

USO DE LA HERRAMIENTA ARTOOLKIT PARA LA SIMULACIÓN DE EVASIÓN DE OBSTÁCULOS

José Fidel Navarro Aguilera, Israel Rivera Zárata, Mauricio Olguín Carbajal
CIDETEC --- Instituto Politécnico Nacional

Resumen

Este trabajo muestra un claro ejemplo de Realidad Aumentada, donde un móvil en un ambiente simulado por medio de la herramienta ARToolKit evadirá los obstáculos que se le interpongan en su camino y así logrará llegar a su destino. Una característica de este trabajo será la realización del código para la evasión de obstáculos.

Palabras Clave: Realidad Aumentada, ARToolKit.

I. Introducción

A través del tiempo se ha intentado que el ser humano deje de realizar actividades que pongan en riesgo su integridad. Por tal motivo se han desarrollado robots, ya sean en forma de vehículo, de insecto, humanoide, etc., que sean capaces de realizar esas tareas que para el hombre resultan imposibles o que son complejas.

Para que el robot o móvil realice una tarea eficiente, se hacen pruebas simuladas para ver qué problemas se encuentran y poder solucionarlos a nivel lógico sin tener que hacer el gasto innecesario en la parte física.

Algunos problemas observados eran que el móvil se encontrara con varios caminos y obstáculos, pudiendo éste tomar el mejor camino y esquivar exitosamente los obstáculos que se le presenten. Para esto se han formulado algunos algoritmos; por ejemplo, uno que hace referencia a una colonia de hormigas [?], en el que éstas, a través de una feromona que dejan por donde van pasando, sirve de guía para la que viene atrás, y así sucesivamente. El contratiempo encontrado es que esta feromona se evaporaba en determinado tiempo; en caso de que una hormiga se tardara, el rastro se perdería.

Otro ejemplo, en el que se emplearon robots futbolistas [?], tiene como objetivo introducir la pelota en la portería rival y esquivar a los contrarios. Con el algoritmo desarrollado se buscó que el móvil diferenciara los objetos de la pelota: para esto se creaba un punto destino y una frontera de evaluación formada por celdas; la que tuviera mayor valor de atracción sería el siguiente punto de avance.

Un ejemplo interesante es el mencionado en [?], donde podemos observar cómo el trabajo en equipo resulta útil. Se emplearon 3 robots: el *líder*, que es el más cercano al obstáculo; el *cargador*, que solo sigue al primer robot; y el *explorador (scout)*, que se encarga de inspeccionar el camino, tomar la decisión de qué ruta es la más adecuada y, si ésta fuese bloqueada, buscar otro camino.

II. Realidad Aumentada

La Realidad Aumentada (RA) es una técnica mediante la cual los usuarios pueden percibir la realidad superponiendo a los objetos reales modelos virtuales enriquecidos [?]. El observador puede trabajar y examinar objetos 3D reales mientras recibe información adicional sobre estos objetos o sobre la tarea que se está realizando. De este modo, la RA permite al usuario permanecer en contacto con su entorno de trabajo, mientras su foco de atención no está en la computadora, sino en el mundo real. El papel que juega la computadora es el de asistir y mejorar las relaciones e interacciones entre las personas y el mundo real.

La diferencia entre la Realidad Virtual (RV) y la RA es el tratamiento que hacen del mundo real. La RV sumerge al usuario dentro de un mundo virtual que reemplaza completamente al mundo real exterior, mientras que la RA deja ver al usuario el mundo real a su alrededor y aumenta la visión que éste tiene de su entorno mediante la superposición o composición de objetos 3D virtuales. Idealmente, esto daría al usuario la ilusión de que los objetos de los mundos real y virtual coexisten.

En definitiva, se podría decir que los sistemas de RA llevan el ordenador al entorno de trabajo real del usuario, mientras que los sistemas de RV intentan llevar el mundo real al interior del ordenador.

La Realidad Aumentada es utilizable en múltiples aplicaciones [?]:

- Pueden organizarse reuniones en las cuales los asistentes vayan equipados con gafas especiales conectadas al ordenador, de forma que todos visualicen un mismo mundo virtual y al mismo tiempo puedan verse entre ellos.
- El cirujano puede recibir información en sus gafas mientras manipula al paciente.
- Este tipo de sistemas lo hemos podido apreciar en películas de ciencia ficción.

III. Herramienta ARToolkit

ARToolkit es un conjunto de librerías para C/C++ que sirven para la creación de aplicaciones de realidad aumentada. Para ello proporciona una serie de funciones para la captura de vídeo y para la búsqueda de ciertos patrones en las imágenes capturadas mediante técnicas de visión por computador. También proporciona una serie de ejemplos y utilidades de gran ayuda al programador que quiera realizar este tipo de aplicaciones. Esta colección de librerías facilita el desarrollo de aplicaciones de Realidad Aumentada.

III.1 Instrucciones de instalación

Para el funcionamiento de esta herramienta fue necesario descargar:

- GLUT
- Visual Studio 6
- Open VRML
- Direct X v. 9 o superior
- DSVL

Ya instalado Visual Studio 8, se procede a descomprimir ARToolkit y colocarlo en la unidad C: en raíz. Después se descomprime el kit de GLUT y se copian los siguientes archivos:

- `glut32.dll` en `C:\Windows\system32`
- `glut.dll` en `C:\Windows\system`
- `glut.h` en `(*)\Include\gl\`
- `glut32.lib` en `(*)\Lib\`

El (*) es la ruta Archivos de Programa/VS8/VC/Include donde irán colocados.

Después se colocaron los archivos `DSVL.dll` y `DVLSd.dll`, incluidos en la carpeta original de `ARToolkit/DSVL/bin`, en `ARToolkit\bin`.

Se ejecuta el script que se encuentra dentro de `ARToolkit\Configure.win32` para crear `Include\AR\config.h`.

Luego se descomprime OpenVRML dentro de ARToolkit; en este se encuentra el archivo `js32.dll` en `OpenVRML/bin` y se coloca en `ARToolkit\bin`.

Es necesario abrir `ARToolkit.sln` o `ARToolkit.dsw`.

IV. Desarrollo

Se pretende crear un ambiente en realidad aumentada por medio de la herramienta ARToolkit, esto es que en un espacio cualquiera se distribuirán diversos objetos, los cuales serán observados mediante una cámara.

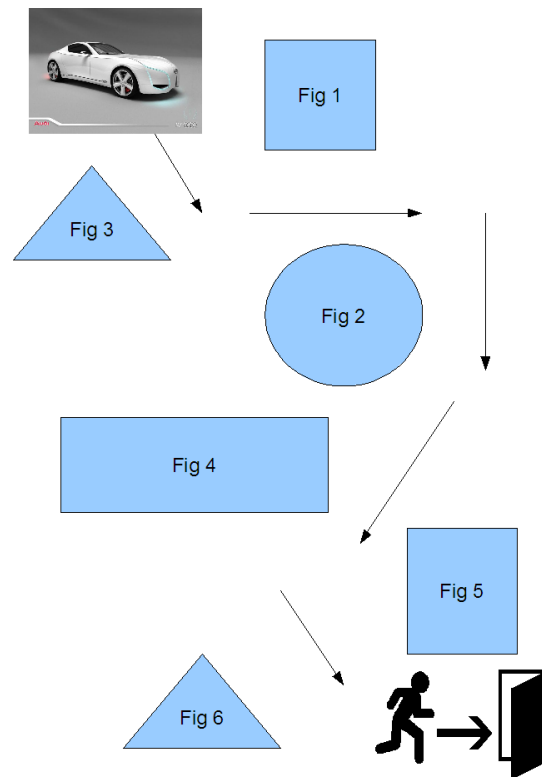


Figura 1 Toma de la cámara.

Al ser pasados a la computadora se lograría identificar las diferentes figuras físicas y se les asignará mediante la herramienta la figura que deseemos en 3D.

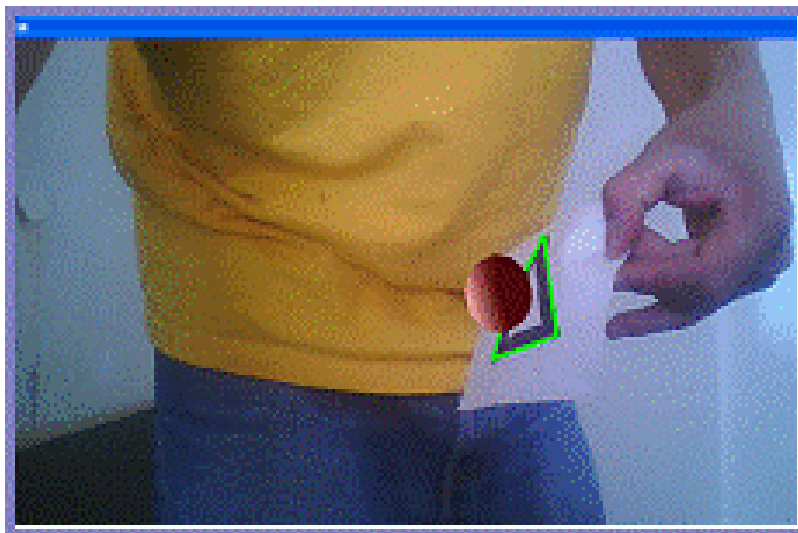


Figura 2 Toma 1: un triángulo y un círculo en 3D.

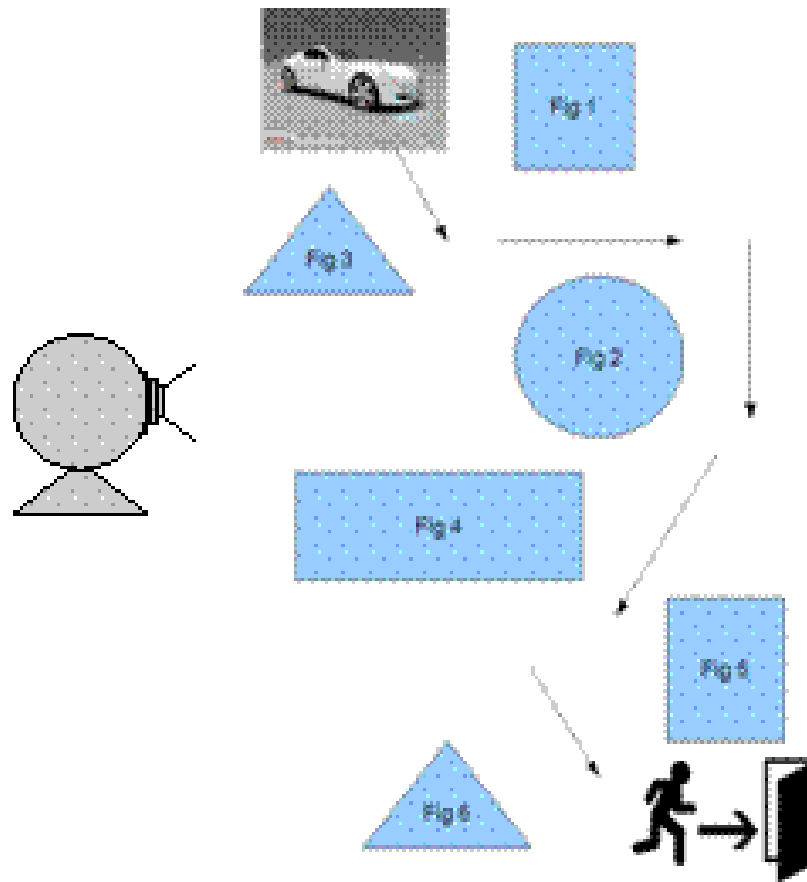


Figura 3 Toma 2: un cuadrado muestra un cubo.



Figura 4 Toma 3: se observa una tetera.

Después de esto se simulará el móvil que tendrá como objetivo esquivar cada uno de estos objetos mediante un programa de enrutamiento; esta sería la parte medular de este trabajo, ya que es nuestro punto a desarrollar.

IV.1 Características del vehículo

El vehículo al ser simulado nos permitirá variar su tamaño, siempre y cuando éste no se vea afectado por el ancho del camino; también nos brinda la oportunidad de ocupar cualquier tipo de móvil, ya sea un robot, insecto o un simple cochecito. De cualquier forma deberá esquivar los obstáculos.

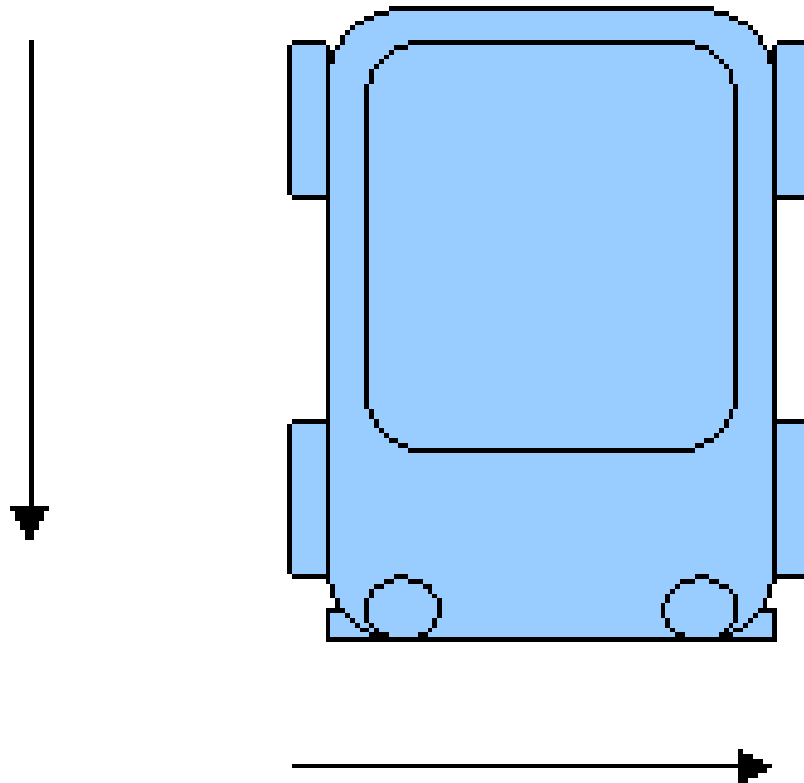


Figura 5 Ejemplo de vehículo con medidas (no es el móvil final).

Para realizar el código de ruteo se estudiarán más a fondo los siguientes ejemplos de algoritmos, para decidir cuál será el más apto para la aplicación que aquí se pretende realizar.

IV.2 Algoritmo ACO (*Ant Colony Optimization*)

El primero, mencionado en [?], describe los algoritmos ACO, los cuales son procesos iterativos: en cada iteración se «lanza» una colonia de m hormigas y cada una de ellas construye una solución al problema. Las hormigas construyen las soluciones de manera probabilística, guiándose por un rastro de feromona artificial y por una información calculada a priori de manera heurística. La regla probabilística para el caso del TSP es:

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in J_k} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta} \quad \text{si } j \in J_k \quad (1)$$

donde $p_{ij}^k(t)$ es la probabilidad con la que, en una iteración t del algoritmo, la hormiga k , situada actualmente en la ciudad i , elige a la ciudad j como próxima parada. J_k es el conjunto de ciudades no visitadas todavía por la hormiga k . $\tau_{ij}(t)$ es la cantidad de feromona acumulada sobre el arco (i, j) de la red en la iteración t . η_{ij} es la información heurística para la que, en el caso del TSP, se utiliza la inversa de la distancia existente entre las ciudades i y j . α y β son dos parámetros del algoritmo que hay que ajustar.

Cuando todas las hormigas han construido una solución, debe actualizarse la feromona en cada arco. La fórmula a seguir es:

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \sum_{k \in T^{\text{best}}} \frac{1}{L^{\text{best}}} \quad (2)$$

donde ρ es el coeficiente de evaporación de la feromona. T^{best} puede ser la mejor solución encontrada hasta el momento o bien la mejor solución encontrada en la iteración. L^{best} es la longitud de la solución T^{best} . Se obliga a que el nivel de feromona permanezca en un rango $[\tau_{\text{mín}}, \tau_{\text{máx}}]$; estos límites se imponen con el objetivo de evitar el estancamiento en la búsqueda de soluciones. Toda la feromona se inicializa con $\tau_{\text{máx}}$.

IV.3 Método de Campos Potenciales Artificiales

El segundo método, analizado en [?], muestra el método de potenciales artificiales el cual permite resolver el problema de la evasión de obstáculos y es muy bueno ya que se puede utilizar en aplicaciones en tiempo real.

Considere un espacio de trabajo bidimensional y un sistema de referencia fijo (X_1, X_2) en el que se mueve el vehículo articulado. Las coordenadas de un punto de referencia del vehículo están dadas por $q = (y_1, y_2)$, las coordenadas del obstáculo están dadas por $q_{\text{obs}} = (s_1, s_2)$ y las de la meta son $q_g = (y_{1g}, y_{2g})$. La fuerza resultante F aplicada al vehículo está dada por:

$$F(q) = F_{\text{att}}(q) + F_{\text{rep}}(q) \quad (3)$$

donde $F_{\text{att}}(q)$ es la fuerza de atracción producida por la meta y $F_{\text{rep}}(q)$ es la fuerza de repulsión inducida por el obstáculo.

La fuerza resultante proviene de un potencial artificial dado por:

$$F(q) = -\nabla U(q) = -\nabla U_{\text{att}}(q) - \nabla U_{\text{rep}}(q) \quad (4)$$

donde ∇ es el operador gradiente.

La fuerza de atracción, así como la de repulsión, están expresadas por el gradiente negativo del potencial como lo muestra la ecuación (4); la fuerza resultante guía al punto de referencia del vehículo hacia la meta evadiendo el obstáculo.

A lo largo del trabajo, $\|\cdot\| : \mathbb{R}^2 \rightarrow \mathbb{R}^+$ denota la norma euclideana. El potencial atractivo es el siguiente:

$$U_{\text{att}}(q) = \frac{1}{2} \xi \rho^2(q, q_g) \quad (5)$$

donde ξ es un factor de escala positivo, q_g es la coordenada de la meta, q es el vector de posición del punto de referencia del vehículo y $\rho(q, q_g) = \|q - q_g\|^2$ es una función definida positiva cuya primera derivada es continua, correspondiente al cuadrado de la distancia entre q y q_g . La función $\rho(q, q_g)$ posee un mínimo global igual a cero en $q = q_g$.

El potencial repulsivo debe ser definido también de manera que la función de potencial artificial $U(q)$ sea positiva, continua, diferenciable y su único mínimo sea cero en $q = q_g$. Con el fin de evitar que se desplace el mínimo global de la función de potencial atractivo, esta función debe tener una región de influencia que delimite al obstáculo. La expresión propuesta por Khatib es:

$$U_{\text{rep}}(q) = \begin{cases} \frac{1}{2} \eta \left(\frac{1}{\rho(q, q_{\text{obs}})} - \frac{1}{\rho_0} \right)^2 & \text{si } \rho(q, q_{\text{obs}}) \leq \rho_0 \\ 0 & \text{si } \rho(q, q_{\text{obs}}) > \rho_0 \end{cases} \quad (6)$$

donde η es un factor de escala positivo, q_{obs} es el punto donde se encuentra el obstáculo y $\rho(q, q_{\text{obs}}) = \|q - q_{\text{obs}}\|^2$ denota el cuadrado de la distancia entre el robot q y el obstáculo q_{obs} . La distancia ρ_0 es una constante que denota el radio de la zona de influencia del obstáculo.

Habiendo definido las expresiones de U_{att} y U_{rep} , las fuerzas inducidas pueden ser calculadas aplicando el gradiente de la ecuación (4):

$$F_{\text{att}}(q) = -\nabla U_{\text{att}}(q) = -\xi (q - q_g) \quad (7)$$

$$F_{\text{rep}}(q) = -\nabla U_{\text{rep}}(q) = \begin{cases} \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2} (q - q_{\text{obs}}) & \text{si } \rho \leq \rho_0 \\ 0 & \text{si } \rho > \rho_0 \end{cases} \quad (8)$$

En el caso de m obstáculos, el procedimiento anterior se puede generalizar considerando la fuerza asociada al i -ésimo obstáculo:

$$F_{\text{rep},i}(q) = -\nabla U_{\text{rep},i}(q) \quad (9)$$

La fuerza total resultante para el caso general queda de la siguiente manera:

$$F(q) = F_{\text{att}}(q) + \sum_{i=1}^m F_{\text{rep},i}(q) \quad (10)$$

El método clásico que se acaba de describir depende únicamente de la posición relativa entre el vehículo y los obstáculos.

Bibliografía

- [1] L. Barcos, V. M. Rodríguez, M. J. Álvarez y F. Robusté. «Algoritmo basado en la optimización mediante colonias de hormigas para la resolución del problema del transporte de carga desde varios orígenes a varios destinos». Departamento de Organización Industrial, Tecnum, Universidad de Navarra, España; E.T.S. Ingenieros de Caminos de Barcelona, Universidad Politécnica de Cataluña, España.
- [2] N. D. Rotstein y A. J. García. «Evasión de obstáculos con bajo costo computacional para un equipo de fútbol de robots». Laboratorio de Investigación y Desarrollo en Inteligencia Artificial (LIDIA), Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur.
- [3] E. Mariscal-García. «Planeación y seguimiento de trayectorias de robots móviles en una simulación de un ambiente real». Universidad Autónoma Indígena de México, Los Mochis, Sinaloa, México.
- [4] Innovatecno. <http://www.innovatecno.com/TextoRA.php>
- [5] Universidad de Guadalajara. Robótica. <http://proton.ucting.udg.mx/materias/robotica/r166/r45/r45.html>
- [6] T. A. Vidal Calleja. *Generalización del método de campos potenciales artificiales para un vehículo articulado*. Tesis de Maestría en Ciencias, Especialidad en Ingeniería Eléctrica. Instituto Politécnico Nacional.