

LA TECNOLOGIA COMO MEDIO DE AYUDA AL PLANETA

1st Juan Esteban Álzate Uribe
Universidad Eafit
Colombia
jealzateu@eafit.edu.co

2nd Santiago Arredondo Quintero
Universidad Eafit
Colombia
sarredondq@eafit.edu.co

Mauricio Toro
Universidad Eafit
Colombia
mtorobe@eafit.edu.co

ABSTRACTO

Hace casi tres décadas en el planeta ha estado disminuyendo dramáticamente la población de abejas en el mundo y esto a la mayoría de la gente la tienen sin cuidado.

Científicos como por ejemplo Albert Einstein

INTRODUCCIÓN

PROBLEMA

PROBLEMAS SIMILARES

PROBLEMA 1: Colisiones 2D en los videojuegos.

Lograr que cuando diferentes objetos que actúan en la pantalla se juntan lo suficiente entre sí parezca que están chocando y puedan reaccionar de alguna manera, y comprobar esto antes de que se realice el movimiento.

SOLUCIÓN: Los métodos más usados son:

- Mascaras de bits aplicadas a los sprites.
- Bounding Boxes, que consiste en el uso de círculos para representar las dimensiones del objeto en el algoritmo de detección.

Para representar formas humanas o cuya anchura difiera de la altura se pueden usar rectángulos.



- Particionamiento del espacio en zonas ocupadas y

Paste the appropriate copyright/license statement here. ACM now supports three different publication options:

- ACM copyright: ACM holds the copyright on the work. This is the historical approach.
- License: The author(s) retain copyright, but ACM receives an exclusive publication license.
- Open Access: The author(s) wish to pay for the work to be open access. The additional fee must be paid to ACM.

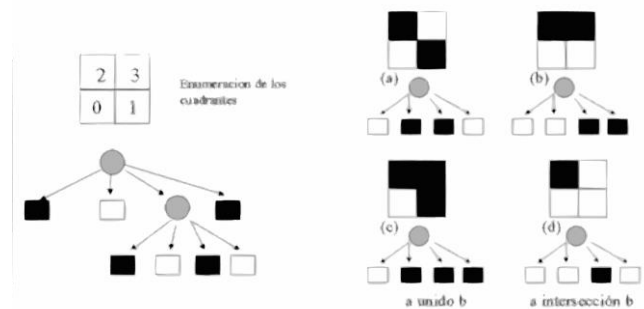
This text field is large enough to hold the appropriate release statement assuming it is single-spaced in TimesNewRoman 8 point font. Please do not change or modify the size of this text box.

Every submission will be assigned their own unique DOI string to be included here.

desocupadas.

Controlamos solamente la presencia o ausencia de una celda dentro de la malla, en este sentido solamente debemos determinar qué celdas se encuentran ocupadas y cuáles no.

Podemos usar los Quadrees (formato de representación 2D usado para codificar imágenes), el cual es derivado subdividiendo sucesivamente el plano 2D en ambas direcciones para obtener cuadrantes.



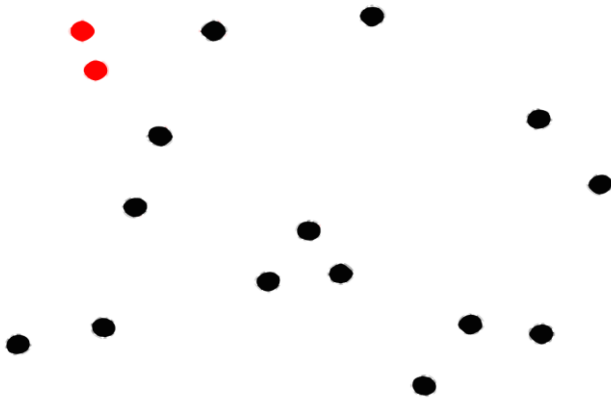
Otra opción es usar arboles BSP (Partición Binaria del Espacio).

PROBLEMA 2: Problemas del par de puntos más cercanos.

Es un problema clásico donde “Dados n puntos en un espacio métrico, se pide encontrar un par de puntos con la distancia más pequeña entre ellos”.

En el plano euclidiano fue de los primeros problemas tratados en el estudio sistemático de la complejidad computacional de algoritmos geométricos.

El par más cercano puede ser encontrado en tiempo proporcional a $O(n^2)$ mediante una búsqueda por fuerza bruta o puede ser resuelto en tiempo de $O(n \log n)$ usando un algoritmo recursivo.



PROBLEMA 3: Localización o modelado simultáneos.

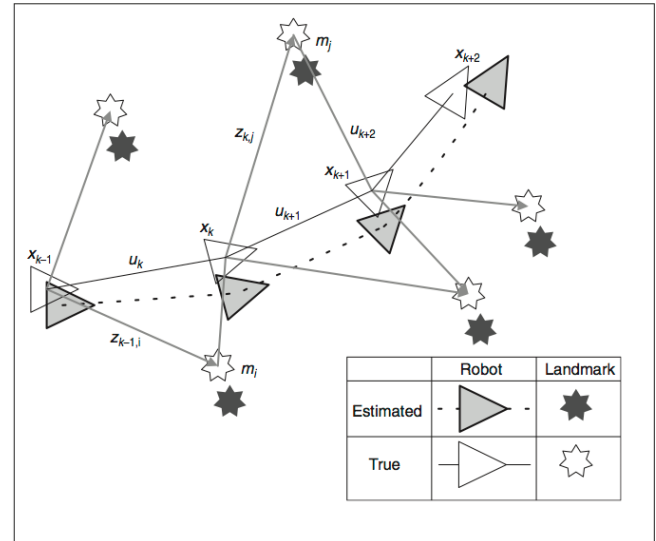
La localización y modelado simultáneos, mapeo y localización simultáneos o SLAM (del inglés simultaneous localization and mapping), es una técnica usada por robots y vehículos autónomos para construir un mapa de un entorno desconocido en el que se encuentra, a la vez que estima su trayectoria al desplazarse dentro de este entorno.

Esta técnica busca resolver los problemas que plantea el colocar un robot móvil en un entorno y posición desconocidos, y que él mismo sea capaz de construir incrementalmente un mapa consistente del entorno al tiempo que utiliza dicho mapa para determinar su propia localización.

Las soluciones que mejores resultados han obtenido a la hora de abordar el problema de SLAM son aquellas basadas en técnicas probabilísticas.

El Monocular SLAM, globalmente conocido como vSLAM (Visual SLAM), consiste en el empleo de una cámara para desarrollar un mapeo y localización simultánea. Este tipo de algoritmos, o técnicas, están compuestos por 5 módulos diferenciados. Estos son la inicialización, la localización, el mapeo, la re-localización y por último la optimización del mapeo. Cada uno de los diferentes algoritmos del diseño hasta la fecha de cada técnica para cada uno de estos 5 puntos. Algunos de los algoritmos más conocidos son PTAM, LSD SLAM y ORB SLAM entre

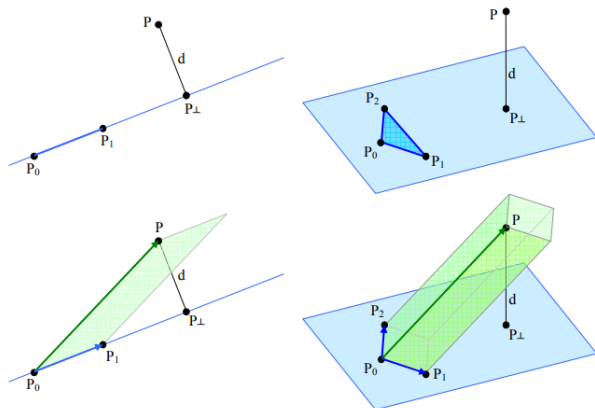
otros.



PROBLEMA 4: Intersecciones.

El cálculo de intersecciones en el espacio tiene varias finalidades en CG. La más simple es el clipping: El recorte de los objetos entre los límites del volumen visual, renderizando sólo la fracción visible de la escena. Las aplicaciones más complejas y masivas se dan al hacer ray-tracing y sus derivados, para utilizar modelos de iluminación y sombreado más realistas, allí hay que calcular una miríada de intersecciones entre rayos y objetos de la escena. También se utilizan para detección de colisiones en simulaciones y juegos o detección de interferencias entre objetos de CAD. Las aplicaciones en geometría computacional abarcan también los procesos de generación de mallas y la determinación de isocurvas e isosuperficies para visualización de datos.

Se pretende que los algoritmos empleados sean rápidos, precisos y robustos. La falta de robustez hace que los errores salten inmediatamente a la vista. Siempre es preferible sacrificar la precisión en aras de la robustez: que el código entregue un resultado útil, en cualquier caso, aún con un pequeño margen de error. El usuario no se da cuenta si el punto de intersección debería estar un píxel más aquí o allá, pero enseguida nota un píxel mal pintado o el ruido de colores del z-fighting; dando resultados visualmente inaceptables; o peor, un “cuelgue” del programa. El requerimiento de velocidad se vuelve obvio si se piensa que un procedimiento de ray-tracing, por ejemplo, debe analizar intersecciones entre millones de rayos y miles de objetos. La primera respuesta que hay que dar, y muy rápidamente, es si puede o no existir la intersección. Veremos las técnicas de partición del espacio que nos permiten acelerar estas consultas y realizar descartes masivos muy rápidamente. La velocidad de las placas gráficas se mide mediante la capacidad de procesar polígonos rápidamente, el polygon throughput está en el orden de los 100 millones de polígonos por segundo.



REFERENCIAS

1. <http://www.dccia.ua.es/dccia/inf/asignaturas/RG/2002/trabajos/colisiones.pdf>.
2. https://es.wikipedia.org/wiki/Problema_del_par_de_puntos_m%C3%A1s_cercanos.
3. https://es.wikipedia.org/wiki/Localizaci%C3%B3n_y_modelado_simult%C3%A1neos.
4. http://www.cimec.org.ar/~ncalvo/Intersecciones_y_Ordenamiento_doc.pdf