

textOsFtextTOsFliningLFliningTLFtextosflininglftabulartabproportionalprosuperiorSup
superiorSup
fontspechyperref

TITLE

Author1

University

APLICACIONES DE REALIDAD AUMENTADA CON MODELOS BLENDERN

Nancy D. Hidalgo Vázquez 1, Mario E. Leal-Miranda 2
waispers@hotmail.com, locus1232000@hotmail.com;
CIDETEC – IPN Centro de Innovación y Desarrollo Tecnológico en Cómputo 1,2
Maestría en Tecnología de Cómputo
Unidad Profesional Adolfo López Mateos, México, D. F., C.P. 07700.

Introducción

Hoy en día es más fácil encontrar aplicaciones que hacen uso de la nueva tecnología de realidad aumentada, la cual consiste en sobre poner elementos virtuales en un ambiente real, y son poco a poco más las compañías que adoptan esta técnica como una herramienta.

Los dispositivos móviles son los que han adoptado mejor esta tecnología debido a su rápida evolución, contar con una cámara, GPS, y una conexión de datos, lo cual los hacen ideales para llevar esta tecnología a cualquier parte.

Ahora estos dispositivos cuentan con sistemas operativos que a la hora de desarrollar son los que deciden todo pues la compatibilidad entre ellos no existe y es por eso que se debe decidir bien.

Un sistema operativo libre, multiplataforma que permita desarrollar aplicaciones sin problemas interactuando con los componentes del dispositivo móvil es una buena opción además el que sea gratis mejora la decisión, Android permite programar aplicaciones en una variación de java llamada Dalvik, proporcionando todas las interfaces necesarias para desarrollar aplicaciones.

Una de las ventajas que se ven al trabajar con Android es la facilidad de encontrar aplicaciones, el poder crear aplicaciones propias e involucrar tecnología que cada vez es mejor y fácil de usar. Si bien esto requiere de ciertas habilidades de programación, muchas de las funcionalidades ya nos las dan programadas, y solo hay que hacer llamadas a sus funciones.

En esta ocasión se genero el modelo de realidad virtual en blender; el crear los modelos de esta manera nos permite una independencia de cada uno de los objetos que se crean o quieran incorporar, para darle cuidado a cada uno de los detalles y crear modelos de mayor complejidad.

En la escena donde se creó el modelo fueron insertados dos objetos a base de primitivas; cubos en este caso, para el modelo de una pintura artística y su marco; dándole a cada cubo el tamaño necesario para exponer las texturas deseadas.

Para tener una mejor vista se utilizaron imágenes jpg e imágenes normalizadas para las texturas, que al unir las crean un efecto de relieve y profundidad los objetos.

[]@|@

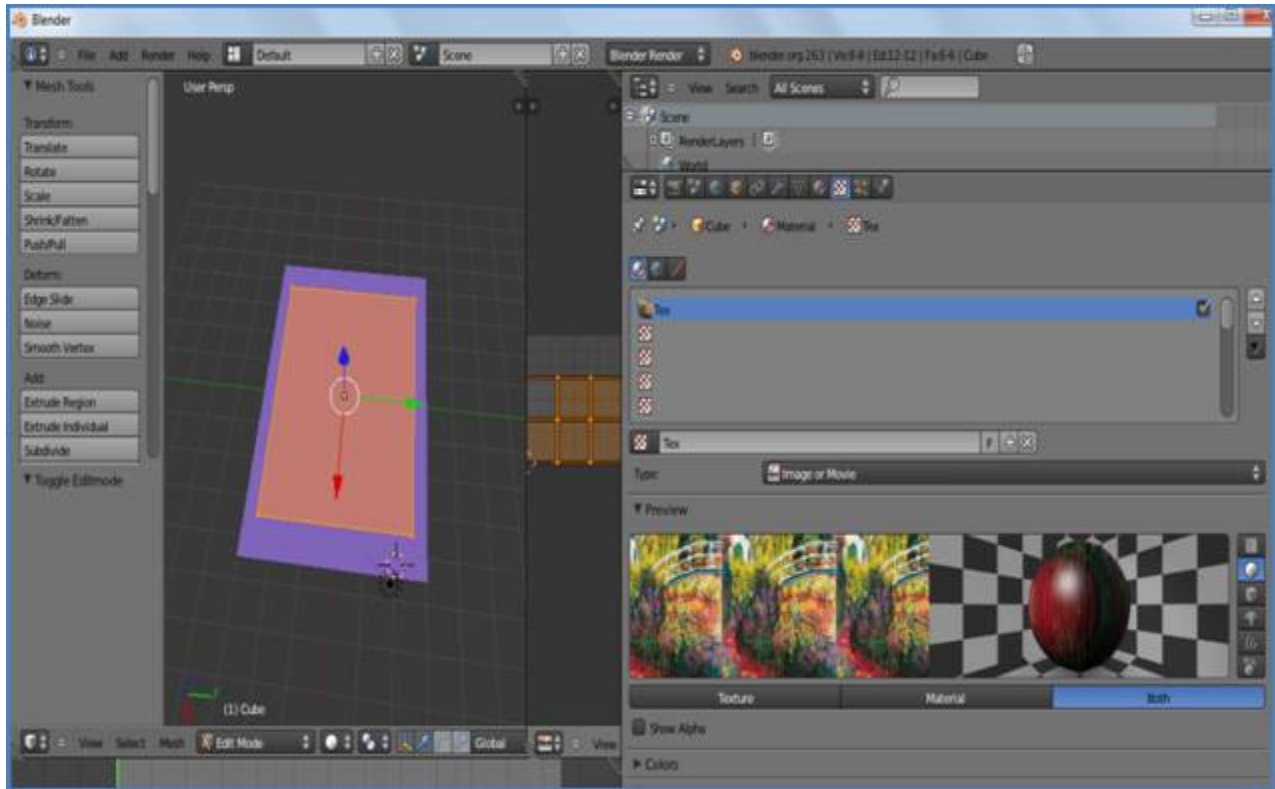


Fig. 1 (Vista del proyecto en blender y materiales y propiedades del modelo)

Ya que fueron elegidas todas las propiedades de los objetos se aplica el renderizado de la escena y obtenemos nuestro modelo.

||@|@

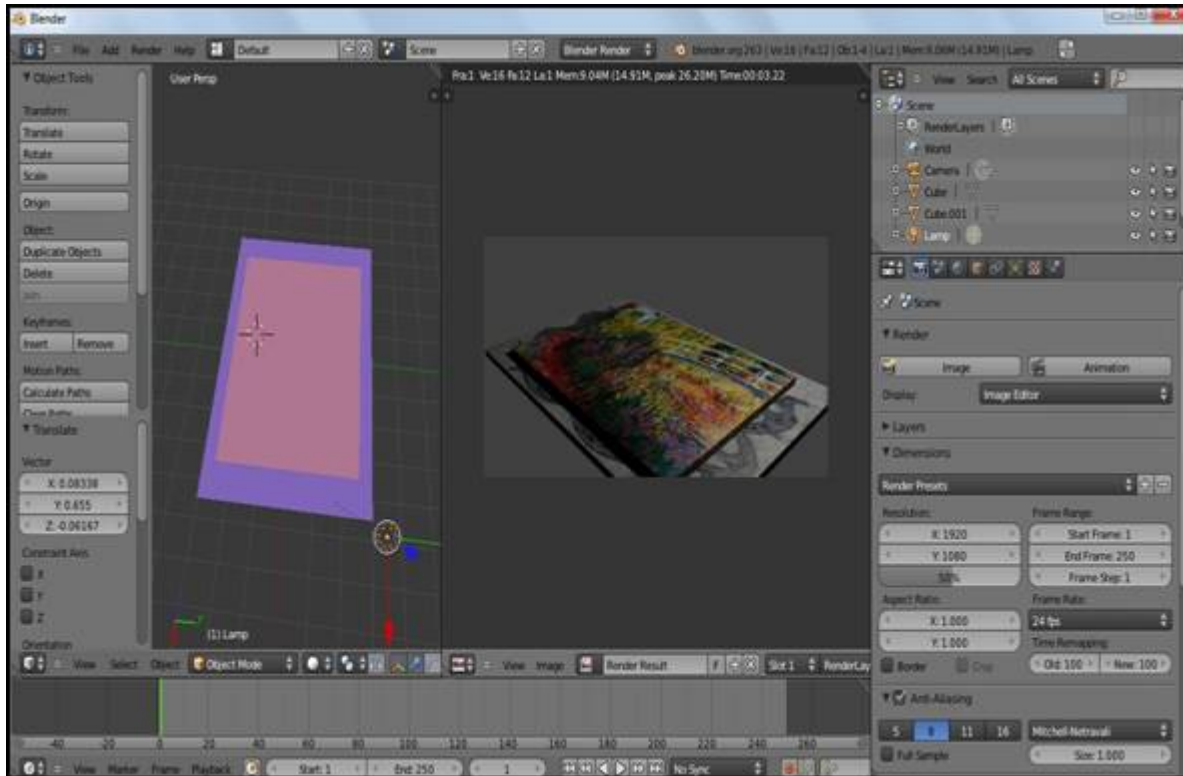


Fig. 2 (Modelo renderizado)

El modelo que se crea se guarda como un proyecto con extensión blend de esta manera es posible importarlo en la herramienta unity. El proyecto de blender guarda todas las propiedades del modelo y puede ser modificado cuantas veces sea necesario.

Para utilizar modelos que hemos creado en blender para realidad aumentada se puede utilizar la herramienta Unity con las extensiones de Qualcomm (se debe contar con la versión completa de Unity o de otro modo no permitirá desarrollar para Android) que podemos descargar de la liga <https://ar.qualcomm.at/qdevnet/sdk/android>, abriendo Unity debemos importar uno de los paquetes que acabamos de bajar, así que vamos a la opción de Assets y en Import Package escogemos Custom Package para localizar alguno de los paquetes, para empezar está bien si elegimos ImageTargets, ahora debemos elegir en la ventana de project el proyecto ImageTargets, podemos ver en la ventana de Scene que tenemos tres tetras sobre un plano diferente cada uno, lo que haremos ahora es cambiar alguna de estas tetras por el modelo que hemos creado en Blender.

||@|@

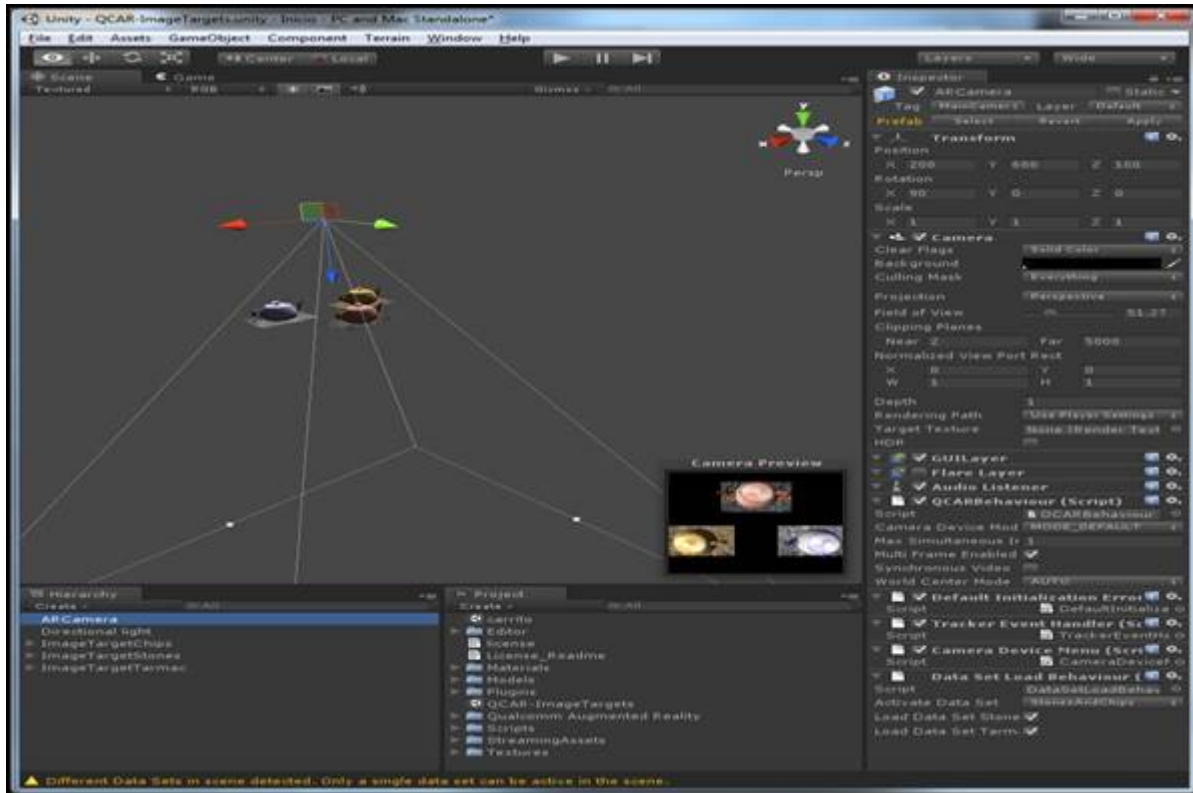


Fig. 3 (Vista de Proyecto ImageTargets)

Para importar nuestro modelo podemos ir a la carpeta de nuestro proyecto Unity y seguir la siguiente ruta: Assets>Models>Textures es aquí donde pondremos las imágenes de nuestro modelo, y dentro de una carpeta que se llame 3D (puede quedar sin la carpeta) pondremos nuestros modelos, ahora en Unity encontraremos los modelos y las imágenes que hemos puesto basta con arrastrar de la ventana Project a Hierarchy, pero debemos colocarlo dentro de un ImageTarget de los tres que tenemos, se decidió hacerlo sobre ImageTargetChips, pero en cualquiera funciona igual, eliminamos la tetra y desplegamos el modelo en ImageTargetChips, tendremos que poner ahora la imagen correspondiente a cada objeto de nuestro modelo, simplemente seleccionamos el objeto en la ventana de Hierarchy y en la ventana Inspector tendremos sus propiedades, para agregar la imagen correspondiente se deberá seleccionar de los recuadros Select Texture, donde veremos las imágenes si las hemos colocado en Texture correctamente.

Ya es visible el modelo pero demos escalarlo para que sea visible a la hora de ejecutarlo en el dispositivo móvil. Para escalar el modelo, en la ventana de inspector nos vamos a donde dice Scale y cambiamos las opciones en X, Y, y Z, esto dependerá mucho del tamaño del modelo que importamos, así que debemos jugar un poco con esto.

||@|@

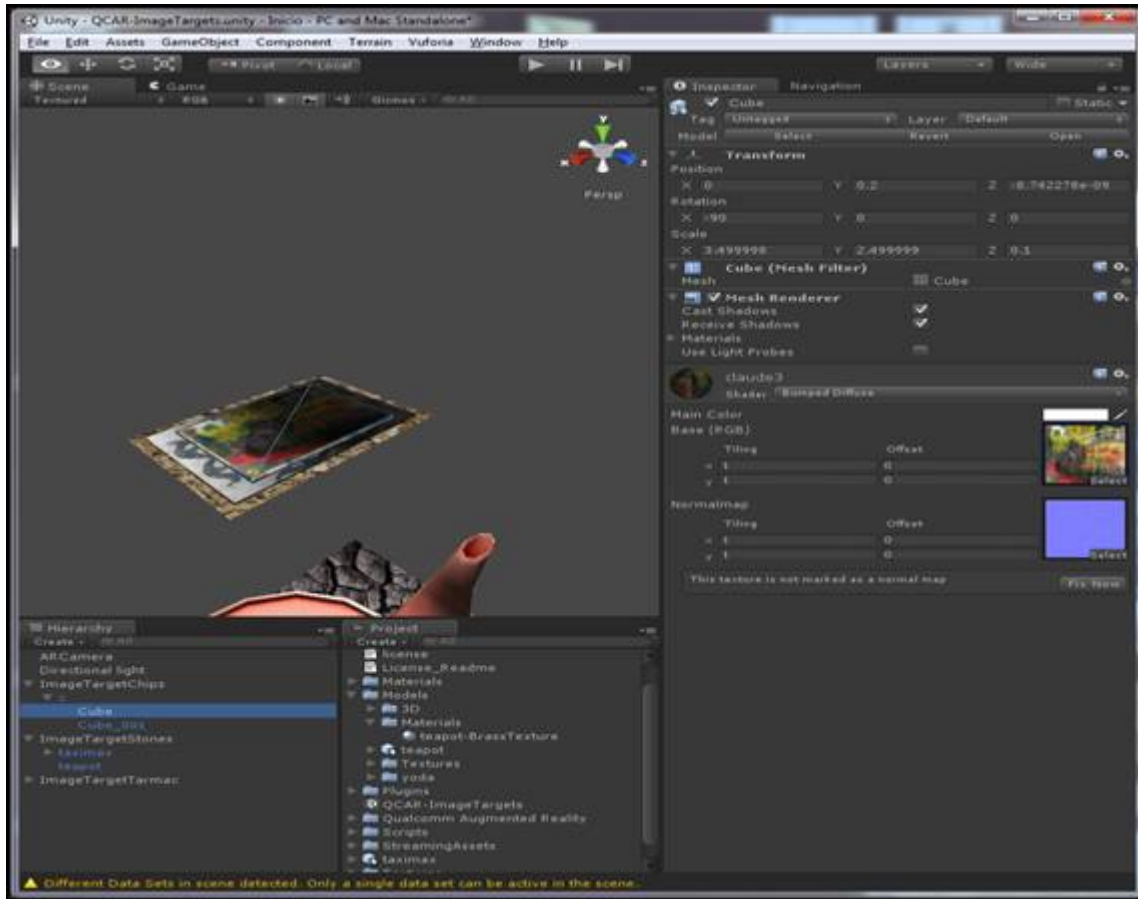


Fig. 4 (Escalamiento del modelo)

Conclusiones

Utilizar un proyecto de blender para crear un modelo nos da la versatilidad de hacer cambios en las propiedades como son tamaño, color, textura etc. Se integra directamente a las carpetas de Assets y Models del proyecto en el que se esté trabajando en unity y permite manipularlos directamente en unity.

Es posible incorporar este trabajo cualquier otros de los ejemplos de Qualcomm Vuforia, se deben seguir los mismos pasos para importar el modelo, y si queremos cambiar el patrón por alguno propio se puede hacer siguiendo las instrucciones en <https://ar.qualcomm.at/qdevnet/sdk/android>, My Trackables (para acceder se debe tener una cuenta de Vuforia desarrollador).

Hasta ahora hemos trabajado con ejemplos estáticos, pero se pretende incorporar acciones a los modelos de tal forma que introduzcamos un algoritmo de ruta corta.

Referencias

- [1] <http://www.robotis.com/xe/darwin> en
- [2] Brushless DC (BLDC) Motor Fundamentals, Padmaraja Yedamale Microchip Technology Inc.
- [3] Técnicas de control para motores Brushless Comparativa entre conmutación Trapezoidal, conmutación Sinusoidal y Control Vectorial, Roger Juanpere Tolrà.

Referencias

- [1] Albert Einstein, Isaac Newton, Marie Curie, Galileo Galilei, Charles Darwin (*mayo - junio, 2025*) *La teoría de la evolución biológica. Boletín UPIITA. año 19, (108) 2025* [liga del artículo](#)