

ABEJAS EN ACCION: EVITANDO LA COLICIÓN DE ABEJAS

ROBÓTICAS COMO ALTERNATIVA PARA LA POLINIZACIÓN

Camila Barona
Universidad Eafit
Colombia
cbaronac@eafit.edu.co

Mariana Gómez
Universidad Eafit
Colombia
mgomezp10@eafit.edu.co

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

RESUMEN

Durante los últimos años, el mundo se ha topado con el problema de la desaparición y exterminio de las abejas que logran la polinización y dan múltiples ventajas a los cultivos y por ende al ser humano, por lo cual se han tomado medidas de solución por medio de la robótica y la programación, sin embargo, el problema de la colisión de estos animales robóticos ha sido preocupante en este campo pues sus posibles choques provocarían daños significativos.

1. INTRODUCCIÓN

Casi dos terceras partes de las plantas cultivadas que son predestinadas para el consumo de la humanidad dependen de la polinización de las abejas. Desde hace 15 años estos seres han disminuido considerablemente su número gracias al uso de pesticidas y ataque de parásitos; por esta razón, diferentes universidades de todo el mundo se han puesto en la labor de crear pequeñas abejas mecánicas que hagan la labor de estos seres vivos, combinando tecnologías innovadoras como robótica en miniatura y algoritmos de procesamiento.

2. PROBLEMA

Tratando la problemática hablada anteriormente, nos concentramos principalmente en el algoritmo de procesamiento; aquí nos centraremos en identificar que abejas robóticas se encuentran a 100 o menos metros de distancia de cualquier otra abeja, tomando como prioridad la eficiencia en tiempo de ejecución del algoritmo. Pero ¿para qué resolver este problema? Se trata con el fin de evitar la colisión de las abejas robóticas que de una u otra manera son una solución efectiva para enfrentar la desaparición de las abejas y su importante proceso en la polinización.

3. TRABAJOS RELACIONADOS

Para el actual informe se realizó una investigación acerca de diferentes estructuras de datos que promovieran a darle una solución efectiva al problema planteado.

3.1 Octree

Un octree o árbol octal es una estructura en "árbol" de datos en la cual cada nodo interno tiene exactamente 8 "hijos". Las estructuras octree se basan en la división recursiva de la caja envolvente de la figura a representar en ocho octantes que cubren totalmente el volumen. Si un octante está totalmente ocupado por el sólido se etiqueta como negro, si no lo está como blanco y si por el contrario está parcialmente ocupado como gris, en cuyo caso es necesario volver a dividirlo en 8 hijos que corresponden con la subdivisión.

La complejidad de este algoritmo es de $O(\log(n))$ y para obtener una solución completa y correcta se ha de tener en cuenta:

- Garantizar que la totalidad de la geometría contenida en el nodo sea usada al calcular su envolvente.
- Asegurar que la envolvente a nivel n este completamente contenida a nivel $n-1$
- Evitar huecos entre nodos adyacentes.

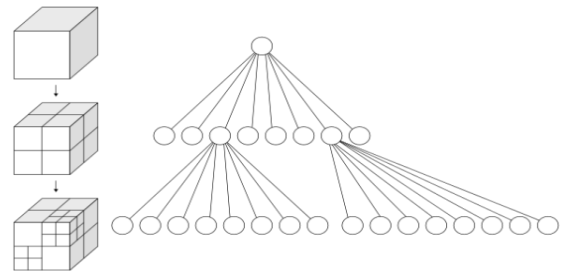


Figura 1

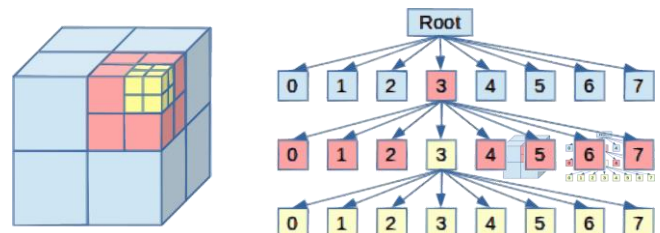


Figura 2

3.2 BSP

Binary space partitioning o Partición Binaria del Espacio (BSP) es un método o algoritmo utilizado para subdividir recursivamente un espacio en elementos convexos empleando hiperplanos. Esta subdivisión da lugar a una representación de la escena por medio de una estructura de datos del árbol conocida como árbol de BSP. Estos dos semiespacios se suelen denominar como positivo y negativo, estando el semiespacio positivo en la parte del plano hacia la que apunta su normal.

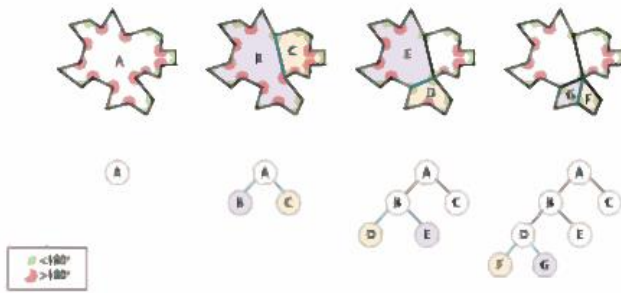


Figura 3

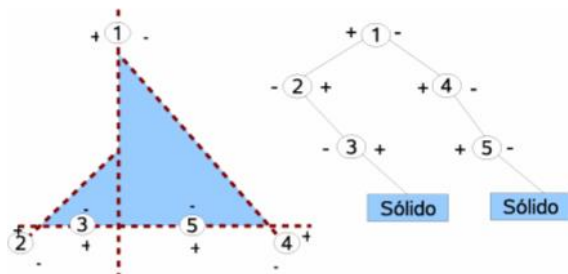


Figura 4

3.3 Quadtree

El término Quadtree, o árbol cuaternario, es un algoritmo que se utiliza para describir clases de estructuras de datos jerárquicas cuya propiedad común es que están basados en el principio de descomposición recursiva del espacio. En un QuadTree de puntos, el centro de una subdivisión está siempre en un punto. Al insertar un nuevo elemento, el espacio queda dividido en cuatro cuadrantes. Al repetir el proceso, el cuadrante se divide de nuevo en cuatro cuadrantes, y así sucesivamente.

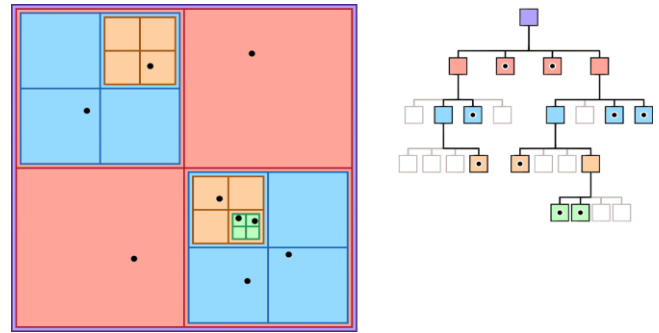


Figura 5

Este tipo de algoritmos son usados para diferentes problemas relacionados con el espacio de los objetos y en este caso el evitar colisiones de figuras, ya que sus diferentes niveles proporcionan diferentes resoluciones de detalles siendo útil de muchas maneras. Por ejemplo si se tiene un punto de datos bidimensional, x , y y entonces estarán inmersos exactamente en un nodo en cada nivel del árbol, por lo cual podrá utilizarse para realizar una búsqueda rápida de un conjunto de datos.

3.4 Arbol kd

U Árbol kd (abreviatura de árbol k-dimensional) es una estructura de datos de particionado del espacio que organiza los puntos en un Espacio euclídeo de k dimensiones.

Dado que hay muchas maneras posibles de elegir planos alineados a los ejes, hay muchas maneras de generar árboles kd. El sistema habitual es:

Conforme se desciende en el árbol, se emplean ciclos a través de los ejes para seleccionar los planos. (Por ejemplo, la raíz puede tener un plano alineado con el eje x , sus descendientes tendrían planos alineados con el y y los nietos de la raíz alineados con el z , y así sucesivamente)

En cada paso, el punto seleccionado para crear el plano de corte será la mediana de los puntos puestos en el árbol kd, lo que respeta sus coordenadas en el eje que está siendo usado.

- Construir un árbol kd estático a partir de n puntos es de $O(n \log n)$.
- Insertar un nuevo punto en un árbol kd balanceado es de $O(\log n)$.
- Eliminar un punto de un árbol kd balanceado es de $O(\log n)$.

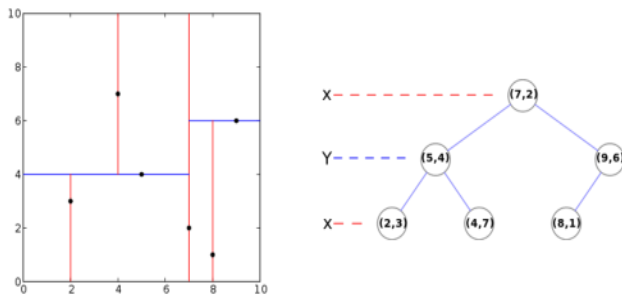


Figura 6

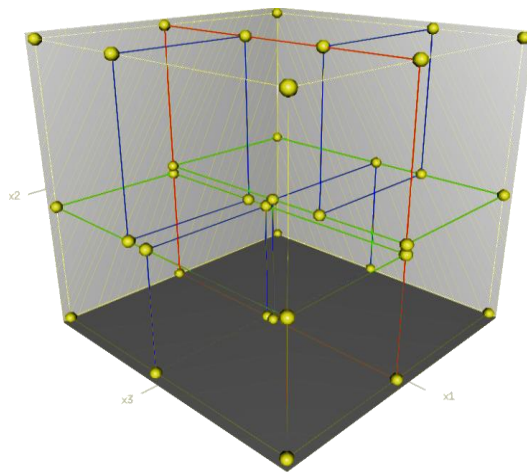


Figura 7

REFERENCIAS

1. Octree, 2017. Consultado el 25 de Agosto del 2018, Wikipedia, La enciclopedia libre. Recuperado de: <http://www.adobe.com/products/acrobat/>.
2. BSP, 2018. Consultado el 25 de agosto del 2018, Wikipedia, La enciclopedia libre. Recuperado de: https://es.wikipedia.org/wiki/Partici%C3%B3n_binaria_del_espacio
3. Quadtree, 2017. Consultado el 25 de agosto del 2018, Wikipedia, La enciclopedia libre. Recuperado de: <https://es.wikipedia.org/wiki/Quadtree>
4. Árbol kd, 2016. Consultado el 25 de agosto del 2018, Wikipedia, La enciclopedia libre. Recuperado de: https://es.wikipedia.org/wiki/%C3%81rbol_kd

5. Mike James. August 09, 2018. Quadrees and Octrees. Recuperado de: <https://www.i-programmer.info/programming/theory/1679-quadrees-and-octrees.html>
6. Adolfo Guzmán Arenas. Uso de árboles k-d como clasificadores supervisados y para substituir a sistemas expertos. Recuperado de: <http://www.cic.ipn.mx/aguzman/papers/89%20Arboles%20kd.htm>
7. Francisco Javier Melero Rus. Bp-Octree: Una estructura jerárquica de volúmenes envolventes. Retribuida noviembre 18. Tomado de la Universidad de Granada: <https://hera.ugr.es/tesisugr/17693895.pdf>
8. Programa utilizado para vectorizar imágenes: Inkscape. <https://inkscape.org/es>

Ilustraciones:

9. Figura 1: <https://es.wikipedia.org/wiki/Octree>
10. Figura 2: <https://geidav.wordpress.com/2014/07/18/advance-d-octrees-1-preliminaries-insertion-strategies-and-max-tree-depth/>
11. Figura 3: https://es.wikipedia.org/wiki/Partici%C3%B3n_binaria_del_espacio
12. Figura 4: <https://hera.ugr.es/tesisugr/17693895.pdf>
13. Figura 5: <https://developer.apple.com/documentation/gameplaykit/gkquadtree>
14. Figura 6: <https://www.ciencia-explicada.com/2015/10/atajos-para-mover-eficientemente-un-quadrotor-a-traves-del-grupo-especial-euclideo-se3.html>
15. Figura 7: https://es.wikipedia.org/wiki/%C3%81rbol_kd