

ADECUACIÓN DE MODELOS VIRTUALES TRIDIMENSIONALES PARA EL DESARROLLO DE UN SOFTWARE SIMULADOR DE CONDUCCIÓN DE AUTOMÓVILES

Gustavo Rubén Di Rado – Gustavo Devincenzi

Departamento de Vías de Comunicación Facultad de Ingeniería UNNE
gdirado@ing.unne.edu.ar

Resumen

Esta publicación está orientada al trabajo efectuado sobre modelos tridimensionales realizados en un entorno de Diseño Asistido por Computadora, con el fin de poder ser utilizados en un espacio de programación multimedial, teniendo en cuenta cuestiones referidas por ejemplo a: modelado del entorno, texturado, importación de modelos de vehículos, separación de diferentes elementos del modelo que serán programados de forma individual, realización de vinculaciones entre ellos, visualización del modelo, colocación de cámaras y luces. El objetivo es el desarrollo de un simulador de conducción de automóviles, que será aplicado a la enseñanza de asignaturas tales como Transportes y Vialidad Especial del Dpto. de Vías de Comunicación de la Facultad de Ingeniería de la UNNE.

Palabras claves: Simulación – Vehículo - Modelos

Introducción:

Los simuladores de conducción de vehículos han estado siendo usados desde hace varios años en diferentes ámbitos de la actividad humana. El primero de ellos fue desarrollado en 1995, y desde esa fecha han ido progresando sus resultados y usos con el incremento constante de los computadores aptos para trabajar con gráficos tridimensionales. Dentro de sus aplicaciones podemos encontrar su uso en entrenamiento y aprendizaje de conducción, estudio y análisis de accidentes de tránsito, y por supuesto dentro del ámbito educativo en cátedras de proyectos de construcción de recintos urbanos, rutas, intercambiadores de autopistas, diseño de recintos de estacionamientos, utilización de semáforos, etc. En este primer trabajo se explican los pasos preliminares para llegar al desarrollo de un simulador.

Entorno de Circulación

Modelado:

El modelado del entorno es una de las partes más importantes en lo que respecta al diseño de un simulador de conducción de vehículos, ya que su utilidad y objetivo de aplicación va a depender de ello.

Verdaderamente tenemos posibilidades casi ilimitadas, partiendo de la creación de módulos de ciudades virtuales o recintos de circulación de vehículos hasta el modelado de caminos rurales, autopistas o intercambiadores de tránsito.

El problema principal inherente a simulaciones de entornos urbanos o rurales es la complejidad de la construcción, que requiere gran performance del equipo informático para poder efectuar los renders, es decir la visualización de la realidad virtual y en especial en estos casos en que este

proceso se realiza en tiempo real, esto es, la captura de cuadros va efectivizándose a medida que se producen los movimientos dentro del mundo virtual.

Uno de los secretos mas comunes es repetir las representaciones de objetos dentro del escenario, por ejemplo edificios, paradas de ómnibus, casas, etc., lo cual no se nota al usuario por la propia extensión de la construcción.



Fig. 1 Representación de entornos urbanos.

En la figura vemos un ejemplo de lo explicado.

La repetición de los objetos debe ser realizada de una manera especial a efectos que no se produzca la sobrecarga de memoria del modelo y por lo tanto un tamaño excesivo en el futuro archivo “World 3d”, y en el ejecutable el cuál contendrá el simulador, por los mismos problemas de performance anteriormente explicados.

La creación de objetos de idénticas características puede ser realizada de tres maneras:

Copiado: Se crean dos objetos distintos e independientes con idénticas características, es decir, ambos pueden ser modificados por separado sin que el otro se vea afectado. Esto produce la máxima carga de memoria para esta acción.

Calco: Se crean dos objetos distintos y dependientes uno del otro, lo que origina que al modificar uno de ellos, el otro se modifica automáticamente.

Referencia: Creamos un segundo objeto dependiente del primero pero no a la inversa, de esta manera modificando el primero, lo hacemos también con el segundo pero no a la inversa. Esta opción nos da la menor carga de memoria al modelo.

Lo anteriormente explicado ha sido comprobado en la construcción del modelo, realizando múltiples pruebas en los tres sentidos.

Un punto a considerar para lograr acelerar la representación en tiempo real (render), a medida que se utiliza el software, es el de regular lo que se denomina modificador “LOD”, que proviene del inglés “Level of Details”. Esto se realiza dentro del entorno de programación, asignándole este atributo a cada modelo, para que a medida que la cámara que toma la escena se aleja, la resolución de los objetos sea menor, con la consiguiente mejora en el tiempo de respuesta.

Por ejemplo una línea de código como la siguiente puede ser aplicada en lenguaje LINGO:

```
member(whichcastmember).model(whichmodel).lod.bias = 50
```

Lo que implicaría que la resolución de ese modelo baja sensiblemente a la vez que se aleja de la posición de la cámara.

Para la realización del presente trabajo, se comenzó el modelado del camino a partir de la realización del perfil en dos dimensiones utilizando entidades denominadas polilíneas, tomando el ejemplo de un pliego de Vialidad Nacional utilizado en la cátedra “Vias de Comunicación I”.

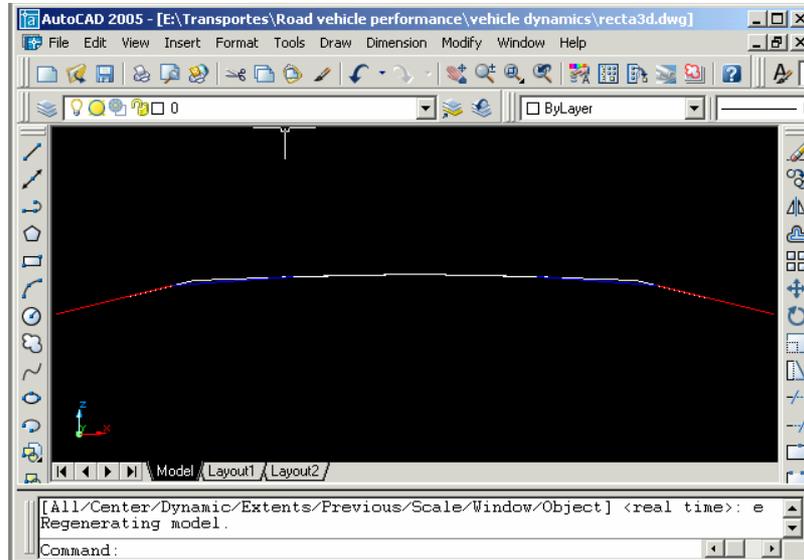


Fig. 2 Diseño del perfil del camino en 2 dimensiones

Luego se procedió a transformar esas entidades en mallas utilizando “Superficies interpolada entre cuatro lados”, definidas por cuatro curvas cualesquiera en el espacio que deben tocarse en sus extremos, con el fin de obtener una superficie cerrada topológicamente poligonal.

La densidad de la malla es regulada por variables y se considera una dirección M, dada por el primer lado que se toca, y dirección N, que queda automáticamente determinada en la dirección perpendicular.

Las variables surftab1 y surftab2 regulan la cantidad de veces que se dividen los lados correspondientes.

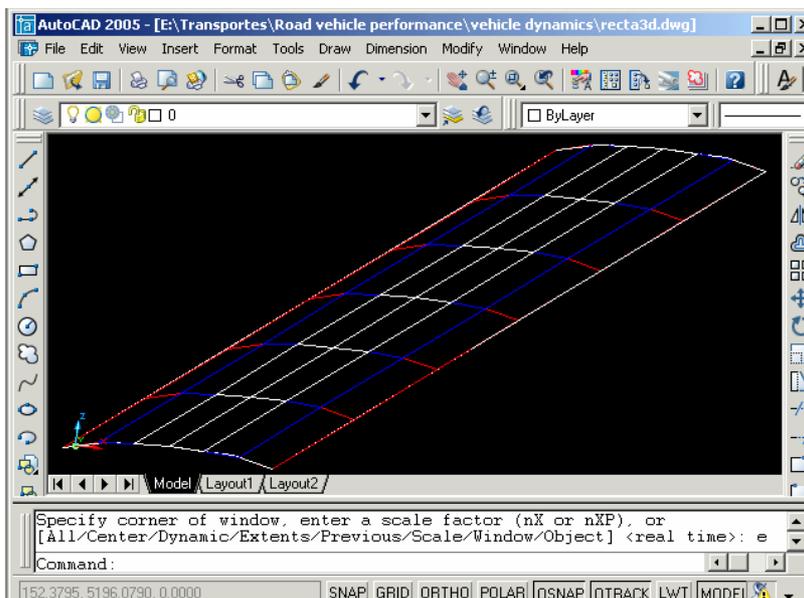


Fig. 3 Diseño del perfil del camino en 3 dimensiones

En la generación de la malla se procedió a colocar diferentes partes de la misma en diferentes capas, a saber: “ruta”, en la parte central de color negro, “banquina1”, en la siguiente de color azul, y “banquina2” en la siguiente de color rojo, de manera que al ser exportadas al formato .3ds, sean tomados como objetos independientes a los cuales se les puede asignar de esta forma texturas distintas.

En este nuevo formato, se terminará de realizar el modelado del entorno, para dar mejor terminación a la escena final.



Fig 4 Modelo del camino terminado.

Texturado:

El texturado se realiza como anteriormente se explicó para dar terminación al entorno, y para que el usuario tenga una visión tridimensional de resolución fotográfica a través de las cámaras que se colocan en la escena, con el fin de que pueda distinguir distintos objetos así como también diferentes materiales tal y como lo haría en una situación real.

Las texturas se crean a partir de archivos raster o de bitmaps, que son aquellos que generan un dibujo mediante el encendido y apagado de los pixeles, los cuales son muy utilizados para la representación de fotos, o de imágenes virtuales con resolución fotográfica.

Estos tipos de dibujos pueden ser obtenidos cuando se utiliza un escáner para digitalizar una imagen o mediante una cámara digital. Dentro de ellos, encontramos diferentes formatos que sirven a diferentes características de visualización y compresión. Algunos de los formatos que podemos nombrar son los siguientes:

JPG: Es un formato comprimido de 24 bits, por ejemplo RGB, posee alta compresión para imágenes fotográficas o similares y una variante con y sin pérdida de información. Tiene deterioro de nitidez en bordes o líneas.

GIF: Es un formato comprimido de 8 bits, se utiliza mayoritariamente en imágenes publicadas en la WEB, tiene poca compresión pero cuenta con imágenes muy nítidas, por su baja información de bits, no se usa para gran cantidad de colores.

TIFF: Almacena imágenes RGB sin comprimir o con compresión sin pérdida.

TGA: Soporta 16, 24 y 32 bits de color RGB. Soporta compresión sin pérdida de información.

BMP: Es el formato de Windows. Hay de 1bit (blanco y negro), 4 bits (16 colores), 8 bits (256 colores), 24 bits (aproximadamente 16 millones de colores). Es un formato que soporta compresión sin pérdida de información.

Algunos de los archivos utilizados en este trabajo son los siguientes:



Fig. 5 Texturas utilizadas en el trabajo.

El procedimiento consiste en asignar estas “fotografías” a diferentes partes del modelo, las cuales se amoldan al formato de las mallas que lo componen según parámetros de mapeado que pueden ser plano, cilíndrico, esférico, cúbico, etc., y mediante un procedimiento de repetición según ejes U, V, o W, paralelos a los ejes coordenados, se cubren completamente los objetos. También la alineación de los mapas se realiza según los ejes coordenados.

Colocación de Cámaras y Luces:

La colocación de una buena cantidad de cámaras ubicadas estratégicamente y una buena iluminación de la escena, contribuyen a dar mayor realismo y a controlar los distintos espacios del entorno, así como visualizar situaciones del movimiento de los vehículos que no podrían ser posibles en la vida real, como por ejemplo, múltiples vistas de la representación de un accidente de tránsito.

Tenemos la posibilidad de indicar diferentes tipos de luces, especificar su intensidad y color. Dentro de los tipos de iluminación podemos citar:

Luz Puntual: se trata de una fuente de luz que irradia en todas direcciones, como sería el caso de una bombilla de luz.

Luz Distante: es una fuente de luz que emite en una sola dirección, que podría imitar el caso de la luz solar. Los objetos alejados se ven tan iluminados como los cercanos.

Foco: emite luz en forma de cono, en una dirección especificada.

Luz Ambiental: determina la luz ambiente o de fondo, proporcionando una iluminación constante por igual en todas las superficies del modelo.

En el presente trabajo se han colocado tres luces puntuales, y una luz distante que deberá ser ajustada para que represente la luz solar. Podrá luego, a través del código LINGO, hacer que dicha luz se apague, prenda o se mueva APRA simular el amanecer, mediodía, atardecer, o noche cerrada o con luna, etc.

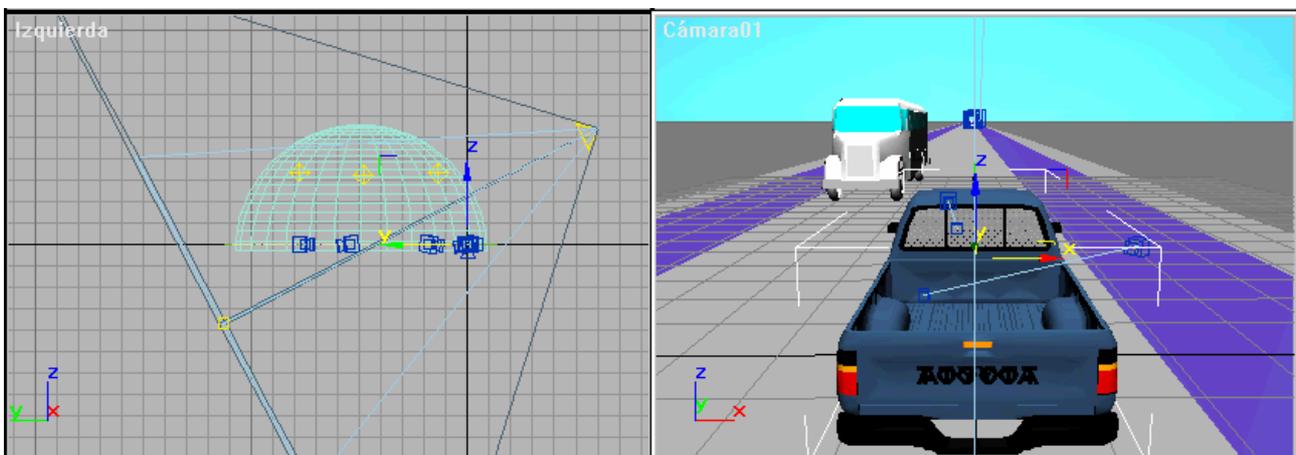


Fig. 6 Iluminación del escenario de circulación.

El número de cámaras que se pueden colocar en la escena es tanto como se quiera, por lo que como se mencionó anteriormente es conveniente aprovechar esta ventaja.

En este caso hemos colocado hasta el momento siete cámaras, que cubren espacios alrededor del vehículo a conducir por el usuario, situando la visualización desde atrás, como se ve en la fig. 6, desde adentro y desde un costado hacia abajo, para apreciar el movimiento de la ruedas (fig. 7), además de en algunos lugares claves del recorrido del camino.

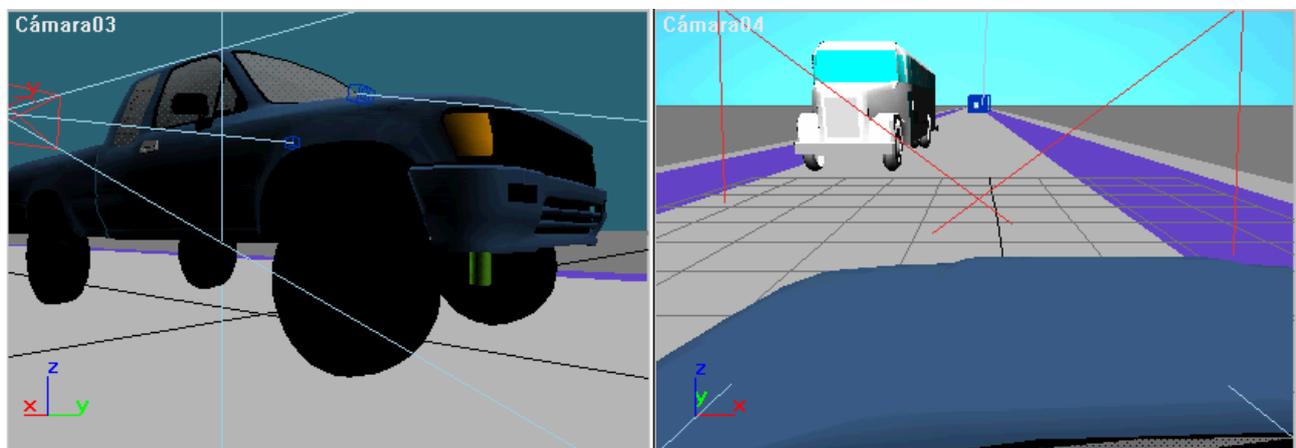


Fig. 7 Posiciones de la cámara.

Mediante un procedimiento que se denomina “vinculación”, se puede lograr que la cámara siga el movimiento de un determinado objeto, estableciendo una relación de hijo y padre entre ellos, de esta manera se han relacionado las cámaras mostradas en las figuras, con el vehículo conducido, así en cualquier parte del recorrido, luego de aplicar al modelo el correspondiente código en LINGO, el usuario puede, presionando un botón, cambiarse a cualquiera de vistas, a medida que avanza por el camino que haya elegido.

Otros parámetros pueden ser cambiados en cuanto a cámaras se refiere como por ejemplo el objetivo en mm y el FOV en grados, que permite mayor o menor apertura de la lente, y por lo tanto una visión más o menos amplia de la escena, planos de recorte, en donde se determina a partir de que distancia se toma el objetivo y hasta donde llegará, así como la distancia al blanco.

A través de la aplicación del código y de la manipulación de objetos se puede lograr también el movimiento de las cámaras colocadas en escena, esto es en traslación y rotación, así, el conductor del vehículo virtual podrá rotar su visión como si estuviera girando su cabeza dentro del escenario 3d para obtener un percibimiento mejor los objetos.

Importación de Modelos de Vehículos

Formatos

Los formatos de importación de archivos tridimensionales, aplicados en este caso a vehículos convencionales de uso particular o público, pueden venir expresados de distintas maneras y deben en todos los casos ser tratados particularmente.

Los tres formatos más comunes son *.dwg, *.3ds y *.max.

*.dwg: Este formato se ha transformado en un estándar para intercambio de archivos de dibujo vectoriales, puede ser abierto por gran cantidad de software de CAD, pero es especialmente preparado por Autodesk para su utilización con Autocad. Es el que tiene menor carga de memoria, pero en perjuicio no cuenta por lo general con el texturado del mallado.

*.3ds: Este formato ha sido creado por Autodesk en principio como archivo de uso para trabajar sobre el software 3ds Studio, pero actualmente es más usado para intercambio de modelos tridimensionales, en este caso se ha usado para pasar archivos desde la plataforma Autocad a 3d Studio., y viceversa. Es un formato cómodo porque trae consigo el texturado de los objetos. En algunos casos, el render tiene algunos problemas a solucionar editando el modelo.

*.max: Es el formato de archivo de 3d Studio max, que finalmente será el definitivo, es decir, al que se pasarán los formatos .dwg y .3ds, ya que desde allí se trabajan los modelos como anteriormente se explicó, y se realiza la exportación al formato .w3d, que será el entorno en el cuál se realizará la codificación. Es ideal por no tener que realizarse conversión, y por tener menor tratamiento posterior.

Visualización:

La visualización de los modelos tiene que ver con el tratamiento que se le debe realizar a las mallas que lo componen para que cuando éstos sean pasados al formato .w3d, no se produzcan “huecos” en su estructura haciendo que se vean transparentes en algunos sectores provocando obvias dificultades para representar la realidad virtual.

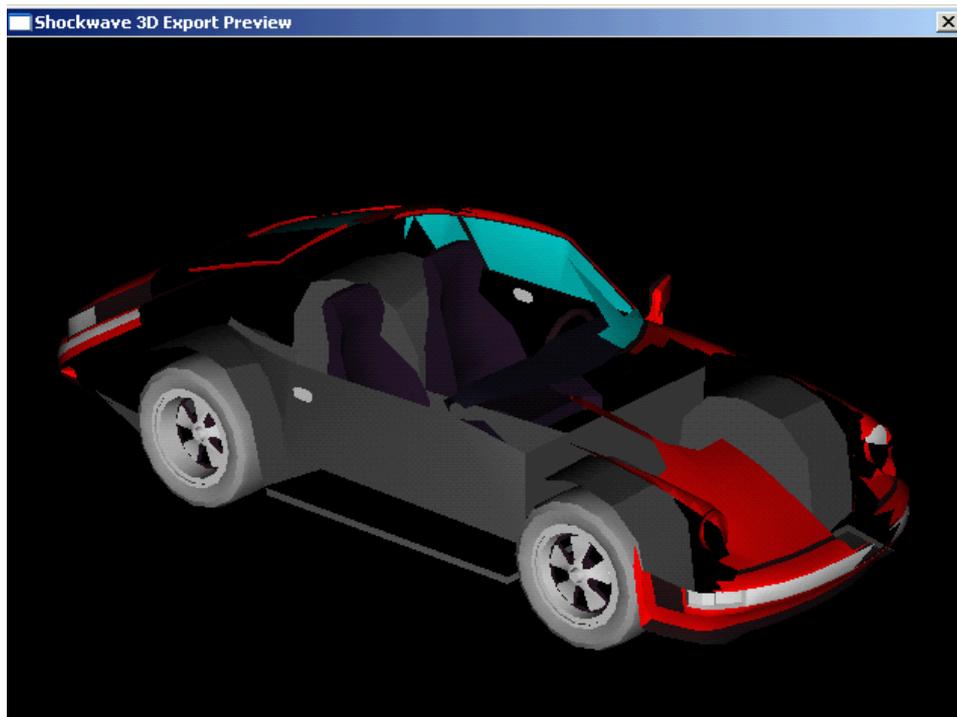


Fig. 8 Visualización Incorrecta de un Modelo de Automóvil

Para corregir este problema, tenemos que proceder a la edición del modelo como “malla editable”, y de allí a los sub-objetos del mismo, ya sea como elementos, polígonos, caras (triangulares), y trabajar con lo que se denominan Normales.

Las normales son vectores perpendiculares a la malla los cuales pueden estar direccionados hacia dentro o hacia fuera de la misma, lo que variará la vista sombreada.

El “secreto” de este problema es tener los vectores dirigidos hacia la dirección de la cámara que toma la escena, es decir, si los vectores apuntan hacia adentro y la cámara se encuentra fuera del objeto, se verán huecos como se aprecia en la fig. 8, si en ese mismo caso la cámara estuviera dentro del coche, la vista sería opaca y se vería el techo desde dentro.

Se procede entonces a editar el modelo a través de sus sub-elementos, y a acceder a las propiedades de superficie para visualizar las normales.

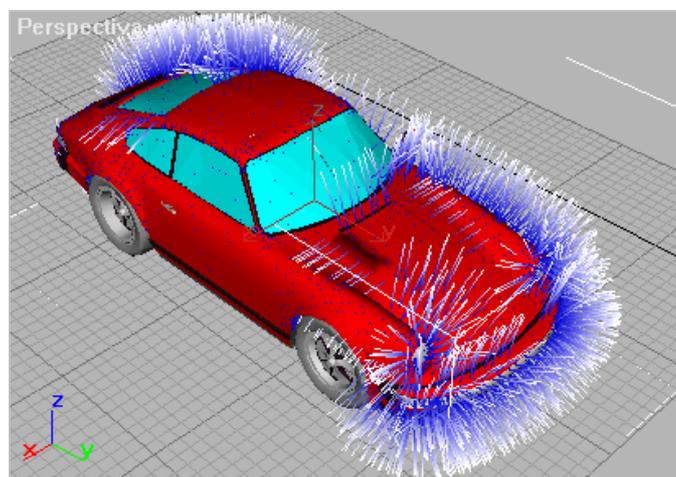


Fig. 9 Visualización del Modelo y sus Normales

Como se observa en la fig. 9, y en correspondencia con la fig. 8, las normales que se direccionan hacia fuera son las que corresponden a la visualización correcta desde la posición de la cámara del automóvil. En los lugares en donde no se aprecian los vectores azules y blancos, las normales apuntan hacia adentro y hay que “voltearlas”, trabajando en este caso particular a nivel caras. Se aclara que en la fig. 9 el modelo está editado a nivel “elemento”.

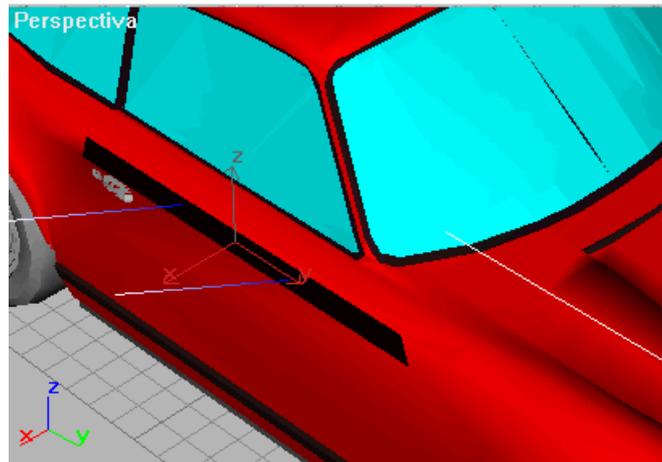


Fig. 10 Edición y Cambio de Orientación de las Normales.

En la fig. 10 se ha procedido a seleccionar una de las caras de la puerta del automóvil, y a reorientar sus normales en dirección externa hacia el lado en que apunta la cámara activa.

Si se pretende mostrar el modelo desde adentro y desde afuera, se pueden construir mallas dobles, y orientar las normales hacia adentro en la malla interior y hacia fuera en la malla exterior.

La visualización puede también ser manipulada desde el código en LINGO, utilizando un comando que se denomina precisamente “visibility”, y se aplica a cada modelo en particular componente del mundo virtual. Este determinará como será representada la geometría del modelo, teniendo las siguientes posibilidades:

#none: El objeto se ve invisible

#front: Solo aquellos polígonos que tienen sus normales dirigidas hacia el lado de la cámara son representados. Este método optimiza la velocidad del render (recordar que esto es muy importante, ya que en este caso la computadora trabaja en tiempo real).

#back: Son representados los polígonos con sus normales dirigidas hacia el lado contrario de la cámara. Se suele utilizar cuando se quiere representar el interior del modelo, o cuando el modelo no está bien construido, posiblemente por haber sido importado desde un archivo con diferente manejo de normales.

#both: Especifica que los dos lados de los polígonos son representados. Se utiliza para ver ambos lados de la superficie, o cuando el modelo está mal construido. Hace más lento el render, y es por esto que conviene tratar el modelo desde antes de su exportación a .w3d.

Conclusión:

El desarrollo de herramientas informáticas como auxilio en la tarea educativa es algo que con el avance de la tecnología se hace cada vez más necesario. En este caso, la creación de un simulador de conducción de automóviles para corroborar cálculos realizados sobre la base de ecuaciones matemáticas en el ámbito de la carrera de Ingeniería Civil, propone que la realidad sea representada lo más fielmente posible tanto desde el punto de vista de la física (será abordado en futuras

publicaciones), como el de la visualización gráfica, y es por esto que debemos estudiar la manera de cumplir con los parámetros expuestos en este trabajo.

Bibliografía:

- [1] Edson Tanaka. Macromedia Director. Axcel Books 1996.
- [2] Macromedia Director. Using Director. Macromedia 1997.
- [3] Macromedia Director. Learning Lingo. Macromedia 1997.
- [4] Macromedia Director. Lingo Dictionary. Macromedia 1997.
- [5] Phil Gross. Director 7 y Lingo. Anaya Multimedia 1999.
- [6] Autocad 2004/2005 Avanzado. José Antonio Tajadura Zapirain. Javier Lopez Fernández. Mc Graw Hill. 2004
- [7] 3d Studio Max 3 Práctico. Guía de Aprendizaje. Daniel Burgos. Mc Graw Hill.
- [8] Lingo Dictionary. 3d Lingo Dictionary. Ayudas del Software Director MX 2004.