ISSN 2071-081X
ISSN 2411-0035

DISEÑO DE UN ALGORITMO DE BÚSQUEDA INFORMADA MEDIANTE EL SIMULADOR ROBÓTICO STAGE

Marcelo Saavedra Alcoba, Luigi Pedro Enríquez Paz

Instituto de Investigaciones en Ciencia y Tecnología
Universidad La Salle Bolivia
msaavedra@doc.ulasalle.edu.bo, lpenriquez@est.ulasalle.edu.bo

Resumen

Se propone una nueva metodología para una búsqueda informada de objetos mediante un robot móvil. La metodología está basada en una combinación de un algoritmo de búsqueda en espiral y un algoritmo de búsqueda mediante un enfoque bayesiano que usa convoluciones entre "probabilidades de observación" y "máscaras de relaciones espaciales" para estimar un mapa probabilidad donde se pueda encontrar el objeto de búsqueda. Las relaciones espaciales entre objetos son representadas mediante el uso de máscaras circulares definidos como sumas ponderadas de relaciones espaciales básicas utilizando matrices de co-ocurrencia como pesos. La metodología está validada en un ambiente tipo oficina en la que existen cuatro clases de objetos comunes ("monitor", "teclado", "cpu" y "router") y se consideró cuatro relaciones espaciales básicas ("muy cerca", "cerca", "lejos" y "muy lejos"). Se realizaron un total de 500 experimentos que comparan 5 métodos de búsqueda utilizando el simulador robótico Stage. Los resultados muestran que el uso de la metodología propuesta tiene una tasa de detección mayor que un 70%, comparado a otros métodos como ser: Método de búsqueda en espiral, Método de búsqueda informada usando partículas entre otros.

Palabras Claves

Búsqueda informada, búsqueda en espiral, robótica móvil, inteligencia artificial.

Abstract

A novel methodology for robots executing informed object search is proposed. The methodology is based mainly on a heuristic spiral search algorithm and a Bayesian framework that uses convolutions between observation likelihoods and spatial relation masks for estimating the probability map of the object being search for. The spatial relationships between objects can be defined as weighted sums of basic spatial relations using co-occurrence matrices as weights. The methodology is validated in an office environment in which four object classes ("monitor," "keyboard," "system unit," and "router") and four basic spatial relations ("very near," "near," far," and "very far") are considered. A total of 500 experiments comparing five object search algorithms were realized by using the robotics simulator Stage. The results show that using the proposed methodology has a higher detection rate of 70%, compared to other methods such as: Method spiral search, method using particles informed, among others.

Key Words

Informed search, spiral search, mobile robot, artificial intelligence.

I. Introducción

La búsqueda de objetos en la vida cotidiana es una de las tareas más fundamentales de los seres humanos. Por esta razón, "buscar" es una de las tareas más básicas que se espera de un robot. Los problemas sobre búsqueda de rutas óptimas han sido muy estudiados en el área de la Inteligencia Artificial, pero buscar un objeto en un ambiente real es un trabajo más costoso ya que los objetos no siempre estarán al alcance del robot, y pueden estar ocluidos por otros objetos. El realizar una búsqueda en un ambiente real implica identificar y elegir un conjunto de acciones apropiadas de un conjunto muy grande de posibilidades. En contraste con los problemas clásicos de búsqueda de IA, en la búsqueda robótica se desea no volver a moverse entre lugares ya visitados (Burlington S, Dudek G., 1999). La situación más común al buscar un objeto en un ambiente interior es que el objeto pueda estar fuera del rango de visión del robot, otro ejemplo puede ocurrir en un ambiente interior como una oficina donde

muchos objetos cambiarán de posición frecuentemente y pueden quedar ocluidos por otros objetos.

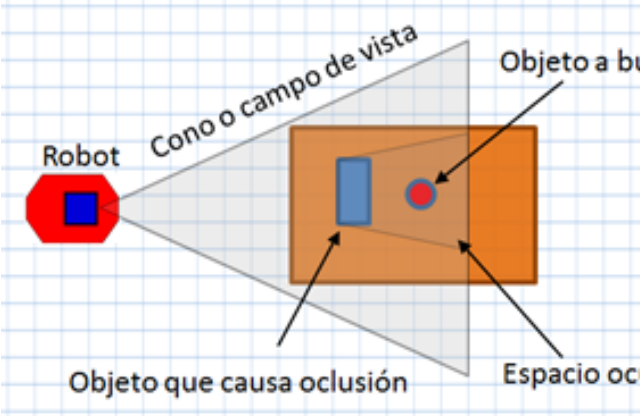


Figura 1. a) Situación de oclusión de objetos

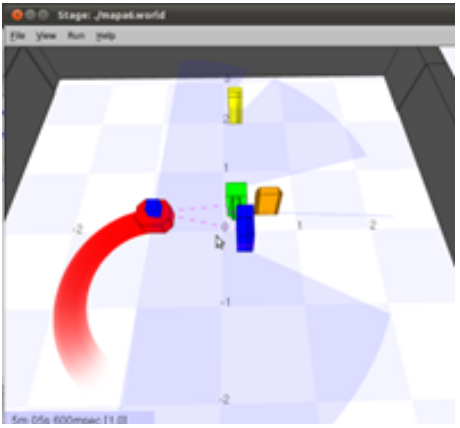


Figura 1. b) Simulator Player/Stage

En la Figura 1, se puede observar que la cámara del robot tiene un rango de detección representado mediante un **cono de detección** o un **punto de vista.** Una búsqueda directa explorando todo el ambiente bastaría para encontrar el objeto, sin embargo se está suponiendo que existe la misma probabilidad de encontrar el objeto en cualquier parte del ambiente. Desde esta perspectiva, se puede apreciar que el espacio de búsqueda de objetos

es enorme, y que la información incompleta de un ambiente podría hacer que la búsqueda de objetos demore mucho tiempo o que simplemente no se encuentre el objeto buscado. Es en esta parte donde la información de contexto puede jugar un papel importante a la hora de buscar objetos, ya que hay conjuntos de objetos dentro de un ambiente particular que tienden a repetirse ya que existe una relación espacial, por lo que es posible deducir la existencia de un objeto A, dado un objeto B. Por ejemplo, dentro de una oficina, se puede ver que el objeto "teclado" está a menudo muy cerca del objeto "monitor". Para un ser humano, la capacidad de deducir que el objeto A tiende a ser "cerca", "muy cerca" o "lejos" de un objeto B es una tarea trivial, ya que los seres humanos pueden aprender las relaciones espaciales entre los objetos mediante la observación de un gran número de configuraciones parecidas.

Para que un robot realice una búsqueda planificada necesita resolver dos sub-tareas, primero donde se realiza la siguiente vista y segundo donde se realiza el siguiente movimiento. Para la segunda sub-tarea se debe cumplir dos requisitos, el primero es que exista un espacio accesible en el ambiente y la segunda es que exista una alta probabilidad de encontrar el objeto. El "no planificar" estas dos tareas, implica examinar todas las posibles configuraciones de los movimientos y vistas del robot hasta encontrar el objeto buscado, esta estrategia resulta ineficiente por varias razones (Shubina H, Tsotsos J, 2010).

En el presente trabajo se llevó a cabo un análisis exhaustivo de las relaciones espaciales entre objetos que comparten un uso común. Para este objetivo, se creó una base de datos de coocurrencias espaciales donde los valores de distancias entre objetos están representados por variables lingüísticas como ser: "muy cerca", "cerca". "lejos" y "muy lejos". Decimos que dos objetos coocurren cuando aparecen juntos en varias imágenes manteniendo una particular relación espacial.

La importante contribución de este trabajo es el uso de las convoluciones de una manera unificada para el cálculo de la probabilidad de la presencia de un objeto a partir de detecciones positivas añadida a una exploración en espiral.

II. Trabajos Relacionados

La búsqueda de objetos en un ambiente real es compleja, ya que el objeto a buscar no siempre estará visible. En el trabajo de Baeza-Yates et al. (1993), titulado *"Searching in the Plane"* (Baeza R.A., Yates J.C. Culberson, Rawlins G.J, 1993), se estudian exhaustivamente varias técnicas sobre búsqueda robótica con el objetivo de encontrar las más optimas dentro de un ambiente plano. Los autores toman en cuenta que el ambiente es desconocido (quizás ilimitado) y además que existe un problema real respecto a la localización del robot. En los resultados obtenidos, los autores sugieren que una manera adecuada de realizar la búsqueda es a través de la aplicación continua de espirales logarítmicas. Posteriormente, Burlington y Dudek (1999) en su trabajo titulado *"Spiral search as an efficient mobile robotic search technique"* (Burlington S, Dudek G., 1999), utilizan este método calibrando valores en la iteración de la búsqueda en espiral para los ejemplos en ambientes dentro de polígonos que pueden ser representados como pasillos. Como una alternativa a este problema, en (Garvey, 1976) propuso la idea de la *búsqueda indirecta*, realizando la búsqueda de otro objeto intermedio que mantenga una relación espacial con el objeto buscado. materializó la idea de la búsqueda indirecta demostrando una mayor eficiencia tanto a nivel teórico como empírico (Wixon L., Ballard D., 1994). Pero el problema con la búsqueda indirecta es que la relación espacial entre el objeto buscado y el objeto intermedio no siempre existe. Además, la detección del objeto intermedio puede ser más difícil que la detección del objeto deseado. De hecho, Ye y Tsotsos han demostrado que la búsqueda de un objeto arbitrario en el espacio 3D es NP-completo (Ye Y., Tsotsos J.K., 1999). En el trabajo de Kollar et al.(1999), se realiza la búsqueda de un objeto en un ambiente conocido de acuerdo al contexto objeto-objeto y objeto-escena (Kollar T., Roy N., 2009), ellos obtienen la coocurrencia de la existencia de objetos en imágenes bidimensionales para obtener una estadística de las correlaciones entre categorías de objetos y entre objetos con etiquetas de lugar (etiquetas semánticas como "cocina"). Estas imágenes fueron tomadas del sitio web de *Flickr,* y no consideran la información de las distancias existentes entre objetos. En el trabajo de *Viswanathan* et. al., se propone un enfoque utilizando los recursos existentes como un conocimiento de sentido común para el aprendizaje automático de relaciones espaciales de los objetos (Viswanathan P., Meger

D., Southey T. Little J y Freeman W., 2009). Ellos usan imágenes marcadas desde la base de datos *LabelMe*, diseñado por Russell et al. (Rusell B., Torralba A., y Freeman W., 2008) y entrenan un clasificador automático de lugares basado en la presencia de los objetos detectados para inferir la probabilidad de existencia de otros objetos y el tipo de lugar (por ejemplo, la cocina o la oficina).

En el trabajo de Loncomilla et al. en (Loncomilla P., Ruiz del Solar J. y Saavedra M., 2013), se presenta un marco bayesiano para una búsqueda informada usando convoluciones entre verosimilitudes de observaciones y máscaras de relación especial, presentado así una mejora al trabajo presentado en (Aydemir A., Sjöö K. y Jensfelt P., 2010). Se puede observar que la tasa de detección aumenta considerablemente usando este nuevo método.

III. Herramienta de simulación STAGE

El costo de un buen robot puede ser un fuerte impedimento y más aún en el caso de investigar con más de un robot. Este es uno de los motivos por los cuales se utilizan simuladores para probar los programas y comprobar los resultados obtenidos, acercándose lo más posible a los que podrían obtenerse con el robot real (Cañas J.M. Matellán V. y Montufar R., 2006). Cabe destacar que las condiciones del entorno simulado no son las mismas que las del entorno real, el cual presenta ruidos en las mediciones de los sensores, situaciones inesperadas, y otros aspectos que a veces no pueden ser reproducidos con fidelidad en la simulación. Los simuladores suelen ser utilizados para comprobar los programas que posteriormente serán implantados en el robot real, de una forma más metódica y exhaustiva, ahorrando en algunos casos bastante tiempo.

a) El proyecto *Player/Stage*

El proyecto *Player/Stage* fue fundado por *Brian Gerkey*, *Richard Vaughan y Andrew Howard* en el año 2000, basándose en software que habían desarrollado con *Kasper Stoy* y otros en los laboratorios de investigación robótica de la Universidad de California del Sur. El nombre del proyecto fue tomado de una célebre frase de *Shakespeare "All the world's a stage*,

And all the men and women merely players" (El mundo entero es un escenario, y los hombres y mujeres son simplemente actores) (Gerkey B.P., Vaughan R.T. and Howard A., 2003).

Este proyecto está compuesto por tres componentes bien diferenciados:

• **Player** es un servidor que permite controlar los dispositivos de un robot y obtener información de sus sensores. Es una capa software que abstrae los detalles del hardware del robot, independizándonos del mismo. Los algoritmos de control del robot funcionarán como clientes de Player (a través de sockets TCP/IP). Así es posible controlar el robot enviando mensajes que sigan el protocolo de comunicación de Player o llamando a funciones de las librerías de Player, que pueden abstraer los detalles de comunicación. La distribución actual de Player incluye librerías en lenguajes tan diversos como C++, Java, Python o Lisp.

• **Stage** es un simulador de robots móviles en 2D que se puede utilizar de manera conjunta con Player si se quiere hacer pruebas iniciales de nuestros algoritmos o en el caso de no disponer de un robot real.

• **Gazebo,** es un simulador de robots móviles en 3D.

b) Arquitectura del simulador *Player/Stage*

Player utiliza una estructura de cliente/servidor para pasar datos e instrucciones entre el código y el hardware del robot. *Player* es el servidor, y el dispositivo de hardware del robot está suscrito como un cliente que se comunica al servidor a través de un proxy.

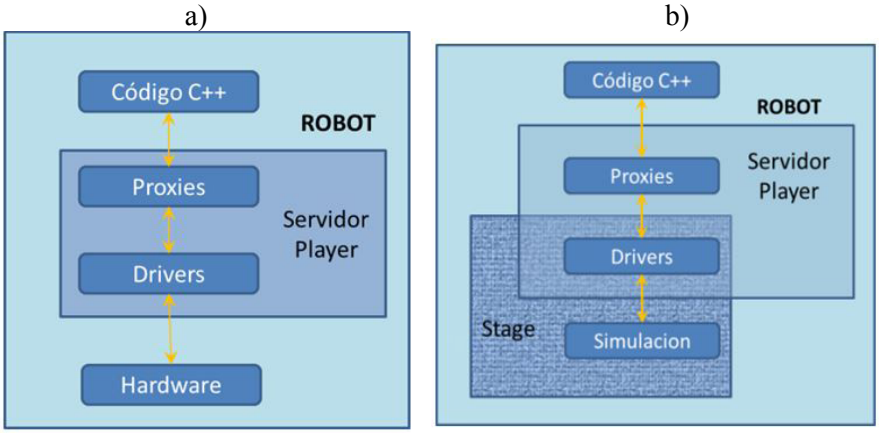
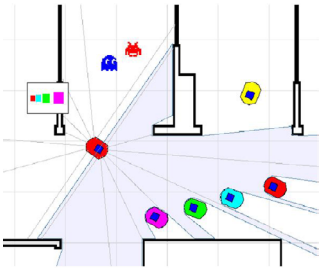


Figura 2. a) Estructura Cliente/Servidor de *Player,* b) Estructura Cliente/ Servidor de *Player* con el simulador *Stage*.

En la Figura 2 a) se muestra la estructura básica del *Player* cuando esta implementado en un robot y en 2 b), se muestra como es la estructura básica de *Player* con el simulador *Stage*.

c) Configuración de *Stage*

El componente *Stage* puede simular a uno o más robots móviles, sensores, objetos y el ambiente de trabajo para interactuar con el robot mediante un entorno bidimensional. *Stage* además proporciona la simulación de una amplia variedad de sensores y actuadores, tales como sonar, láser, dispositivos de detección de color por visión, odometría y otros. En la Figura 3 se puede observar una imagen de la versión *Stage* 3.2.2, simulando varios robots, obstáculos y el campo de visión por láser.



Stage está conformado a su vez por tres componentes:
- Ambiente de trabajo o entorno: descripción del mundo
- Modelo de robot
- Control a través de Player

Figura 3. Entorno del simulador *Stage* y detalle de sus componentes

Metodología

A continuación se hace un estudio de algunos métodos de búsqueda existentes, de acuerdo a esta teoría se plantea el algoritmo propuesto.

I. Métodos de búsqueda no informados

a) Método de búsqueda en espiral

La idea principal de realizar una búsqueda en espiral es examinar la mayor parte de un ambiente desconocido sin repetir el espacio recorrido. En la Figura 4 se puede ver una exploración en forma de espiral seguido del resultado del cono de detección, como se puede ver fácilmente gran parte del mapa está siendo explorado con este tipo de búsqueda.

a) Movimiento en espiral partiendo desde el centro del mapa, b) Cono de detección del robot mostrando el espacio explorado, c) Movimiento en espiral desde el simulador Player/Stage.

Figura 4. Ejemplo de búsqueda en espiral

Existe un problema en particular en este caso, y es que si el mapa es desconocido, el punto inicial es desconocido para el robot, por lo que podría empezar en una esquina. En caso de que la espiral comience en una esquina y la dirección del robot tiende a chocar con la pared, el efecto causado será bordear toda la pared (un seguidor de paredes). Esto puede ser un problema, por lo cual se presenta una solución que es la de cambiar la dirección de la espiral y decreciendo después de un cierto tiempo, con esta solución el mapa volverá a ser explorado con una tendencia de llegar al punto central como se puede ver en la Figura 5. En el caso que este cerca de un objeto o de una pared, se utilizará el mismo evasor de obstáculos y evasor de paredes usados en los otros métodos para que el robot no tenga colisiones.

Figura 5. Movimiento en espiral partiendo desde una esquina,
a) trayectoria del robot, b) cono de detección del robot

b) Método de búsqueda mediante maximización de probabilidad

Un mapa probabilidad $P(a_{i,j})$ para el objeto A se calcula mediante el uso de detecciones negativas de ese objeto, entonces el objeto A se busca mediante los de puntos de vista que maximizan la probabilidad de que contenga A. Este es el algoritmo de referencia utilizado en (Aydemir A., Sjöö K. y Jensfelt P., 2010), y no hay información desde los objetos secundarios que son utilizados.

II. Métodos de búsqueda informados

a) Método de búsqueda informada mediante convoluciones de información positiva

Para comprender este método primero se explicará los conceptos de mascara de coocurrencia y matriz de coocurrencia.

i. Máscaras de coocurrencias de objetos

Las máscaras de coocurrencias de objetos se utilizan para representar la relación espacial en forma de anillos, con esto se puede obtener un radio de distancias representando a las variables lingüísticas: “Muy Cerca”, “Cerca”, “Lejos” y “Muy Lejos”. Para obtener las máscaras y la matriz de coocurrencia, se usa el método descrito en (Loncomilla P., Ruiz del Solar J. y Saavedra M., 2013), donde ellos obtienen las máscaras considerando estadísticas de distancia entre objetos. Un ejemplo de máscara compuesta se muestra a lo largo de una rejilla con un tamaño de píxel de 0,1 [m] en la Figura 6.

a) b)

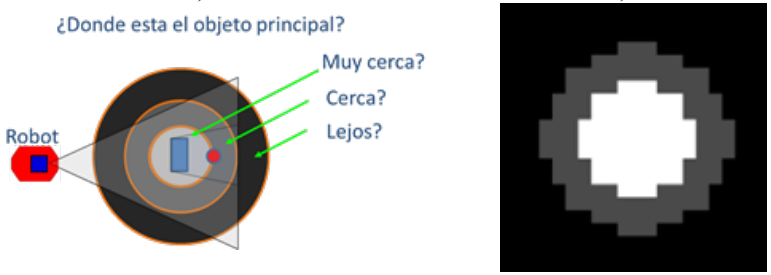


Figura 6. a) Representación gráfica de las relaciones espaciales, b) Máscara compuesta por la suma de las máscaras “muy cerca”, “cerca” y “lejos”, mostrado sobre una rejilla.

Una matriz de coocurrencia es el conjunto de valores de coocurrencia de dos o más objetos con una relación espacial particular. Las estadísticas de las distancias entre el objeto “Monitor” y los objetos “Teclado”, “Cpu” y “Router”, delimitan las relaciones espaciales básicas y se crean umbrales entre los objetos como un problema de clasificación de distancias.

ii. Creación de matriz de coocurrencias de objetos

Teniendo las distancias de umbrales obtenidos, en el trabajo de (Loncomilla P., Ruiz del Solar J. y Saavedra M., 2013) se toman estadísticas de coocurrencias entre objetos obteniendo la siguiente tabla:

Tabla 1. Probabilidades de existencia del objeto primario

	Objetos Secundarios		
Variables Lingüísticas	**Teclado**	**CPU**	**Router**
Muy Cerca	0,7731	0,1779	0,0606
Cerca	0,1428	0,4908	0,1515
Lejos	0,0462	0,2577	0,4848
Muy Lejos	0,0378	0,0736	0,3030

Como se puede ver en la Tabla 1, el valor para la variable "Muy Cerca" del objeto "Teclado" es de 0.7731, por lo tanto existe 77.30% de probabilidad que un "Monitor" este cerca al objeto "Teclado".

iii. Método mediante convoluciones

En el trabajo descrito en (Loncomilla P., Ruiz del Solar J. y Saavedra M., 2013), se realiza un modelo bayesiano para la búsqueda informada de objetos mediante convoluciones entre probabilidades de la observación y las máscaras de relación espacial. La metodología propuesta está diseñada para encontrar un objeto utilizando una búsqueda informada es decir, mediante el uso de la información acerca de otros objetos, que tienen una relación espacial con el objeto que se ha encontrado. En (Aydemir A., Sjöö K. y Jensfelt P., 2010), las relaciones espaciales son definidas en un sentido probabilístico. Definimos una relación espacial entre dos objetos como la distribución de probabilidad de la pose del primer objeto dada la pose conocida del segundo objeto:

$$Rel_{A,B}(\pi_A, \pi_B) = p(\pi_A \mid \pi_B) \quad (1)$$

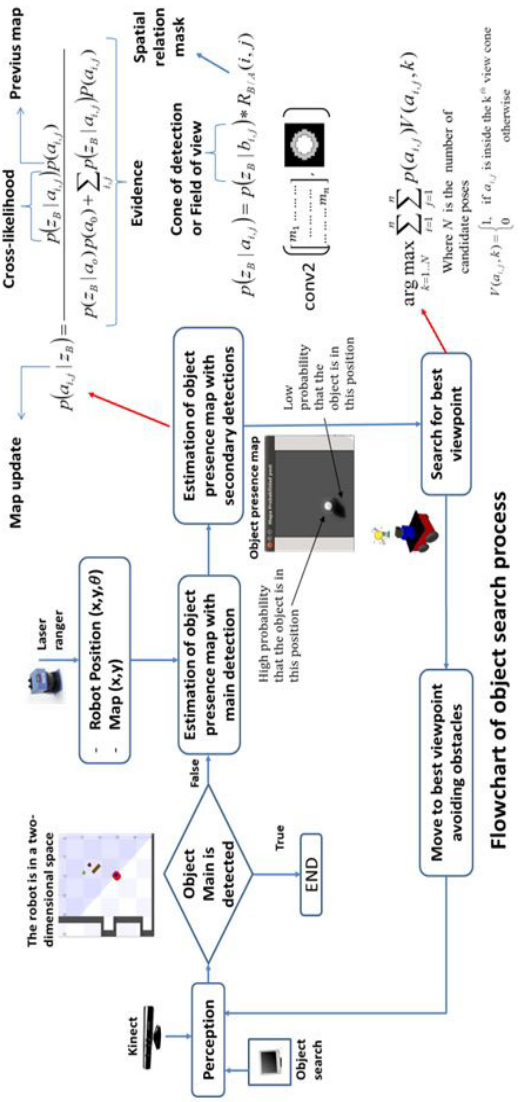


Figura 7. Diagrama de flujo y detalle del método bayesiano descrito en [10].

En la Figura 7 se muestra el diagrama de flujo general del trabajo realizado en (Loncomilla P., Ruiz del Solar J. y Saavedra M., 2013). Como se puede ver el robot está en un espacio de dos dimensiones con parámetros mediante coordenadas (x,y). El término $p(\alpha_{i,j})$ representa la probabilidad

de que el centro del objeto principal *A* se encuentra en la celda (*i,j*). Tanto las observaciones positivas y negativas $\mathbf{z}_A$ proporcionan información valiosa para el proceso de búsqueda de objetos, y se puede utilizar para calcular la actualización de probabilidad $p(\alpha_{i,j}|\mathbf{z}_A)$. La probabilidad $p(\alpha 0)$ es tratada como un caso especial, y representa la probabilidad de que el objeto está fuera de la región de búsqueda. Las detecciones positivas $\mathbf{z}_A$=*true* proporcionan la información acerca de los lugares en los que el objeto tiene una alta probabilidad de estar, por medio de la verosimilitud $p(z_A=true|\alpha_{i,j})$, que es definida sobre la celda (*i, j*). La verosimilitud tiene un alto valor sobre la celda donde se ha detectado el objeto y un valor bajo en las otras celdas. Las detecciones negativas $\mathbf{z}_A$=false proporcionan la información $p(\mathbf{z}_A=false|\alpha_{i,j})$ sobre las celdas donde los objetos tienen una baja probabilidad de estar, que son las celdas visibles desde el punto de vista actual que tienen una baja probabilidad de contener el objeto. El problema abordado en este trabajo es encontrar un objeto principal *A*, mediante movimientos apropiados del robot. Así se realiza una búsqueda que continuara hasta que el objeto principal *A* sea encontrado. En consecuencia, el proceso de búsqueda incluye sólo detecciones negativas del objeto principal *A*. Dos casos son considerados:

$$p(a_{i,j} \mid z_A = false) = \frac{p(z_A = false \mid a_{i,j})p(a_{i,j})}{p(a_o) + \sum_{i,j} p(z_A = false \mid a_{i,j})p(a_{i,j})} \qquad (2)$$

$$p(a_0 \mid z_A = false) = \frac{p(a_0)}{p(a_o) + \sum_{i,j} p(z_A = false \mid a_{i,j})p(a_{i,j})} \qquad (3)$$

El objeto secundario *B*, puede provocar detecciones positivas y negativas $\mathbf{z}_B$, que se puede utilizar para calcular la actualización de la probabilidad:

$$p(a_{i,j} \mid z_B) = \frac{p(z_B \mid a_{i,j})p(a_{i,j})}{p(z_B \mid a_o)p(a_0) + \sum_{i,j} p(z_B \mid a_{i,j})P(a_{i,j})} \qquad (4)$$

$$p(a_0 \mid z_B) = \frac{p(z_B \mid a_o)p(a_0)}{p(z_B \mid a_o)p(a_0) + \sum_{i,j} p(z_B \mid a_{i,j})P(a_{i,j})} \qquad (5)$$

Los términos $p(z_B \mid a_{i,j})$ y $p(z_B \mid a_0)$ serán llamados verosimilitudes-cruzadas, ya que relacionan la detección de un objeto secundario *B*, con la presencia del objeto principal *A* en el mapa. Estas probabilidades pueden ser derivadas considerando las probabilidades $p(b_{u,v})$ para la presencia de un objeto secundario *B*, en las localizaciones (*u,v*) de la grilla:

$$p(z_B \mid a_{i,j}) = \sum_u \sum_v p(z_B \mid b_{u,v}) p(b_{u,v} \mid a_{i,j}) \qquad (6)$$

$$p(z_B \mid a_0) = \sum_u \sum_v p(z_B \mid b_{u,v}) p(b_{u,v} \mid a_0) \qquad (7)$$

El término $p(b_{uv}|a_0)$ es considerada una constante en (*u,v*) cuya suma tiene un valor igual a uno, porque *B* supone que está en el mapa. El término $p(b_{u,v}|a_{i,j})$ corresponde a la relación espacial entre el objeto principal *A* en la posición (*i,j*) y un objeto secundario *B* en la posición (*u,v*). Mediante la sustitución de este término con la relación espacial $R_{B/A}$, no hay necesidad de almacenar un mapa para el objeto secundario; sólo el mapa para el objeto principal y las verosimilitudes de las detecciones del objeto secundario son necesarios:

$$p(z_B \mid a_{i,j}) = \sum_u \sum_v p(z_B \mid b_{u,v}) R_{B/A}(u-i, j-v) \qquad (8)$$

$$p(z_B \mid a_0) = \frac{1}{n_U n_V} \sum_u \sum_v p(z_B \mid b_{u,v}) \qquad (9)$$

Donde n_u n_v es el tamaño del mapa. La ecuación (8) puede ser implementado como una convolución en el espacio (*i,j*) entre una imagen de la verosimilitud y la máscara $R_{B/A}(i,j)$ que describe la relación espacial entre los objetos principales y secundarios, que se llamará *mascara de relación espacial:*

$$p(z_B \mid a_{i,j}) = p(z_B \mid b_{i,j}) * R_{B/A}(i,j) \qquad (10)$$

El sistema propuesto es muy versátil, ya que cualquier relación espacial puede ser representada mediante una máscara adecuada. Debe tenerse en cuenta que se pueden añadir objetos secundarios al sistema mediante la creación de máscaras de relación espacial adicionales. En caso de que estas relaciones sean encadenadas, como ejemplo de un objeto *A* está cerca de *B*, y el objeto *B* está cerca de *C*, entonces la máscara de la relación de cadena puede ser obtenida por convolución de las máscaras originales:

$$P(z_C \mid a_{i,j}) = P(z_C \mid b_{i,j}) * R_{B/A}(i,j) \qquad (11)$$

$$P(z_C \mid a_{i,j}) = P(z_C \mid c_{i,j}) * R_{C/B}(i,j) * R_{B/A}(i,j) \qquad (12)$$

$$\Rightarrow R_{C/A} = R_{C/B} * R_{B/A} \qquad (13)$$

Una ruta para buscar el objeto puede ser creado mediante la generación de puntos de vista óptimos en cada iteración. Los puntos de vista óptimos se generan a partir de un conjunto de *k* poses aleatorias alcanzables en un tiempo fijo, y se selecciona una que maximice la probabilidad de encontrar el objeto buscado en el área visible, como se muestra en (Aydemir A., Sjöö K. y Jensfelt P., 2010):

$$\arg\max_{k=1..N} \sum_{i=1}^{n} \sum_{j=1}^{n} p(a_{i,j}) V(a_{i,j}, k) \qquad (14)$$

Donde *N* es el número de poses candidatas plantea, y $V(a_{i,j}, k)$ es definida como:

$$V(a_{i,j}, k) = \begin{cases} 1, & \text{Si } a_{i,j} \text{ esta dentro del punto de vista k}^{th} \\ 0 & \text{En otro caso} \end{cases} \qquad (15)$$

Para mayor entendimiento de este método, se pueden revisar los trabajos en (Loncomilla P., Ruiz del Solar J. y Saavedra M., 2013) y (Aydemir A., Sjöö K. y Jensfelt P., 2010).

a) Método de búsqueda informada usando partículas

El algoritmo de *Aydemir* et al. (2010) es utilizado para la construcción de un mapa de probabilidad $P(a_{i,j})$ para el objeto A con detecciones negativas de ese objeto. Luego, el objeto *A* se busca mediante los puntos de vista que maximicen la probabilidad que contenga al objeto *A*. Cuando un objeto secundario es detectado, un conjunto de partículas se genera alrededor de la detección en el interior del cono de vista actual, y los que cumplen la relación espacial calculada a partir de la Tabla 1 se utilizan para seleccionar el punto de vista siguiente. Una relación espacial se considera cumplido cuando su valor actual es igual o mayor que la mitad de su valor máximo posible. P. 3-4

b) Método hibrido propuesto

Entre los métodos de búsqueda robótica estudiados, tanto de búsqueda informada como no informada, se escogen 2. El primero donde el robot no tiene información de otros objetos y solo busca el objeto principal recorriendo el ambiente en forma espiral, cabe destacar que este modelo es uno de los más óptimos para problemas de búsqueda sin información. El segundo es el método de búsqueda informada planteada en (Loncomilla P., Ruiz del Solar J. y Saavedra M., 2013) (método usando convoluciones) donde el robot utiliza la información de otros objetos que puedan tener relación con el objeto buscado. Es casi obvio que el tener información del ambiente supera a cualquier método sin información. En este trabajo se combinan ambos métodos, tanto el de búsqueda no informada como el de búsqueda informada para poder lograr una mejor búsqueda. El algoritmo de combinación propuesto se puede ver en la Figura 8.

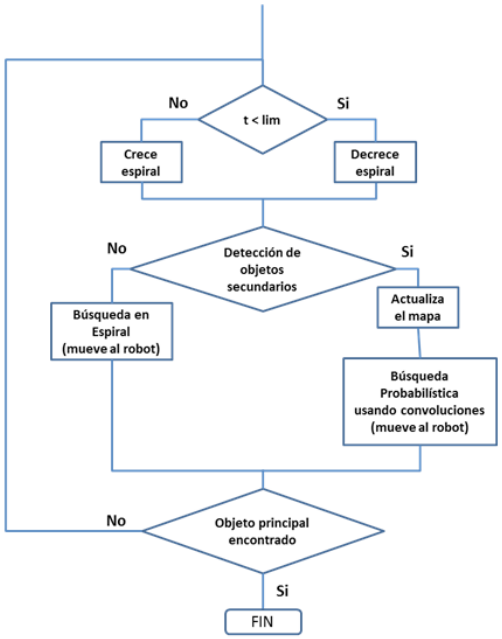


Figura 8. Diagrama de flujo del método propuesto

En el algoritmo mostrado anteriormente, t es el tiempo de cada acción del robot y lim es un límite de tiempo que define cuando crece la búsqueda en espiral y cuando decrece. Si existe una detección de objetos secundarios, el método usara la búsqueda probabilística. En caso contrario volverá a la búsqueda mediante la espiral.

Resultados

Se ejecutaron un total de 500 experimentos comparando cinco algoritmos explicados en la sección IV. Los resultados de los experimentos se muestran en la Tabla 2.

Tabla 2. Resultados

Algoritmo	% tasa de detección
a) Método de búsqueda en espiral	53
b) Método de búsqueda mediante maximización de probabilidad	31
c) Método de búsqueda informada mediante convoluciones de información positiva	65
d) Método de búsqueda informada usando partículas	42
e) Método hibrido propuesto (convoluciones y espiral)	71

A partir de los resultados de los experimentos, se puede notar que el método de búsqueda mediante maximización de probabilidad *(b)* junto con el método descrito en (Aydemir et al, 2010) que usa partículas *(d)* tiene un desempeño bajo en las pruebas realizadas. El algoritmo de búsqueda en espiral *(c)* para ser un método no informado tiene un alto desempeño en la búsqueda de objetos, vale mencionar que la búsqueda en espiral es muy eficiente en ambientes sin paredes, ya que en un ambiente con paredes este tipo de búsqueda puede ser más lento. Los algoritmos que utilizan convoluciones *("c" y "e")* para integrar la información acerca de los objetos secundarios en la distribución de probabilidad del objeto principal tienen el mejor rendimiento. El método hibrido propuesto *(e)* en este trabajo, es el de mayor rendimiento, ya que aumenta la tasa de detección hasta un 71%.

Conclusiones y recomendaciones

En este trabajo, fue propuesto y probado una nueva metodología para realizar una búsqueda informada de objetos. La metodología se basa en la integración de la información proporcionada por los objetos secundarios en la distribución de probabilidad del objeto principal y la búsqueda en espiral que es una de las más optimas en la búsqueda no informada. Cinco algoritmos de búsqueda de objetos se compararon mediante el uso de relaciones espaciales estimadas a partir de datos del mundo real mediante la realización de un total de 500 simulaciones en Player/Stage. Los resultados

muestran que la tasa de éxito del método hibrido propuesto incrementa la tasa de detección hasta un 71%. El trabajo futuro incluye el uso de conjuntos difusos para modelar las relaciones espaciales básicas y la realización de experimentos con un robot real para validar los resultados en el mundo real.

Referencias

Aydemir A., Sjöö K. y Jensfelt P. (2010). Objetct search on a mobile robot using relational spatial information. Proc. of the 11th Conference on Ittelligent Autonomous Systemas (IAS-11).p. 3-4.

Baeza R.A., Yates J.C. Culberson, Rawlins G.J. (1993). "Searching in the plane". Information and Computation 106 vol. 2, p. 234-252.

Burlington S, Dudek G. (1999). Spiral search as an efficient mobile robotic search technique. Centre for Intelligent Machines McGill University. p. 5-9.

Cañas J.M. Matellán V. y Montufar R. (2006). Programación de Robots Móviles. RIAI: Revista Iberoamericana de Automática e Informática Industrial Vol. 3, p. 99-110.

Garvey, T. (1976). Perceptual strategies for purposive vision. Technical report, SRI International vol. 117. p. 60-105.

Gerkey B.P., Vaughan R.T. and Howard A. (2003). The player/Stage project: tools for multi-robot and distributed sensor systems. Proceedings of the 11th International Conference on Advanced Robotics ICAR 2003. Coimbra-Portugal: ICAR. p. 1-6.

Kollar T., Roy N. (2009). Utilizing object-object and object-secene context when planning to find things presented at the in ICRA `09. Proceedings of the 2009 IEEE International Conference on Robotics and Automation. p. 2-7.

Loncomilla P., Ruiz del Solar J. y Saavedra M. (2013). A bayesian framework for informed search using convolutions between observation likelihoods and spatial relation masks. ICAR, Montevideo, Uruguay. p. 3-6.

Rusell B., Torralba A., y Freeman W. (2008). Labelme: A database and web-based tool for image annotation. International Jorunal of Computer Vision vol. 77, p. 157-173.

Shubina H, Tsotsos J. (2010). Visual search for an objetct in a 3d environment using a mobile robot. Computer Vision and Image Understanding 114, Vol. 5, p. 535-547.

Viswanathan P., Meger D., Southey T. Little J y Freeman W. (2009). Automated Spatial-Semantic Modeling with Applications to Place and informated Search. Canadian Conference on Computer anr Robot Vision. p. 3-6.

Wixon L., Ballard D. (1994). Using intermediate object to improve efficiency of visual search. Int. J. Computer Vision 18 vol. 18, p. 209-230.

Ye Y., Tsotsos J.K. (1999). Sensor Planning for 3D Objetc Search. Computer Vision and Image Unsderstanding vol. 73-2, p. 145-168.

Recibido: 16-06-2015
Aceptado: 20-08-2015