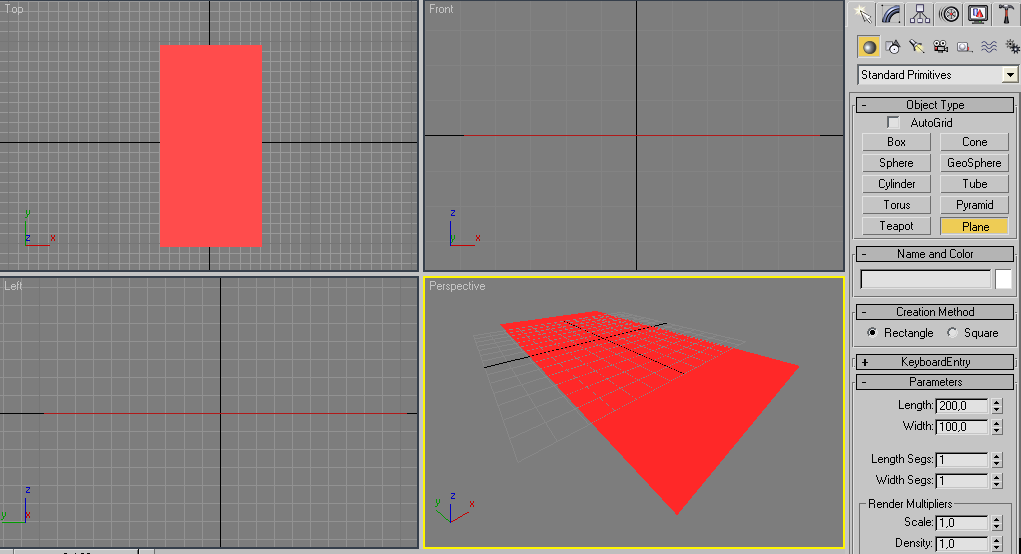


Figura : Plano principal de vuelo

Una vez más se ha utilizado la herramienta de 3ds *Standard primitives/Plane*, para confeccionar el plano, y a la derecha de la imagen se visualizan las dimensiones que se han utilizado para el mismo.

Al recrear la sensación de sobrevolar la ciudad de Málaga, es decir, el avión realice sus giros, ascensos y descensos en el mundo virtual, es necesario que los márgenes del plano nunca sean visualizados. El avión nunca puede sobrepasar los lados que forman y limitan el plano, es más, en ningún caso, en la pantalla de visualización, pueden mostrarse los límites del plano, ya que si fuera así se estaría literalmente ‘saliendo’ del mundo virtual propiamente dicho.

Esta es una de las razones por las que el plano necesita unas dimensiones considerables en cuanto a tamaño. Por otro lado, la textura que se le aplica al plano es una imagen de alta resolución, por tanto se hace necesario, que las dimensiones del plano sean adecuadas para aprovechar las oportunidades que la calidad de la imagen nos ofrece.

### Texturización del plano de vuelo

Como se ha descrito, el mundo virtual, principalmente consta de un único plano, para recrear el simulador de vuelo que se está buscando, pero aún no tenemos escenario virtual en sí. Es necesario crear una textura y aplicársela al plano para conseguirlo.

#### Confección de la imagen de textura

Confeccionar la imagen es el primer reto que ha sido necesario solventar para conseguir que el mundo virtual sea lo más atractivo y real posible.

La imagen debe representar la ciudad de Málaga y encontrar un mapa de alta resolución, que posibilite acercarnos a pocos metros de la superficie del terreno con una buena calidad de imagen, y que además, podamos alejarnos de la superficie visualizando una vista global de la ciudad.

Para salvar este obstáculo se ha recurrido a la conocida herramienta de Google, Google Maps. Esta aplicación, como es bien sabido, proporciona la capacidad de navegar en su vista ‘Satélite’ sobre el terreno orográfico de cualquier ciudad. Además para ciudades importantes o capitales de provincia, el detalle que consigue, y el zoom que se puede obtener sobre una parcela de terreno es muy aceptable y de buena calidad.

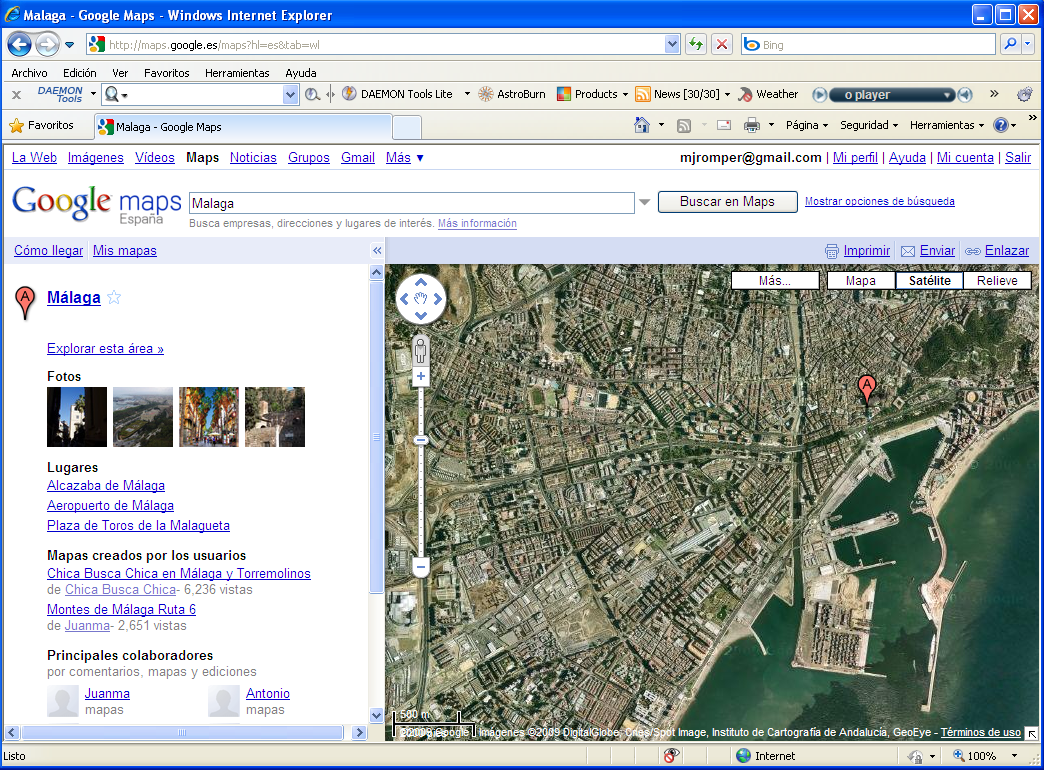
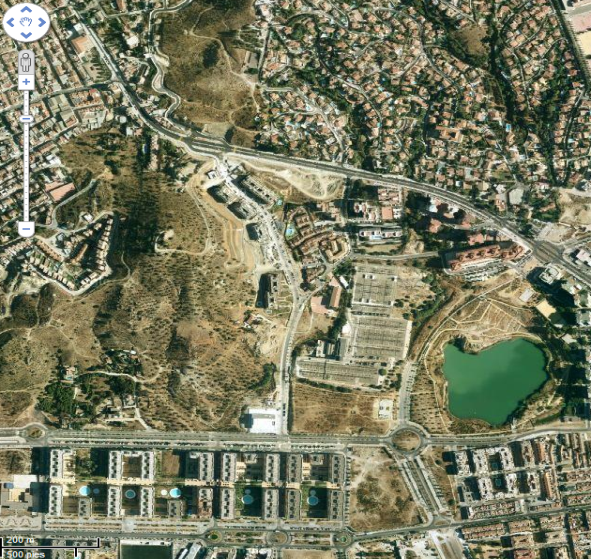


Figura : Google Maps

Ha sido necesario cuadricular la porción de terreno que se quería obtener como textura que representa la ciudad que se quiere sobrevolar (Málaga), con el objetivo de realizar capturas de pantalla de las zonas limitadas en la cuadricula como se muestra en la figura XXXX, con la resolución suficiente para que a la hora de realizar zoom (equivalente a un descenso del avión), la calidad de detalle de la porción de terreno que se visualiza sea de buena calidad.

En la siguiente imagen se puede observar de manera aproximada como se ha realizado el cuadriculado de la porción de terreno elegida como representativa de la ciudad de Málaga. Como se observa se ha divido en unos 60 cuadrantes. Cada uno de ellos aproximadamente tiene una resolución equivalente a sobrevolar la porción de terreno sobre unos 500 pies de altura como muestra la escala que Google Maps proporciona.

Figura : Cuadriculado de la imagen



De esta manera, por ejemplo, la porción de terreno correspondiente al cuadrante numerado B2 corresponde con la figura siguiente. En ella se aprecia el nivel de detalle y resolución que se pretende conseguir para garantizar un zoom sobre el terreno adecuado.

Figura : Cuadrante B2

Se unen todas las capturas, correspondientes a cada uno de los cuadrantes, componiendo la textura completa. En este punto se ha tenido especial cuidado para hacer casar los limites de cada cuadrante con los que lo circunda, evitando saltos abruptos y discontinuidades de la orografía del terreno.

Como se describirá más adelante, en el apartado ‘Texturización del plano de vuelo’, es conveniente que la textura final tenga un carácter cíclico, es decir, que empiece como termine por cada uno de sus lados. Esto es imposible de conseguir para todas las direcciones del plano, pero si se puede conseguir realizando un reflejo espejo de la textura y adosándoselo a la que ya teníamos. Por tanto con este proceso se obtiene cliclicidad al menos en la dirección transversal de la textura.

La imagen final que nos sirve de textura para el plano de vuelo y que hace las veces de terreno el cual se sobrevuela se muestra en la siguiente imagen. Se observa también como al hacer zoom sobre la textura creada se pueden conseguir visualizar como ejemplo, la ETS de Telecomunicación y el campo de futbol de la Rosaleda, con un detalle muy aceptable.



Figura : Textura final y detalle de Zoom

Como comentario final, decir que esta imagen, dada su alta resolución, se comprime en formato JPG pesando unos 4,5 MBytes. Este punto es muy importante ya que a mayor tamaño de la textura mayor velocidad de proceso y memoria es necesaria en la maquina que realiza la simulación del entorno virtual. El mundo virtual al visualizarse se carga en memoria RAM, por tanto cuanto mayor es el tamaño y número de texturas que se utilizan en él, mayor memoria y capacidad de proceso es requerida, por tanto es conveniente las imágenes que se utilizan como texturas tengan el menor peso en megabytes posible.

#### Texturización del plano de vuelo

El siguiente paso en el proceso de modelado de Simulador de vuelo es la texturización del plano de vuelo.

Se puede pensar que la mejor manera de texturizar el plano es simplemente colocar la textura sobre él, de manera que los vértices de la textura coincidan con los vértices del plano, ocupando de esta manera, toda la superficie.

Efectivamente es posible realizarlos así, no obstante para sacar mayor partido a la textura que se ha creado, se va a seguir un proceso de texturización un poco más complejo, aprovechando el carácter cíclico que se le ha proporcionado a la textura.

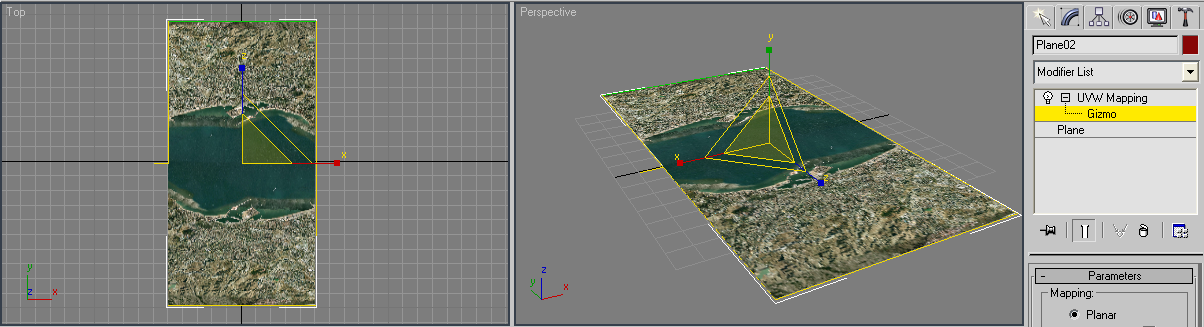
Se explica a continuación por qué se le ha dotado a la textura de un carácter cíclico, y cómo se ha utilizado.

En los mundos implementados en los capítulos anteriores, como se ha descrito, la sensación de movimiento y navegación a través del mundo virtual se realiza realizando desplazamientos sobre el “punto de vista” o “view point”. El “view point” es una cámara que hace las veces de ojo humano, que se desplaza y rota en el espacio tridimensional, proporcionado la sensación de irnos moviendo en el interior del mundo virtual.

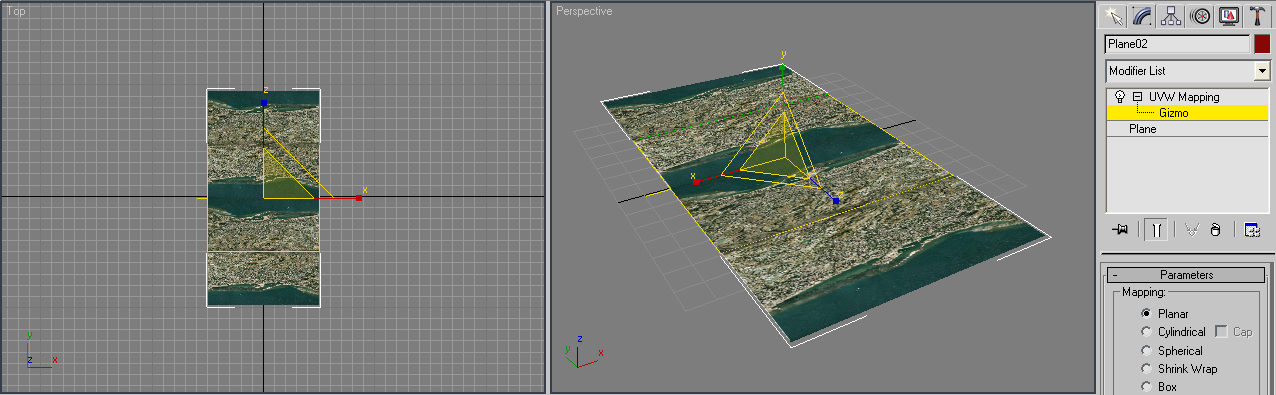
En cambio, el entorno que ocupa este capítulo, no se realizan desplazamientos ni giros sobre el punto de vista, sino que la sensación de movimiento y navegación se consigue desplazando la textura cíclicamente sobre el plano que texturiza, manteniendo el punto de vista fijo en toda la simulación. Este es el por qué de la conveniencia de una textura cíclica, al menos en una dirección.

En este sentido, lo que se hace es texturizar en primera instancia, la superficie del plano, pero aprovechando la ciclicidad de la textura. El proceso que se ha realizado, utilizando 3ds para realizarlo, se describe a continuación:

1. Se texturiza la superficie del plano de manera que la textura se aplique automáticamente sobre él.

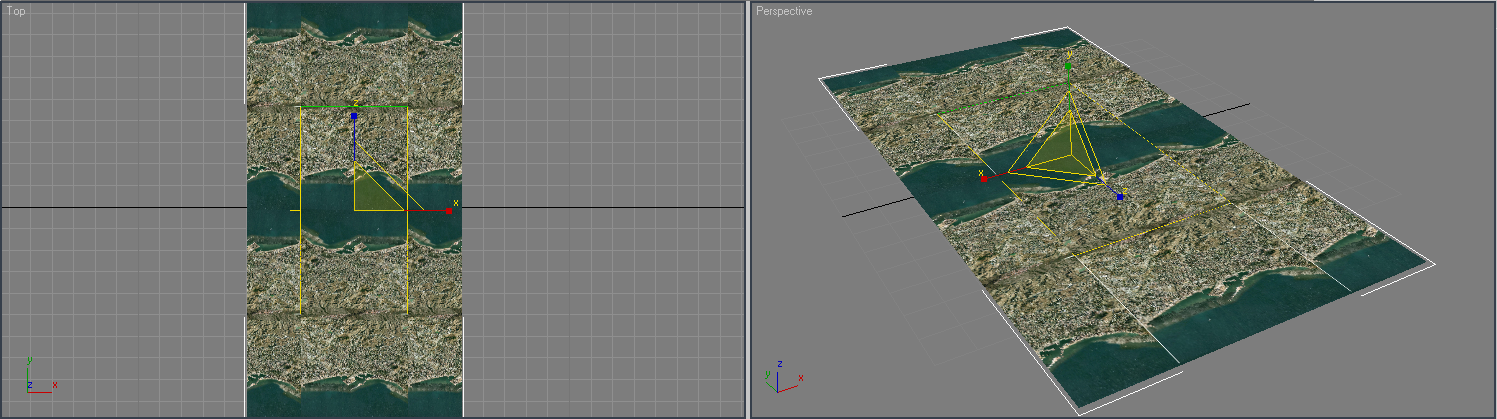


1. Se aplica el modificador “UVW Mapping” al plano, con el parámetro “Mapping” en “Planar”. De esta manera se asegura que la textura se aplica al plano, de manera que los vértices de la textura correspondan con los vértices del plano, ocupando la textura toda la superficie del plano.
2. Seguidamente con la herramienta de escalado reducimos el “Gizmo” de la textura. El Gizmo es la región cuadrangular (cuadrado amarillo) sobre la que la textura se está aplicando físicamente, o en términos 3D, donde se realiza el mapeado punto a punto de la textura. Fuera de la zona que limita el Gizmo, la textura se aplica cíclica y repetidamente. Reducimos el Gizmo en la dirección ‘z’, obteniendo una texturización como se muestra.



Como se observa se repite la textura en la dirección ‘z’, fuera de los límites del Gizmo.

1. Se hace lo mismo para la dirección ‘x’, reduciendo el Gizmo de esta dirección y observándose como la textura también se repita cíclicamente. En esta dirección se está aprovechando la ciclicidad que tiene la textura.



Se observa en esta última imagen como el Gizmo solo ocupa la parte central del plano que se ha texturizado, manteniéndose la forma original de la textura dentro de él. Además si se solapasen cuatro planos, como el que se muestra, cada uno de ellos adosado a cada lado del plano que se ha creado, se puede comprobar el carácter cíclico que tiene el plano modelado.

Los desplazamientos, que no escalados, de la textura sobre el plano aportan la sensación de movimiento y cuando la textura se pierde por uno de los lados, aparece por el lado opuesto cíclica y automáticamente. Esta facilidad a la hora de aplicar y desplazar texturas es la causante de que con un solo plano tengamos todo un escenario infinito por el que el espectador puede experimentar la sensación de volar.

Las mecanismos de desplazamiento de la textura sobre el plano se realizan mediante un procesado Matlab del mundo virtual exportado a lenguaje VRML. En epígrafes (XXX) posteriores volveremos sobre este punto, detallando de qué desplazamientos se trata y cómo se realizan para hacer experimentar al espectador la sensación de vuelo panorámico sobre la orografía malagueña.

### Modelo del avión 3D

En este mundo virtual se ha optado por incorporar un modelo 3D de un avión. De esta forma, la experiencia de sobrevolar la ciudad no se realiza en primera persona, sino que es a través de los movimientos de un tercero (el modelo 3D del avión), de sus movimientos, giros, ascensiones y descensos, como se represente la experiencia de este mundo virtual.

En definitiva este mundo virtual puede asemejarse a un pequeño juego, en el que el espectador controla un pequeño “avioncito” que vuela sobre la ciudad.

Existen multitud de diseñadores amateur que “cuelgan” sus modelos 3D en Internet, para su libre descarga, con el objetivo de divulgar sus modelos desarrollados. Utilizando esta alternativa y puesto que el objetivo de este proyecto no se centra en el modelado de formas complejas y altamente detalladas, se ha utilizado el modelo 3D de un aeroplano como el que se muestra en la siguiente figura.

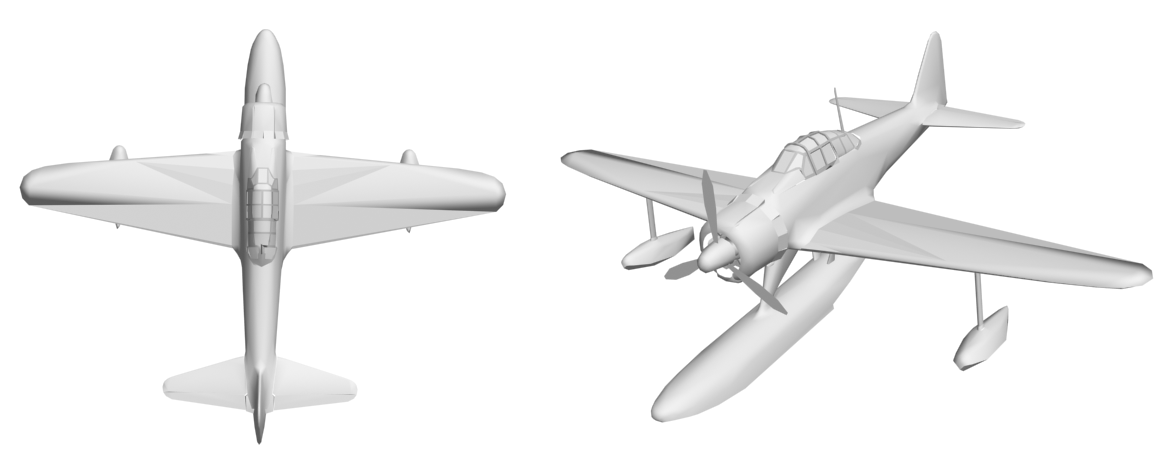


Figura : Modelo de avión 3D

No obstante el modelo ha sido modificado y optimizado, se reduce el número de polígonos y se texturiza:

* Reducción del número de polígonos, ya que al tratarse de un modelo original muy detallado, el número de polígonos utilizados en su implementación es muy elevado. Por tanto se le ha aplicado un modificador de optimización en 3ds (modificador Optimize), con el objetivo de, manteniendo su forma original, reducir el número de polígonos.

Esto es muy necesario al utilizar modelos complejos. La escena virtual es al final renderizada en tiempo real, utilizando OpenGL o DirectX como motor de renderizado y pintado de la escena. Cuanto mayor sea el número de polígonos mayor uso de CPU y tarjeta gráfica será necesario para realizar el renderizado de la escena en tiempo real.

Además hay que tener en cuenta como va a ser la visualización del modelo. Es muy probable que no necesitemos el modelo completo del aeroplano, debido a que únicamente se vaya a visualizar desde un punto de vista concreto, por tanto todos los polígonos que no sean necesarios, ya que nunca se va a observar, pueden ser perfectamente suprimidos del modelo.

* Re-texturización del modelo. El modelo original, aunque muy detallado en cuanto al modelado tridimensional, carecía de texturas. Por tanto se le han aplicado texturas que aunque son sencillas (solo se han utilizado tres texturas de colores simples) otorgan al modelo del aeroplano del un mayor realismo. Se puede observar en la siguiente figura.

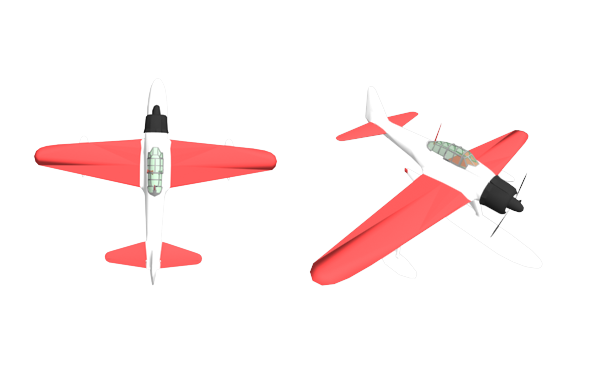


Figura : Modelo 3D avión texturizado

## Ubicación de los elementos

El siguiente paso en el proceso de creación del Simulador de Vuelo es la colocación estratégica en el espacio de cada uno de los elementos de los que consta.

Este epígrafe describe cómo han de ser situados en el espacio los elementos 3D modelados, colocándolo unos respecto de otros con el objetivo de caracterizar la escena de la manera más próxima a la realidad para el espectador:

* De cómo se sitúe el plano respecto del punto de vista (o view point) depende el grado profundidad que se desee obtener en la escena.
* De cómo se sitúe el avión respecto del plano depende la sensación de estar volando sobre la ciudad de Málaga, la velocidad de vuelo, la capacidad de giro, la altura del vuelo, etc.,…

### Ubicación del Plano

Como se verá en el epígrafe de integración con el interfaz de navegación del sistema BCI existente en el DTE de la ETSIT (XXX), el punto de vista o cámara que hace las veces de espectador, y desde el cual se observa el mundo virtual, viene impuesto por el propio interfaz de navegación con el que se integra el Simulador de Vuelo.

Este punto de vista tiene las siguientes coordenadas en el espacio: X = 0, Y = 0 y Z = 1.1. Es decir, se sitúa en el origen de coordenadas en el plano XY y se eleva 1.1 unidades en el eje Z.

Además el objetivo del punto de vista, es decir, la dirección de observación, se orienta en dirección Y, de manera que si a 100 unidades de distancia colocamos un ‘tetera’ se conseguiría observar la misma como se muestra en la parte superior derecha de la siguiente figura (recuadrada en amarillo).

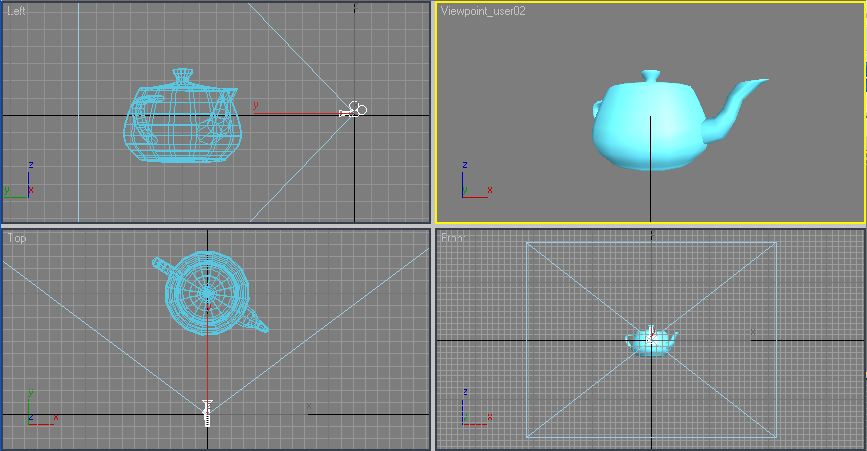


Figura : Punto de vista. Vistas Left, Viewpoint (amarilla) y Front.

El objetivo es colocar el plano justo delante del punto de vista, pero con ciertos matices.

Como sabemos el plano está texturizado con la imagen de la orografía de Málaga, y la textura representa la vista de planta de la ciudad, es decir, como se observara desde la perpendicular, de arriba hacia abajo, el terreno. Esta característica de la textura no es muy adecuada para proporcionar a la escena de un buen realismo, ya que la textura en sí, no tiene aspectos de profundidad o incluso relieve.

Para conseguirla, lo que se ha realizado es colocar el plano frente al punto de vista, pero no de forma perpendicular a la dirección de visionado (objetivo de la cámara), sino con algo de inclinación, para otorgar a la escena de cierta sensación de profundidad y relieve del terreno.

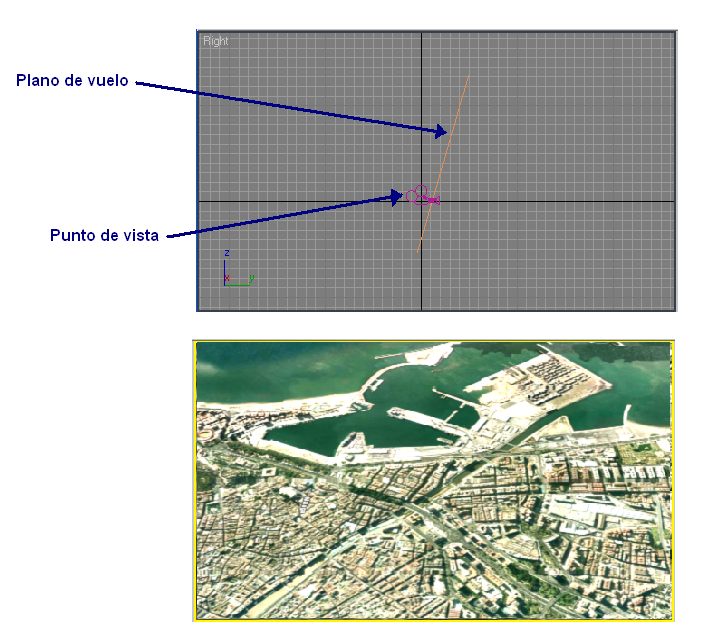


Figura : Ubicación del plano de vuelo y visualización desde el punto de vista

### Ubicación del Avión

El avión, como cabe esperar y para que se encuentre en la zona de visualización, se sitúa entre el punto de vista y el plano de vuelo, de manera que simule estar sobrevolando el terreno.

Como el plano está inclinado con respecto a la dirección de visualización y dado que el avión debe situarse aproximadamente paralelo al plano, el avión sufre una inclinación similar a la del plano.

En la siguiente imagen se muestra el resultado de incluir el avión en la escena.



Figura

Se puede observar que las dimensiones del avión son bastante reducidas. Esto es debido a dos razones:

1. El avión se sitúa muy cerca del objetivo del punto de vista, por tanto sus dimensiones no pueden sobrepasar del área de visionado, y tampoco debe ocupar toda la ventana de visionado y tapar el terreno que se sobrevuela, por tanto se ajustan sus dimensiones para que el avión quede cerca del punto de vista pero sin ocupar toda la pantalla.
2. Las dimensiones deben ser reducidas con respecto al plano de de vuelo. Para recrear la sensación de descender o, lo que es lo mismo, perder altura de vuelo, es necesario acercar el plano en dirección –Y, hacia el avión, por tanto, cuanto menores sean las dimensiones del avión mayor será el margen de acercamiento que puede recrearse del avión sobre el terreno, sin que el avión deje de visualizarse correctamente o incluso perderse tras el plano de vuelo.

En resumen, lo que se hemos realizado a lo largo de los epígrafes anteriores es el modelado 3D de los elementos que componen el entorno virtual, la texturización y ubicación estratégica de los mismos.

Los elementos que hasta ahora componen la escena son:

* El plano de vuelo, con la textura asignada.
* Un modelo 3D de un avión, obtenido ya modelado, que se ha optimizado y texturizado.

Notamos que aún no tenemos punto de vista en la escena. Éste se incluye al realizar la integración con el interfaz de navegación y que se detalla en el epígrafe XXXX, que de por sí incluye el propio punto de vista que necesita la escena para ser visualizada. El punto de vista que se ha utilizado mientras se realizaba el modelado ha sido incluido únicamente para el propio modelado.

Con este punto se concluye el modelado tridimensional **estático** de este mundo virtual, y todo se recoge en un fichero de 3DStudio, con extensión “.max”.

Los siguientes pasos serán la exportación a VRML para ser tratado en Matlab, y las labores de integración con el interfaz de navegación y desarrollo de algoritmos en Matlab para “animar” el mundo virtual y conseguir el movimiento necesario en la escena para recrear la sensación de estar realizando un recorrido aéreo sobre la ciudad.

## Exportación a VRML

Una vez que la escena ha sido creada y todo está emplazado en su lugar correspondiente, la escena está preparada para ser exportada a VRML. Desde la versión 5, 3dStudio crea ficheros VRML (con extensión “.wrl”) que son compatibles con el estándar VRML97.

Cuando se realiza la exportación de un entorno virtual desarrollado en 3dStudio al lenguaje propio de VRML, se tiene la oportunidad de configurar un número importante de parámetros en el cuadro de diálogo VRML 97 Exporter que se encuentra en el menú File/Export de este programa de diseño 3D y que se muestra a continuación:

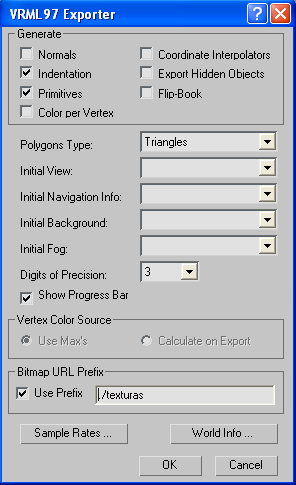


Figura : Cuadro de diálogo VRML 97 Exporter

Los parámetros que se pueden configurar a la hora de realizar la exportación son:

* *Normals*: crea normales reales, creando fichero VRML más grandes. Esta casilla se selecciona si la geometría usada en el modelado necesita de suavizado o “smoothing”, y los visores de VRML utilizados para visualizar el mundo requieren de estas normales para la correcta visualización.

En mundo virtual del Simulador de Vuelo no tiene geometría que necesiten de esta propiedad, además se comprueba que con los visores VRML utilizados, es decir, Cortona 3D Viewer y Orbisnap (visor VRML de Matlab), se obtienen resultados muy óptimos sin marcar esta opción.

* *Indentation*: realiza el sangrado apropiado del fichero VRML para una mejor lectura del mismo.

Los ficheros VRML resultado de la exportación son ficheros de texto plano. Si no se realiza un sangrado adecuado del código resultado, es prácticamente imposible distinguir los objetos y parámetros de cada objeto en el fichero WRL, por tanto se marca esta opción para obtener un código más legible.

* *Primitives*: exporta las primitivas de VRML en lugar de las primitivas de 3DStudio. Se consiguen ficheros más cortos.

Es importante que el fichero resultado de la exportación tenga el menor contenido posible. De esta manera al visor VRML le será más fácil procesar el código y menor uso de CPU será necesario para visualizar el mundo virtual en tiempo real.

* *Color per Vertex*: exporta el color de los vértices de la geometría.

Los vértices de las geometrías que componen el mundo virtual no son de mucha importancia que se exporten con un color concreto, ni con mucha precisión. Lo que sí interesa es que las caras de los polígonos, que limitan esos vértices, sí se exporten el color correctamente.

* *Coordinate Interpolators*: habilita exportar animaciones que utilizan interpolación de coordenadas, no solo las transformaciones simples de movimiento, rotación y escalado habituales. Se crean ficheros VRML más extensos.

Para los mundos virtuales creados donde la animación y movimiento de la escena en fases posteriores a la exportación no es necesaria.

* *Export Hidden Objects*: si se marca la exportación incluye objetos que se hayan escondido, y serán visibles en VRML. Normalmente se deja desmarcada.
* *Flip-Book*: exporta la escena en múltiples ficheros como una tasa de muestreo especificada. Se utiliza para escenas con animaciones (que no es el caso).
* *Polygon Types*: determina como son traducidas las caras de la geometría como nodos “IndexedFaceSet” de VRML. El tipo por defecto es “Triangles”, que traduce las caras como composiciones triangulares. “Ngons” traduce tantas caras como sea posible. “Visible edges” traduce, únicamente, las caras que son visibles.

A mayor complicación en la segmentación de las caras de la geometría, mayor es el tamaño del fichero resultado. El Simulador de Vuelo no tiene formas complejas y se ha selecciona el tipo “Triangles” tal y como viene por defecto.

* *Initial View*: lista todas las cámaras o puntos de vista de la escena y pone en primer plano la vista seleccionada inicial. Determina que vista es la primera en experimentar el usuario.

En mundo virtual creado aún no existe un punto de vista o cámara por tanto no tenemos ninguno que seleccionar.

* *Digits of Precision*: Numero de digitos decimales usados para calcular las dimensiones. Por defecto es 4. Un número menos da como resultado un fichero VRML de menor tamaño, pero un número mayor puede ser necesario si partes de la escena se han situado 100.000 unidades, o más, alejadas del centro de la escena.

Aunque las dimensiones del plano de vuelo son bastante elevadas, no superan las 200 unidades, en el caso peor, de separación con el centro de la escena, por tanto con el mismo fin de obtener un fichero VRML lo más compacto posible se utilizan 3 dígitos de precisión.

* *Bitmap URL prefix*: especifica la localización de las texturas asignadas a los objetos de la escena. Si no se selecciona, los ficheros de imagen deben situarse en la misma ubicación del fichero VRML.

En esta parámetro solo hay que tener cuidado de colocar la imagen de textura que recrea la orografía de la ciudad de Málaga en el directorio que se determine en esta opción, para que al visualizar el mundo en el visor de VRML sepa dónde buscarla.

El resto de parámetros no son de demasiado interés y no se detallan en este proyecto.

El resultado de este proceso es un fichero, con extensión “wrl”, de texto plano el que es fácilmente distinguible cada uno de los nodos VRML que forman parte del mundo virtual. La correspondencia de un elemento en 3DStudio y su equivalente en un nodo VRML es usualmente 1 a 1, aunque puede que existan agrupaciones de elementos en 3DStudio que al exportarlos a VRML se obtengan en nodos independientes, o viceversa.

## Integración y animación

Una de las partes que incluye este mundo virtual, al igual que se ha realizado con los mundos modelados anteriormente, es la integración con el sistema interfaz de navegación existente y desarrollado por el Departamento de Tecnología Electrónica.

La integración con el interfaz, en los mundo virtuales descritos en los capítulos anteriores, era casi inmediata ya que lo único que era necesario era incluir el nodo VRML “silla de ruedas”, que incluye el punto de vista del espectador. Los movimientos en el interior de la escena se delegan a este interfaz, que es capaz de proporcionar órdenes a la “silla”.

Es decir, en esos casos la experiencia de navegación a través de la escena virtual, bien la Vivienda Virtual o la ESTIT Virtual, se realizaba aportando movimiento al nodo “silla”, que obedecía las ordenes del sistema BCI realizando los desplazamientos y giros interpretados tras el procesamiento del sistema de captura de señales EEG.

Los elementos 3D que componen el mundo virtual, tanto en el caso de la vivienda como en el de la ETSIT, permanecen siempre inmóviles, siendo el punto de vista, adosado al nodo VRML “Silla” el que emula la sensación de movimiento a la escena.

En el caso del Simulador de Vuelo, la integración con el interfaz de navegación es algo más compleja, debido a que son los propios elementos de la escena, y a través de modificaciones sobre los mismos, los que hacen que la escena tome vida y movimiento.

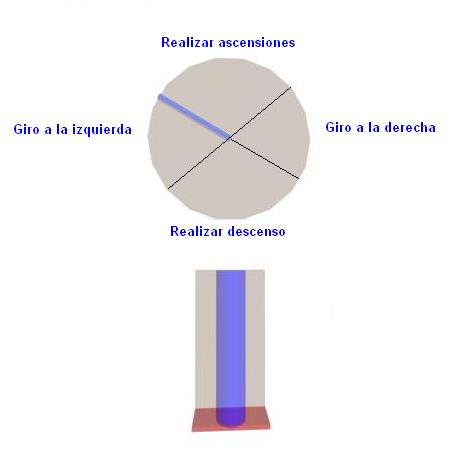
La integración con el sistema de navegación existente nacesita de dos etapas fundamentales:

1. Integración con el interfaz de navegación con el que se integra el mundo virtual, y que lleva adosado el punto de vista del espectador.
2. Implementación de algoritmos y métodos en código Matlab, para realizar los movimientos sobre los elementos 3D que componen el Simulador de Vuelo, es decir, sobre el modelo del avión y el plano de vuelo.

### Integración con el interfaz de navegación

El interfaz de navegación existente y con el que se integra el mundo virtual del Simulador de Vuelo consta de tres elementos que se agrupan bajo un mismo elemento o nodo VRML, los tres elementos son:

* Aéreas de selección de movimiento de navegación: se pueden realizar la toma de cada una las siguientes opciones de movimiento:
  + Ascensiones, el avión toma altura en su vuelo sobre la ciudad.
  + Giros a la derecha del avión.
  + Descensos. O lo que es lo mismo, perdida de altura de vuelo.
  + Giros a la izquierda del avión.
* Barra indicadora del nivel de concentración del sujeto



Figura

En el proceso de selección de una opción una barra va recorriendo, en sentido horario, las cuatro aéreas que componen el círculo de selección de opción, mientras la barra de concentración indica el grado de concentración en la toma de cada una de las opciones. En el momento en el que el individuo alcanza un umbral de concentración determinado, el entorno virtual la interpreta y realiza las modificaciones necesarias en sus elementos 3D para simular el movimiento que determina la opción elegida.

Una vez que disponemos del mundo virtual del Simulador de Vuelo bajo el estándar VRML, para integrar el interfaz de navegación es necesario incluir en el código fuente VRML del Simulador de Vuelo el nodo VRML que modela el interfaz de navegación.

Es tan fácil como abrir el fichero VRML del Simulador de Vuelo en un edito de texto e incluirle el nodo con el interfaz de navegación.

Si se ejectua el Simulador de Vuelo con el visor de entornos VRML Cortona3D Viewer se puede observar el resultado tras la integración con el interfaz de navegación desarrollado en el DTE.



Figura

### Implementación del movimiento

El interfaz con el que se ha integrado el Simulador de Vuelo, al interpretar una de las elecciones de movimiento, como se ha explicado anteriormente, lo que realiza son desplazamientos del punto de vista.

En el caso de este mundo virtual al interpretar un opción de navegación, el movimiento de la escena no lo simula los desplazamientos sobre el punto de vista, sino que son los propios elementos de la escena los que se mueven. De esta manera se describe a continuación como se ha implementado cada una de las cuatro opciones de navegación que soporta el interfaz con el que se ha realizado la integración.

Antes de nada hay que tener en cuenta que ¡¡el avión nunca puede estar parado en el aire!! ya que en caso contrario los pasajeros sufriría un grave accidente.

Esta circunstancia hace que aunque no se haya tomado alguna de las opciones de navegación, el avión sigue avanzando, sobrevolando el terreno en la dirección actual y en línea rectal. Lo primero que se debe explicar es cómo se recrea esta situación con los elementos que disponemos.

Como se explico en el epígrafe XXX, recrear la sensación del avión sobrevolando el terreno, no se realiza realizando un desplazamiento de avance del modelo 3D del avión sobre el terreno, ni desplazamientos del punto de vista del observador, sino que es el propio terreno (textura del terreno) el que se desplaza, permaneciendo en todo momento el modelos del avión y el punto de vista inmóvil en sus coordenadas originales.

1. **Avance continuo del avión**

Se consigue realizando el desplazamiento de la textura, más bien del Gizmo que la representa, en dirección –Y del eje de coordenadas. Esta dirección en realidad es contraria al avance natural a la que tendría que realizar el avión y el punto de vista, pero la sensación es la misma, es decir, que el avión parece que sea el que está avanzando, pero en realidad lo que avanza es el terreno, pero en sentido contrario.

FOTO CON UNA SECUENCIA DEL MOVIMIENTO DE LA TEXTURA Y LA SENSACIÓN QUE SE CONSIGUE

Los cambios necesarios a nivel de implementación de código Matlab han requerido que en las líneas de código en los que se realizaban movimientos del punto de vista, en el interfaz existente, ahora lo que se haga es el desplazamiento de la textura en la dirección que se ha detallado. Se ha implementado un método Matlab, llamado “goAhead”, para llevarlo a cabo y este método se ejecuta en todo momento siempre y cuando no se haya tomado alguna de las opciones de navegación. El código necesario para esta implementación se muestra en el APENDICE XXX.

1. **Giro a la derecha o izquierda**

Estas opciones de navegación se recrean realizando movimientos diagonales de la textura en la dirección contraria al movimiento natural que se pretende conseguir. Además en estos casos, el propio modelo del avión también se modifica, dotándole de un movimiento de rotación a su eje longitudinal en la dirección del giro que se haya seleccionado. Es decir, el avión sigue literalmente inmóvil en la escena, permaneciendo en las coordenadas originales, pero se rota su eje hacia la derecha o izquierda según se haya elegido un giro u otro, dotando a la escena de mayor realismo a la hora de recrear cada uno de los movimientos. En realidad la textura avanza y el avión rota y la sensación se aproxima a un giro y avance natural tal y como lo haría un avión real.

La siguiente secuencia de imágenes plasma el resultado

1. **Ascensiones y descensos**

En este caso la textura debe de seguir avanzando tal y como se describe en el punto 1, pero además entra en juego la posición del plano de vuelo al que se le aplica la textura. Si se desplaza paulatinamente el plano en dirección –Y, lo que se está haciendo es alejar el plano del avión, por tanto se experimenta la sensación de ascender el vuelo y alejarnos del terreno.

Si, por el contrario, se desplaza el plano en dirección Y, se está acercando a la posición donde permanentemente se localiza el avión, y se consigue así la sensación de estar descendiendo.

Si además, se acompañan estos desplazamientos del plano texturizado con la rotación del eje transversal del avión simulando que, o bien, levanta unos grados el morro en los ascensos y lo agacha en los descensos, se consigue así la experiencia completa de estar realmente cada uno de los movimientos.

Se puede observar el efecto conseguido en la siguiente secuencia de imágenes.