

Aplicación móvil para mostrar sitios turísticos empleando realidad aumentada y geolocalización

Jonathan García Rosas, Rafael de la Rosa Flores, Hilda Castillo Zacatelco, Ana Patricia Cervantes Márquez

Facultad de Ciencias de la Computación Benemérita Universidad Autónoma de Puebla.
Av. San Claudio y 14 Sur, Ciudad Universitaria, Puebla, Puebla, México.
jesyehil@hotmail.com, {rafael, patty, hilda}@cs.buap.mx

Resumen. Las aplicaciones de realidad aumentada en móviles están enfocadas al uso de marcadores artificiales que consisten en un método invasivo en exteriores para que el usuario los enfoque y despliegue la información que dichos marcadores contengan. Se usa localización espacial por geoposicionamiento para mostrar realidad aumentada sobre determinadas áreas de la ciudad de Puebla, así como tecnologías de reconocimiento espacial usando visión artificial sin marcadores para mostrar objetos en 3 dimensiones sobre la fachada de la Catedral de Puebla

Palabras Clave: Realidad aumentada, geolocalización, marcadores

1. Introducción

La realidad aumentada es una técnica usada para combinar el mundo real con elementos virtuales en un dispositivo donde actúa primordialmente una cámara [1]. Además de estar ligada a dispositivos especiales que simulan objetos virtuales, la realidad aumentada se puede ver en aplicaciones de computadora [3] y, gracias a la proliferación de dispositivos móviles, se puede ver también en aplicaciones que hacen uso de sensores del dispositivo para dar una mayor muestra de convencimiento a los usuarios que hacen uso de ella. Gran parte de las aplicaciones de realidad aumentada están enfocadas al uso de marcadores artificiales, siendo pocas las que hacen usos de servicios de geolocalización y reconocimiento de patrones sobre marcadores naturales.

En este artículo se presentan dos formas para mostrar realidad aumentada. La primera de ellas consiste en geoposicionar marcadores sobre un navegador de realidad aumentada para mostrar su ubicación en tiempo real, mientras que la segunda forma emplea librerías y algoritmos de Vuforia para ubicar la fachada de la catedral de Puebla y superponer un objeto tridimensional al enfocar con la cámara [2]. Ambos desarrollos son empleados en un dispositivo de la línea Smartphone para el sistema operativo Android. Estas tecnologías fueron aplicadas en el sector turístico haciendo énfasis en exhibir los principales sitios de interés del centro histórico de la ciudad de Puebla. Se usan conexiones a un servicio web [4] que devuelve una lista de marcadores y

también se usa la API de Google Maps para descargar los mapas donde se muestran dichos marcadores.

2. Proceso de interacción

Esta sección describe la interacción que existe entre la aplicación ejecutada en el dispositivo móvil, los servicios de geolocalización usados y los servicios web para la obtención de información.

Al ejecutarse la aplicación se solicita la posición mediante GPS, GLONASS o A-GPS de acuerdo a las características del móvil, una vez obtenida, se envía al servidor web junto con el radio en kilómetros para verificar los puntos cercanos a la posición y devuelve los marcadores en formato JSON al cliente. Cuando el cliente recibe los datos correspondientes, se almacenan en el móvil y se muestran en pantalla auxiliados de la cámara del dispositivo, esto es, sobreponiendo la información sobre la pantalla e interactuando cada vez que el móvil cambia de dirección de enfoque. Se hace una solicitud al servidor de Google Maps para ubicar en un mapa los marcadores recibidos. La aplicación puede mostrar el mapa de forma normal o auxiliada con la cámara trasera del móvil mostrar los marcadores recibidos con realidad aumentada. Se despliegan información almacenada en el móvil de cada sitio y animaciones y objetos 3D en determinados lugares importantes (Figura 1).

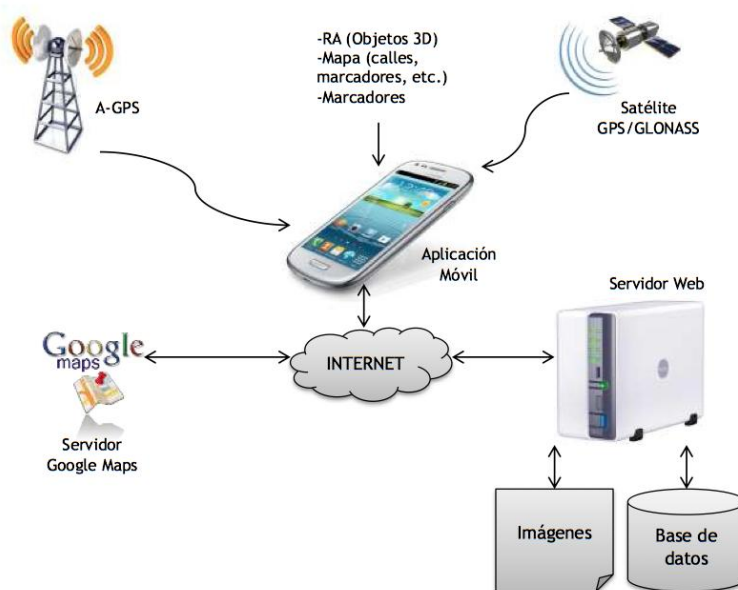


Fig. 1. Interacción del sistema.

El sistema requiere como entrada la ubicación del móvil con tres parámetros: latitud y longitud, la cual es devuelta por el GPS, GLONASS o A-GPS, y un radio de

alcance en kilómetros. Después de obtener la ubicación obtendrá la lista de lugares o marcadores cercanos (de acuerdo al radio deseado) desde el servidor en formato JSON. Un ejemplo de petición hacia el servidor sería "19.00504,-98.204773,2", donde los parámetros corresponden a la latitud, longitud y el radio de alcance respectivamente. Los marcadores recibidos se muestran en tiempo real en la pantalla del dispositivo tomando en cuenta su ubicación geográfica y son almacenados en caché mientras la aplicación esté activa.

2.1. Formatos del navegador RA

La ubicación del sitio será enviada desde el dispositivo móvil al servicio web REST de la siguiente forma:

http://148.228.xx.xx/ServicioWebRest/Api/Sitios/Sitio/latitud,longitud,radio

Donde latitud y longitud corresponden a la ubicación actual del usuario, y radio corresponde a la distancia en kilómetros a la redonda a la que se quiere obtener marcadores.

Ejemplo de envío:

http://148.228.xx.xx/ServicioWebRest/Api/Sitios/Sitio/19.00504,-98.204773,2

El servidor devuelve una lista de sitios turísticos (denominada geoCTI) calculados de acuerdo al radio que se envió como parámetro. Cada elemento de la lista tiene los siguientes parámetros: id, id_categoria, summary, tittle, elevation, lng, lat, url_imagen

Donde id es el identificador del marcador, id_categoria es el identificador para la categoría de dicho marcador, summary tiene un pequeño resumen del marcador, tittle contiene el nombre del marcador, elevation tiene la altitud a la que se encuentra geográficamente el marcador, lng es la longitud de acuerdo a la posición geográfica del marcador, lat es la latitud de acuerdo a la posición geográfica del marcador y url_imagen contiene un vínculo de una imagen del sitio que almacena el marcador.

Un ejemplo del marcador devuelto por el servicio web es el siguiente:

```
"Id":2,"id_categoria":1,"summary":"Fuente de la facultad de ciencias de la computación","tittle":"Fuente FCC","elevation":2100,"lng":-98.20451,"lat":19.005008,"url_imagen":http://cs.buap.mx
```

3. Diseño del sistema

En esta sección se describe el sistema a partir de bloques que fueron generados para dividir las tareas que se desarrollan en el sistema.

El sistema se compone de 5 bloques principales: Presentación, Navegación RA, Mapas, Visualización RA y, Servicios web. Cada bloque cuenta con distintas capas que ejecutan diversas funciones. Para el bloque de presentación, las capas se encargan de la comunicación con el servidor web y la interacción con el usuario. En el caso del bloque de Navegación RA, contiene las capas encargadas de la manipulación con el hardware del dispositivo y las operaciones que se realizan para mostrar la realidad aumentada. El bloque de Mapas es el encargado de mostrar los mapas. El bloque de

visualización RA es el encargado de gestionar las librerías de Vuforia para mostrar la realidad aumentada con vistas en 3D. Por último, el bloque de Servicio Web [4] contiene las capas encargadas de responder a peticiones por parte de los usuarios, así como interactuar con la base de datos de marcadores. En la Figura 2 se observa el diagrama por bloques del sistema y la interacción entre bloques.

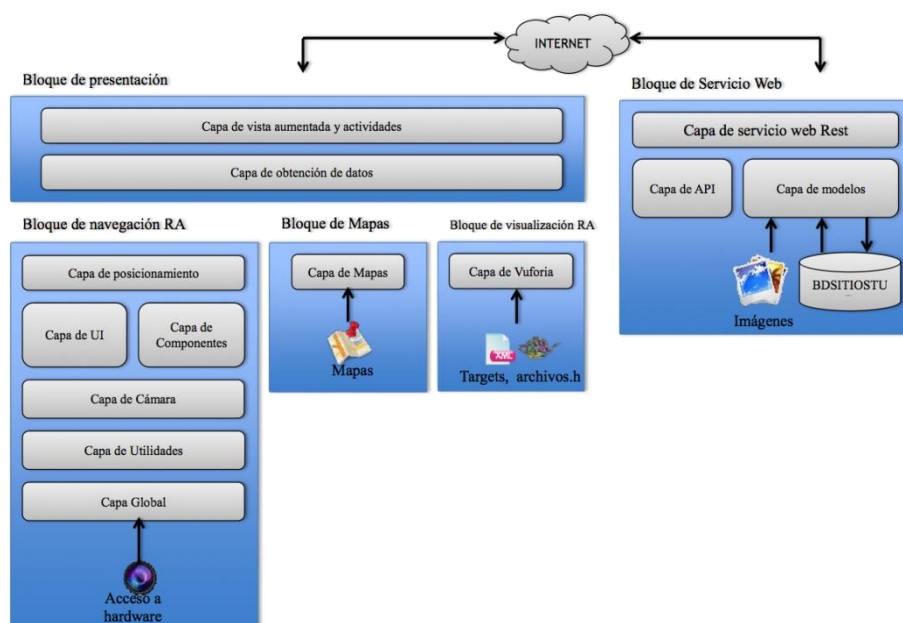


Fig. 2. Diagrama de bloques del sistema.

A continuación se describe cada bloque con sus respectivas capas.

Bloque de presentación. Este bloque es el encargado de la comunicación con el servidor web Rest así como la interacción entre el sistema y el usuario. Contiene las siguientes capas:

- *Capa de vista aumentada y actividades:* Esta capa será la encargada de mostrar todas las interfaces al usuario. Para la visualización del navegador de realidad aumentada debe tener presente el uso del bloque de navegación RA. Para la visualización de objetos en 3D se debe comunicar con el bloque de visualización RA. La visualización de mapas se realiza en conjunto con el bloque de Mapas.
- *Capa de obtención de datos:* Esta capa es la encargada de la comunicación con el servidor web para obtener los sitios turísticos.

Bloque de Navegación RA. Este bloque se encarga de ejecutar el motor de navegación de realidad aumentada.

- *Capa de posicionamiento.* Representa la ubicación física del dispositivo móvil y la disposición de los elementos en pantalla.
- *Capa de UI.* Es la encargada de dibujar los elementos como líneas, puntos, círculos, textos e imágenes.
- *Capa de componentes.* Gestiona los elementos que se muestran en pantalla, dichos elementos son: Marcadores, radar y barra de zoom.
- *Capa de cámara.* Obtiene el acceso a la cámara del dispositivo y contiene las clases para mostrar la imagen capturada en pantalla en tiempo real.
- *Capa de utilidades.* Contiene las clases que realizan las operaciones matemáticas necesarias para mostrar marcadores y calcular la inclinación del dispositivo.
- *Capa global.* Esta capa se encarga del almacenamiento general de los sitios mientras corre la aplicación.

Bloque de Mapas. Este bloque es el encargado de controlar la comunicación con el API de Google Maps [8] para mostrar mapas.

- *Capa de mapas.* Esta clase muestra los mapas y los sitios turísticos.

Bloque de visualización RA. Este bloque es el encargado de mostrar la realidad aumentada a partir de marcadores naturales usando las librerías de Vuforia.

- *Capa de Vuforia.* Esta capa se encarga de renderizar y mostrar un objeto 3D, así como del preprocesamiento digital de una imagen en tiempo real.

Bloque de Servicio Web. Este bloque contiene todas las capas para el manejo de solicitudes en un servicio web. Cuenta con las siguientes capas:

- *Capa de Servicio Web Rest.* Esta capa será la encargada de recibir solicitudes desde el equipo móvil. Es capaz de devolver la información de los sitios cercanos a la posición desde donde se hizo la solicitud.
- *Capa de modelo.* Esta capa contiene las operaciones y consultas que se realizan con la base de datos y se encarga de procesar la información que se reciba para devolverla en el formato correcto.
- *Capa de API.* Esta capa redirige las acciones hacia el Controlador del servicio web a partir de la solicitud url.

4. Implementación

En esta sección se muestra la implementación de la aplicación Android y el servicio web, ambos casos de acuerdo a sus bloques correspondientes. Se tomaron en cuenta distintas tecnologías de desarrollo. En el caso del desarrollo de la aplicación en Android, se programaron las clases en Java y las interfaces en XML con el IDE Eclipse siguiendo el modelo vista controlador [5]. Se usó el lenguaje de programación C++ para los objetos 3D que se despliegan mientras se observa la realidad aumentada,

además de usar las librerías de Vuforia. En el caso del servicio web, se implementó un Servicio web Rest con las tecnologías de ASP.NET MVC para la programación [6] y SQL Server para el almacenamiento de los sitios turísticos.

De acuerdo al diagrama de bloques del sistema que se muestra en la sección de Diseño, los cuatro primeros bloques contienen las clases para la aplicación móvil, mientras que el último bloque contiene las clases para el servidor web Rest.

4.1. Bloque de presentación

La capa de vista aumentada y actividades contiene las vistas desarrolladas en XML y las que se crean en tiempo de ejecución, las cuales muestran tres opciones de navegación: navegador RA, catedral y Acerca de (Figura 3).



Fig. 3. Interfaz de la actividad inicios.

Para la capa de obtención de datos, se tienen las clases que realizan las conexiones para traer los marcadores desde el servicio web REST usando una dirección URL como se muestra en el siguiente ejemplo:

<http://148.228.xx.xx/ServicioWebRest/Api/Sitios/Sitio/>

4.2. Bloque de navegación RA

En este bloque se crean las clases para el motor de navegación de realidad aumentada. Este contiene la capa de posicionamiento, que se usa para determinar la posición geográfica del dispositivo (con los parámetros latitud, longitud y altitud), su posición física y la posición de los objetos en pantalla (atributos x y y).

La capa de Interfaz de usuario (UI) Es la encargada de dibujar todos los componentes que se muestran en la interfaz. Los objetos que se dibujan en tiempo de ejecución son los siguientes: Cajas (Figura 4), Textos (Figura 4), Imágenes, Círculos (Figura 5), Íconos (Figura 6), Líneas (Figura 5), Puntos (Figura 5).

Las siguientes Figuras ilustran los objetos que pueden ser creados en la capa de Interfaz de usuario.



Fig. 4. Caja con texto con referencia de un marcador



Fig. 5. Circulo dibujado para simular un radar con dos líneas, además de puntos de posiciones.



Fig. 6. Ejemplo de ícono de un marcador de la categoría monumento.

La capa de componentes se encarga de fusionar los dibujos realizados en la capa de interfaz de usuario. Se muestran tres componentes principales en pantalla: Radar (Figura 7), Marcador (Figura 8) y Barra vertical (Figura 9).

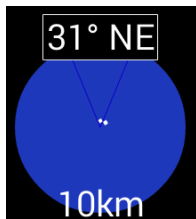


Fig. 7. Radar en la interfaz de navegación RA.



Fig. 8. Marcador en la interfaz de navegación RA.



Fig. 9. Barra vertical de zoom (SeekBar).

La capa de cámara es la encargada del acceso a la cámara en el dispositivo Android. La capa de Utilidades se encarga de las operaciones matemáticas necesarias para posicionar los objetos en la pantalla mediante el uso de matrices, y además se encarga de la obtención de los ángulos de inclinación del dispositivo.

La capa global es la encargada de almacenar los marcadores e imágenes que se usan mientras la aplicación esté corriendo.

4.3. Bloque de mapas

En este bloque se gestionan las conexiones con el API de Google Maps [8] para mostrar los mapas en una actividad y para mostrar rutas entre la ubicación del dispositivo, y un marcador en concreto.

La capa de mapas muestra el mapa con la ubicación del dispositivo móvil además de los marcadores con los sitios turísticos en un radio cercano a la ubicación del dispositivo. Tiene la particularidad de trazar rutas desde el punto en que se encuentra el dispositivo y otro marcador seleccionado mediante el API de rutas Google Maps [8].

4.4. Bloque de visualización de RA

Este bloque contiene las clases de Vuforia para el procesamiento de imágenes y la renderización de un objeto en tres dimensiones para superponerlo sobre una escena, en este caso, la fachada de la catedral de Puebla.

La capa de Vuforia se encarga del procesamiento de imágenes en tiempo real desde la cámara del dispositivo. Su función es encontrar el marcador en una escena y superponer un objeto renderizado en 3D para su visualización [7]. La creación de marcadores se realiza desde la página de Vuforia (Target Manager Vuforia). Para ello se necesitan fotografías del lugar en el que deseamos mostrar la realidad aumentada. Este sistema nos devuelve un archivo binario .dat y otro XML para referenciar el marcador cuando se ejecute la aplicación. En la Figura 10 se muestra uno de los marcadores usados para la catedral de la ciudad de Puebla.

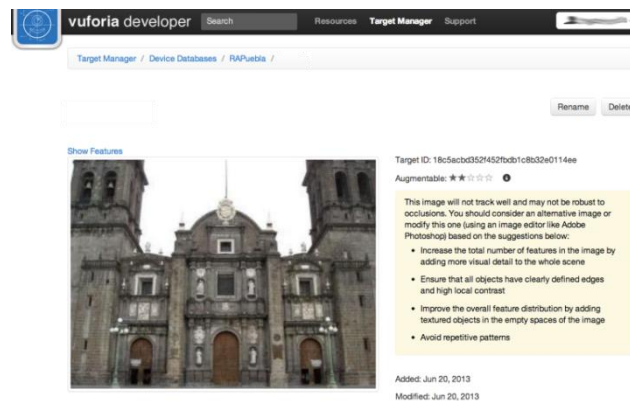


Fig. 10. Marcador implementado para la catedral de Puebla. Está almacenado en un archivo binario .dat al que se accede desde el tracking generado en XML.

Para la creación del objeto en 3D que será visualizado sobre la fachada, se usa Blender para modelar el objeto, determinar su textura y su posición inicial. En la Figura 11 se observa su perspectiva de diseño.

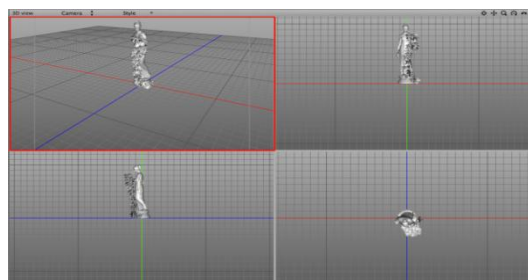


Fig. 11. Perspectivas el diseño de un objeto en 3D con el software Blender. Este objeto es usado para mostrarlo sobre el marcador al que se apunte con la cámara.

4.5. Bloque de servicio web

Este bloque maneja las peticiones que se realizan por medio del método GET de http, gestiona las solicitudes con la base de datos, además de realizar los cálculos para determinar la distancia a la que se encuentran los marcadores de la posición enviada y devolver los marcadores que correspondan al radio solicitado.

La capa de servicio web Rest contiene los controladores para recibir peticiones desde el dispositivo móvil o desde cualquier navegador web, con la particularidad de serializar una lista de marcadores que se devuelven en formato JSON. Esta capa, fue concebida como una API para facilitar el acceso a los marcadores y su uso a la conveniencia que se disponga. Un ejemplo de objeto (denominado geoCTI) serializado que se envía de retorno es el siguiente:

```
{
  "geoCTI":
  [
    {
      "Id":2,
      "id_categoria":1,
      "summary":"Fuente de la facultad de ciencias de la compu-
tación",
      "tittle":"Fuente FCC",
      "elevation":2029,
      "lng":-98.20451,
      "lat":19.005008,
      "url_imagen":"http://cs.buap.mx"
    }
  ]
}
```

5. Implementación

La aplicación puede ser instalada al generar un archivo .apk desde el entorno de desarrollo (para este caso, se uso el IDE Eclipse). Al arrancar la aplicación en un dispositivo con Android, se muestra la interfaz de inicio con una actividad por dos segundos. Destacan el título con el nombre la aplicación, el ícono de la aplicación así como el logotipo del CTI y de Vuforia (Figura 12).

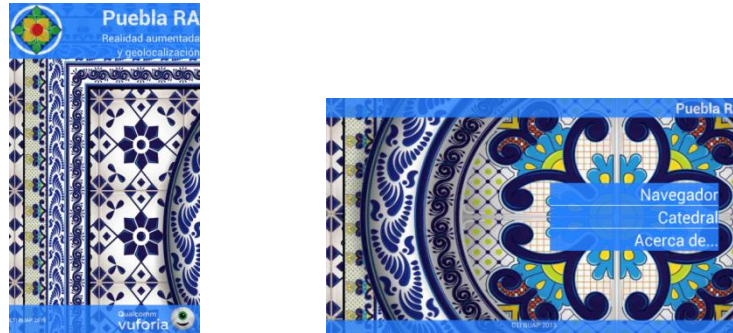


Fig. 12. Arranque de la aplicación (izquierda) y menú (derecha)

5.1. Navegador RA

El navegador de realidad aumentada se muestra accediendo desde el menú en la opción Navegador. Esto mostrará una actividad como se observa en la Figura 13.



Fig. 13. Vista principal del navegador de realidad aumentada.

El navegador RA contiene tres componentes que se describen a continuación:

- *Radar*. Muestra todos los marcadores que se encuentren dentro del límite elegido con la barra de zoom. Cada marcador o sitio turístico es mostrado como un punto de color blanco y cambian de acuerdo a la orientación del dispositivo. En la parte superior del Radar se observa la orientación actual del dispositivo, mientras que en la parte inferior se muestra el radio en metros o kilómetros a los que se desea ver marcadores (Figura 14).

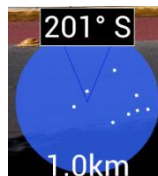


Fig. 14. Vista del Radar en el navegador con orientación hacia el sur y un radio de 1 km.

- *Barra de zoom.* Se encarga de incrementar o disminuir el radio al que se desean ver los marcadores. El radio máximo de visualización es de 10 kilómetros. Al incrementar o disminuir el valor de la barra de zoom, el valor del Radio también cambia. La Figura 15 muestra la barra de zoom de la aplicación.



Fig. 15. Barra de zoom para cambiar el radio.

- *Marcadores.* Los marcadores se muestran sobre la vista de la pantalla de acuerdo a su enfoque en determinada orientación. Todos los marcadores se muestran al enfocar hacia un lugar con el dispositivo móvil en posición horizontal. Estos son mostrados con el nombre del marcador y un ícono de acuerdo a su categoría. En la Figura 16 se observan 2 marcadores al enfocar al sureste.

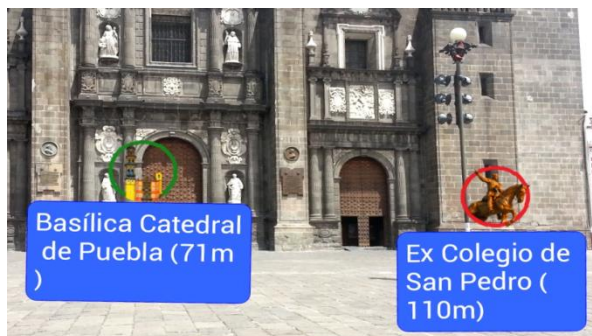


Fig. 16. Marcadores o sitios enfocados al sureste.

Al pulsar sobre un marcador, nos muestra un cuadro de diálogo en el cual vemos el nombre del marcador, una fotografía de referencia del sitio y la descripción del sitio turístico. También tiene la opción “Como llegar” para visualizar la ruta más cercana a pie para llegar al sitio (Figura 17).



Fig. 17. Al pulsar un marcador, se muestra información del mismo con la opción de trazar una ruta para llegar a él.

5.2. Mapa

Cuando giramos nuestro dispositivo de forma que quede en paralelo al suelo, se muestra el mapa con nuestra ubicación y todos los marcadores que se encuentren dentro del radio deseado. Cada marcador es representado con el ícono de su categoría (Figura 18).



Fig. 18. Mapa con marcadores.

Al elegir la opción “Como llegar” sobre un marcador, se muestra la ruta más corta en el mapa para llegar a dicho sitio (Figura 19).



Fig. 19. Ruta a pie para llegar desde la ubicación actual hacia el marcador elegido.

5.3. Catedral

Al elegir la opción Catedral desde el menú principal, nos muestra una vista con las indicaciones para visualizar la realidad aumentada en la catedral de la ciudad de Puebla. Para ello, se presiona el botón Iniciar.

Al iniciar esta actividad, se muestra una vista de la cámara del dispositivo. La principal diferencia entre la vista de esta cámara con la vista del navegador RA es la precisión para auto enfocar auxiliado del hardware del dispositivo (en el caso en que la cámara del dispositivo cuente con autoenfoque). Para mostrar la realidad aumentada es necesario seguir las siguientes indicaciones:

1. Ubíquese frente a la fachada principal de la catedral.
2. Apunte con el dispositivo con dirección a la fachada.
3. Visualizar la realidad aumentada que se despliega.

En la Figura 18 se observa el objeto estatua mostrado al enfocar la catedral de la ciudad de Puebla. En la Figura 19 se observa el mismo objeto en modo apaisado del dispositivo.



Fig. 20. Objeto estatua superpuesto sobre la fachada de la catedral.



Fig. 21. Objeto estatua superpuesto sobre la fachada de la catedral en modo apaisado.

6. Conclusiones y perspectivas

En este trabajo se presentan dos alternativas de realidad aumentada empleada como método de divulgación de sitios de interés, además de métodos de geolocalización para ubicar cada sitio. Mediante el empleo de estas tecnologías se creó una aplicación para Android que aprovecha los recursos y sensores con los que cuentan los dispositivos móviles hoy en día. Al ser los recursos de hardware con los que cuenta el móvil uno de los aspectos más importantes a la hora de desarrollar aplicaciones, estos se aprovechan al máximo puesto que son los encargados de realizar tareas de reconocimiento de imágenes en tiempo real, renderización de objetos en tercera dimensión, así como posicionamiento en pantalla de objetos en 2D, tareas que ocupan un gran porcentaje de la memoria y el poder de procesamiento del procesador del dispositivo. El consumo de recursos de memoria es muy alto para este tipo de tareas y la aplicación desarrollada solo está disponible para dispositivos que cuenten con ciertas características de hardware. El sistema cuenta con un límite de veinte marcadores que se muestran en tiempo real, esto debido a la carga de objetos que se muestran puede resultar muy grande en cuanto al consumo de memoria.

Las tecnologías usadas en este proyecto son tanto de uso libre como privativas. Es necesario mencionar que se usaron marcas registradas como es el caso de Vuforia, que exigen que un logotipo de dicha marca sea mostrada cuando se usa su tecnología para desarrollar y mostrar realidad aumentada. Otro caso es el uso de la API de Google Maps, la cual tiene un límite de 20,000 consultas diarias desde la aplicación y además se exige que los datos que sean consultados a sus servidores (como el uso de las rutas) sean expresamente utilizados para visualizarlos en un mapa.

Otro aspecto a tomar en cuenta es que la aplicación desarrollada necesita de una constante conexión a internet para descargar la información desde un servidor. Cuando se eligió la forma de transmisión de datos, se determinó usar el formato JSON, el cual es un formato muy popular de transmisión así como uno de los que menos tráfico genera a la hora de realizar las consultas a un servidor. Aunque el sistema está preparado para funcionar con redes WiFi, se debe contemplar el uso de datos por parte de un proveedor de internet para móviles que garantice una conexión a internet en todo momento. Otro aspecto a tomar en cuenta es el uso de un chip GPS instalado en el dispositivo, puesto que la aplicación está preparada para funcionar sin la necesidad del mismo, el uso de este servicio mejorará la precisión de la ubicación del dispositivo y los sitios turísticos a su alrededor.

El servicio web se desarrolló con la finalidad de que cualquier persona pueda consultar los lugares que se encuentran a su alrededor, pero con la limitación de obtener solo los 20 lugares más cercanos de acuerdo a su posición o la posición que ellos especifiquen, pero existe la opción de consultar todos los lugares que se encuentren en la base de datos, con la limitantes de que cada persona debe ocupar y manipular los datos de acuerdo a su conveniencia. Dicho servicio web fue probado en el laboratorio del Centro de Tecnologías de la Información de la Facultad de Ciencias de la Computación de la BUAP.

Referencias

1. Raghav Sood. Pro Android Augmented Reality (2012) Chapter 1, Applications of Augmented Reality. Pages 1-12. Chapter 2, Basics of Augmented Reality on the Android Platform. Pages 13-31.
2. Lee, T., Höllerer, T. (2007) Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking. IEEE International Symposium on Wearable Computers (ISWC '07).
3. Kato, H., Billinghurst, M.(1999) Marker tracking and HMD calibration for a video-based augmented reality conferencing system, Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR 99).
4. Alfonso Felipe Lima Cortés (2012): Desarrollo de aplicación en Android con acceso a Web Service. Introducción al desarrollo de aplicaciones móviles en Android. Tercer congreso regional en TIC.
5. Trygve Reenskaug and James Coplien, (2009). The DCI Architecture: A New Vision of Object-Oriented Programming, www.artima.com/articles/dci_vision.html
6. Simple Example of MVC (Model View Controller) Design Pattern for Abstraction. <http://www.codeproject.com/Articles/25057/Simple-Example-of-MVC-Model-View-Controller-Design> Consultado Abril de 2013.
7. Vuforia developer Resources. Vuforia SDK Architecture. <https://developer.vuforia.com/resources/dev-guide/vuforia-ar-architecture> Consultado Agosto de 2013.
8. API de Google Maps, <https://developers.google.com/maps/?hl=es>, Consultado Noviembre del 2014.