

Universidad de Oriente



Santiago de Cuba

Facultad de Ingeniería Eléctrica

*TESIS EN OPCIÓN AL GRADO
ACADÉMICO DE MASTER EN CIENCIAS*

*Generación de trayectorias para un robot móvil utilizando
visión basado en campos potenciales.*

Autor: Ing. José Alfredo Pérez Milanés

Tutora: MSc. Daily Milanés Herмосilla

2016

"Año 58 de la Revolución"

Dedicatoria

A mis padres, por su amor y apoyo.

Agradecimientos

A mi esposa Lissy y mi hijo por el amor y felicidad que me regalan a diario.

A mi tutora MsC. Daily Milanés Hermosilla, por su guía y preocupación.

A los familiares y amigos que me brindaron su apoyo y compañía.

Resumen

El presente trabajo aborda el tema de la generación de trayectorias en robots móviles que se auxilian de la visión para la obtención de la información de su entorno. Se realiza una caracterización de los robots móviles y de los principales métodos para la generación y seguimiento de trayectorias. Se diseña e implementa a escala de simulación un método para la generación de la trayectoria basado en campos de potencial, usando el Matlab. Para el seguimiento de la trayectoria generada se utiliza el algoritmo de persecución pura. Los resultados demuestran la validez del algoritmo al ser comparado con otro de mayor complejidad y obtener buenos resultados.

Abstract

This paper addresses the issue of generation paths in mobile robots that are aided vision to obtain information from their environment. A characterization of mobile robots and the main methods for the generation and path tracking is performed. It is designed and implemented to scale simulation method for generating the trajectory based on potential fields, using Matlab. To track the path generated pure pursuit algorithm is used. The results demonstrate the validity of the algorithm when compared to other more complex and get good results.

INDICE DE FIGURAS

<i>Figura 1.1. Categorías de los robots móviles según forma de locomoción.....</i>	<i>7</i>
<i>Figura 1.2. Robots móviles terrestres. Clasificación según forma de locomoción.</i>	<i>7</i>
<i>Figura 1.3. Configuración de los RMR.</i>	<i>16</i>
<i>Figura 1.4. Tipos de ruedas.</i>	<i>16</i>
<i>Figura 1.5. Esquema de un robot móvil de tracción diferencial.</i>	<i>16</i>
<i>Figura 1.6. Esquema del robot móvil de tracción sincrónica.....</i>	<i>17</i>
<i>Figura 1.7. Esquema de tracción tipo triciclo.....</i>	<i>17</i>
<i>Figura 1.8. Esquema de tracción tipo ackerman.</i>	<i>18</i>
<i>Figura 1.9. Robot móvil de tracción diferencial realizando una trayectoria.</i>	<i>19</i>
<i>Figura 1.10. Robot móvil tipo triciclo realizando una trayectoria.....</i>	<i>19</i>
<i>Figura 1.11. Esquema clásico de lazo de control en un robot móvil.</i>	<i>20</i>
<i>Figura 1.12. Esquema del sistema de seguimiento de trayectoria.....</i>	<i>22</i>
<i>Figura 1.13. Grafo de visibilidad en un entorno de dos obstáculos.</i>	<i>26</i>
<i>Figura 1.14. Retracción del espacio libre en un diagrama de Voronoi.....</i>	<i>26</i>
<i>Figura 1.15. Planificación con el espacio libre de obstáculos modelado mediante cadenas</i>	<i>27</i>
<i>Figura 1.16. Descomposición en celdas y grafo de conectividad</i>	<i>28</i>
<i>Figura 1.17. Descomposición trapezoidal del espacio libre.....</i>	<i>28</i>
<i>Figura 1.18. Grafo de conectividad de una descomposición trapezoidal.....</i>	<i>29</i>
<i>Figura 1.19. Ejemplo de Campo Potencial Artificial con 2 obstáculos.</i>	<i>30</i>
<i>Figura 1.20. Seguimiento de caminos mediante persecución pura.</i>	<i>35</i>
<i>Figura 1.21. Etapas de un Sistema de Visión por Computador</i>	<i>38</i>
<i>Figura 1.22. Lazo de control de un robot que incorpora visión.</i>	<i>39</i>
<i>Figura 1.23. Direcciones para entornos de a) 4 vecinos b) 8 vecinos.</i>	<i>40</i>
<i>Figura 1.24. Contorno de un objeto y su código de cadena 003311445667.....</i>	<i>41</i>
<i>Figura 1.25. Firmas generadas para una circunferencia y para un cuadrado.....</i>	<i>41</i>
<i>Figura 1.26. Influencia de la posición del centroide.....</i>	<i>42</i>
<i>Figura 1.27. Varias distancias para un mismo ángulo.</i>	<i>42</i>
<i>Figura 1.28. Aproximación poligonal.</i>	<i>43</i>
<i>Figura 1.29. Aproximación poligonal por partición sucesiva.</i>	<i>44</i>
<i>Figura 1.30. Aproximación poligonal para un contorno cerrado por partición sucesiva. ...</i>	<i>44</i>

Figura 2.1. Diagrama del robot móvil de configuración diferencial, con las velocidades asociadas a la rueda.....	49
Figura 2.2. Diagrama de las velocidades en el robot móvil.	50
Figura 2.4. Respuesta del modelo del robot ante velocidades iguales en el mismo sentido.	53
Figura 2.5. Respuesta del modelo del robot ante velocidades iguales en sentido contrario.	54
Figura 2.6. Respuesta del modelo ante velocidades diferentes en las ruedas y en el mismo sentido.	54
Figura 2.7. Respuesta del modelo ante velocidades diferentes en las ruedas y en el mismo sentido, con relaciones entre estas diferente.	55
Figura 2.9. Etapas de implementación del algoritmo.....	59
Figura 2.10. Dimensión de un pixel en el plano.....	60
Figura 2.12. Umbral seleccionado para la escala de grises.	61
Figura 2.13. Imagen en escala de grises a la que se le aplica el umbral.....	61
Figura 2.14. Función bwconvhull aplicada a un objeto.	62
Figura 2.15. Función imdilata aplicada a un objeto.	62
Figura 2.16. Supresión de mínimos locales utilizando bwconvhull y imdilata.	63
Figura 2.17. Segunda aplicación de la función bwconvhull.	64
Figura 2.18. Mínimo local formado entre el borde del plano y un obstáculo.	64
Figura 2.19. Fragmento de la matriz que representa el área de trabajo.	66
Figura 2.20. Potencial alrededor del punto de llegada.	67
Figura 2.21. Campo de potencial alrededor de un obstáculo.	68
Figura 2.22. Vista general del campo de potenciales.	69
Figura 2.23. Trayectoria generada utilizando el método de campos potenciales artificiales.	70
Figura 2.24. Trayectoria generada utilizando el método de campos potenciales artificiales.	70
Figura 2.25. Orden de encuesta de las celdas vecinas.	71
Figura 2.26. Representación de una recta en el plano cartesiano.....	73
Figura 2.27. Rangos en los que se divide la 8 vecindad.	74
Figura 2.28. Ejemplo de orientación calculada en el plano.	74
Figura 2.29. Trayectoria generada añadiendo el uso de la orientación en el programa....	75
Figura 2.30. Trayectoria generada añadiendo el uso de la orientación en el programa....	75

<i>Figura 2.31. Distancia de mira hacia adelante (Look Ahead)</i>	<i>77</i>
<i>Figura 2.32. Efectos de cambios en el valor de mira hacia adelante.....</i>	<i>77</i>
<i>Figura 2.33. Respuesta obtenida utilizando el método planteado en [9].....</i>	<i>79</i>
<i>Figura 2.34. Respuesta obtenida utilizando el método propuesto.</i>	<i>79</i>
<i>Figura 2.35. Respuesta obtenida utilizando el método planteado en [9].....</i>	<i>80</i>
<i>Figura 2.36. Respuesta obtenida utilizando el método propuesto.</i>	<i>80</i>
<i>Figura 2.37. Trayectoria obtenida disminuyendo el tamaño de la barrera.</i>	<i>81</i>
<i>Figura 2.38. Trayectoria obtenida modificando el punto de inicio de la trayectoria.....</i>	<i>81</i>
<i>Anexo 1. Representación del plano luego de ser tratado digitalmente.</i>	<i>88</i>
<i>Anexo 2. Campo de potenciales generado, con la trayectoria generada.....</i>	<i>89</i>
<i>Anexo 3. Especificaciones de la cámara utilizada para la adquisición de la imagen del plano.</i>	<i>90</i>

INDICE

INTRODUCCION.....	1
CAPÍTULO I. CARACTERIZACIÓN DE LOS ROBOTS MÓVILES Y ANÁLISIS DE LOS MÉTODOS PARA LA PLANIFICACIÓN Y SEGUIMIENTO DE TRAYECTORIAS.	6
1.1 Robots móviles. Clasificación.....	6
1.1.1 Evolución de los robots móviles.....	8
1.1.2 Elementos de los robots móviles.....	9
1.1.3 Sensores y actuadores en los RMR.....	12
1.1.4 Configuraciones cinemáticas de los RMR.....	15
1.1.5 Restricciones cinemáticas.....	18
1.2 Modelado cinemático y control de robots móviles.....	19
1.3 Planificación y seguimiento de trayectorias.....	22
1.3.1 Planificación basada en grafos.....	25
1.3.2 Planeación de trayectorias empleando campos potenciales.....	29
1.3.3 Algoritmos de búsqueda aleatoria.....	31
1.3.4 Métodos de Seguimiento.....	33
1.4 Navegación basada en visión.....	36
1.4.1 Visión artificial.....	37
1.4.2 Métodos utilizados para la extracción de características de objetos.....	40
1.5 Conclusiones del capítulo.....	45
CAPÍTULO 2. GENERACIÓN Y SEGUIMIENTO DE TRAYECTORIA PARA UN ROBOT MÓVIL BASADO EN CAMPOS POTENCIALES UTILIZANDO VISIÓN.....	46
2.1 Fundamentos teóricos de la propuesta.....	46
2.2 Cinemática de un robot móvil.....	47
2.2.1 Modelo cinemático de un robot móvil de tracción diferencial.....	49
2.3 Método de Campos Potenciales Artificiales.....	55
2.3.1 Implementación del algoritmo.....	59

2.3.2 Obtención y tratamiento de la imagen del plano.....	59
2.3.3 Preparación del campo de potenciales.	64
2.3.3.1 Inicialización de variables.	66
2.3.3.2 Creación del potencial alrededor del punto de llegada.	67
2.3.3.3 Creación del potencial alrededor de los obstáculos.	67
2.3.4 Generación de la trayectoria.	69
2.3.5 Utilizando la orientación para mejorar la trayectoria generada.	72
2.4 Seguidor de trayectorias.....	76
2.5 Análisis de resultados.....	78
2.6 Conclusiones del capítulo.....	82
CONCLUSIONES.....	83
RECOMENDACIONES.....	84
BIBLIOGRAFÍA.....	85
ANEXOS.....	88

INTRODUCCION

La robótica ha jugado un papel preponderante durante el desarrollo de la humanidad y su evolución ha ido siempre de la mano con la construcción de artefactos que materialicen el deseo de crear entes que faciliten el trabajo. En antiguas civilizaciones, como la griega, se hablaba de seres mecánicos con vida que eran movidos por mecanismos construidos con poleas y bombas hidráulicas. Sin embargo, el concepto de robot como tal, comenzó a hilvanarse en la civilización árabe, donde se les dio sentido a dichos mecanismos para confort del ser humano, es así como da inicio la sorprendente evolución de la robótica.

Aunque con el paso del tiempo fueron desarrolladas un gran número de figuras dotadas de partes móviles, no se tenía un concepto general de cómo definir las. En 1920 el escritor checo Capek, en su obra dramática Rossum's Universal Robots (R.U.R.), acuñó el término robot a partir de la palabra checa robota, que significa servidumbre o trabajo forzado, incluyendo en ese mismo trabajo, las tres leyes de la robótica.

1. Un robot no hará daño a un ser humano o, por inacción, permitir que un ser humano sufra daño.
2. Un robot debe obedecer las órdenes dadas por los seres humanos, excepto si estas órdenes entrasen en conflicto con la 1ª Ley.
3. Un robot debe proteger su propia existencia en la medida en que esta protección no entre en conflicto con la 1ª o la 2ª Ley.

La Robótica experimentó una rápida evolución a lo largo del siglo pasado, y hoy en día puede considerarse una disciplina científica plenamente asentada, de la que están surgiendo nuevas áreas de conocimiento.

Los robots en la actualidad, no sólo se limitan a asemejar a los seres humanos, sino que han tomado formas diversas para satisfacer ciertas necesidades de la mejor manera.

A consecuencia de la aparición de la computadora y de los cada vez mayores volúmenes de integración de circuitos, se pudieron desarrollar los primeros intentos de un verdadero robot en la década del cuarenta. En el Massachusetts Institute of Technology (MIT) en 1952, aparece la primera máquina de control numérico para automatizar algunas tareas industriales. Por su parte, la compañía Unimates, introdujo el primer robot industrial en la General Motors en 1961.

La robótica móvil es un área que permite extender las aplicaciones, en comparación a los robots fijos, puesto que se pueden desarrollar nuevas tecnologías con la implementación de robots no sedentarios, que puedan realizar navegación autónoma. A diferencia de los robots fijos, los móviles se encuentran en constante cambio de su posición y la orientación de acuerdo de su eje de coordenadas base (sistema global), por lo tanto deben estar en constante reconocimiento del entorno que lo rodea, para adaptarse y realizar un movimiento natural evadiendo los obstáculos que se le presenten. Para que un sistema de robótica móvil pueda trasladarse en un entorno debe contar con sensores que le permitan monitorear el ambiente constantemente.

Dentro del estudio de la robótica móvil y sus diferentes configuraciones es posible encontrar diversas aplicaciones que van desde el entretenimiento, medicina, milicia, sistemas de rescate, hasta la exploración en otros planetas.

En la mayoría de aplicaciones en las cuales se implementan robots móviles se enfrenta un problema similar, lograr que el robot se ubique en una posición específica o lograr que el robot se traslade de un punto específico a otro, logrando evadir los obstáculos o buscando la trayectoria más óptima. Para ello es necesario determinar una trayectoria adecuada para que un robot pueda desplazarse desde un punto inicial a uno objetivo. Dado que el espacio de trabajo suele tener obstáculos es importante garantizar una trayectoria libre de colisiones.

Una primera clasificación de los métodos para planear los movimientos de robots móviles, es dinámica o estática. En el caso dinámico, no se tiene información completa del espacio de trabajo y los obstáculos se consideran móviles. Para el

caso estático, se tiene conocimiento completo del espacio de trabajo y los obstáculos no se mueven. En [12, 15, 18, 24], se puede apreciar algunas aplicaciones para generar trayectorias de robots móviles ante la presencia de obstáculos estáticos y dinámicos.

Son muchas y muy variadas las técnicas utilizadas para la generación de trayectorias: Mapas de Carreteras (*Roadmap*), Grafos de Visibilidad, Diagramas de Voronoi, Descomposición Celular, Campos Potenciales, etc. No obstante, estas técnicas sólo permiten determinar un conjunto de puntos intermedios o configuraciones libres de colisión, pero no el camino que une dos o más configuraciones en su totalidad, es decir, como una trayectoria completa. Esto significa que dichos algoritmos sólo son capaces de unir los distintos puntos mediante segmentos de recta, de modo que el camino para ir de una configuración libre de colisión a la siguiente no queda claramente definido.

Por esto son de gran importancia los algoritmos de planificación de trayectorias, que son usados en diferentes aplicaciones en donde se hace necesario seguir un trayecto evitando obstáculos, los cuales pueden ser de diferente naturaleza, aumentando o disminuyendo la complejidad del algoritmo.

El resultado de la planificación se implementa en sistemas de navegación controlada. Una característica común de todos estos sistemas, es el uso de un controlador, el cual, en tiempo real permite que el sistema siga la trayectoria previamente obtenida. En el control para el seguimiento de trayectorias se hace necesario conocer la posición del vehículo en el plano XY.

El marco de referencia de este trabajo se centra en la implementación de un algoritmo para el control de la navegación de un robot móvil con modelo cinemático del sistema de propulsión, en este caso un robot de tracción diferencial, debido a que es un sistema simple y adecuado para la navegación en entornos cotidianos no muy exigentes. Este tipo de direccionamiento viene dado por la diferencia de velocidades de las ruedas laterales. Dos ruedas montadas en un único eje son propulsadas y controladas independientemente, proporcionando tracción y

direccionamiento. Además, permite cambiar la orientación del robot sin movimientos de traslación. Se utilizará el software MATLAB como entorno de cálculo y visualización en la programación y control del robot.

El **problema** consiste entonces en la insuficiencia de técnicas para la integración robot-visión en la planificación y seguimiento de trayectorias en robots móviles.

Para ello se define como **objeto de la investigación** los algoritmos para la generación y seguimiento de trayectorias en Robots móviles.

Como **objetivo de la investigación** el diseño de la planificación y seguimiento de trayectorias de un robot móvil de tracción diferencial utilizando el Matlab a partir de imágenes digitales.

Como **campo de acción** las técnicas de procesamiento de imágenes y métodos basados en campos potenciales para la navegación de robots móviles.

Se plantea la siguiente **hipótesis**: Si se diseñan métodos para la planificación y seguimiento de trayectorias de un robot móvil utilizando visión, se contaría con técnicas que permitan una mejor integración robot-visión en la planificación y seguimiento de trayectorias en robots móviles.

Para dar cumplimiento al objetivo de la investigación se trazaron las siguientes tareas:

1. Caracterizar desde el punto de vista histórico, gnoseológico y en la actualidad el uso de la visión en robots móviles.
2. Caracterizar desde el punto de vista histórico, gnoseológico y en la actualidad la planificación y seguimiento de trayectorias en robots móviles.
3. Modelar el robot de configuración diferencial, obtención del modelo cinemático.

4. Diseñar el método que permita la integración robot-visión para la planificación y seguimiento de trayectorias en robots móviles basado en campos potenciales.
5. Implementación a escala de simulación del método desarrollado utilizando el Matlab.
6. Comparación del método desarrollado con otro método basado en campos potenciales.

Técnicas y métodos empleados en la investigación:

1. Técnicas empíricas.
2. Análisis de fuentes documentales.
3. Observación.
4. Entrevistas a los expertos.
5. Método histórico – lógico.
6. Método de análisis – síntesis.

Aporte de la investigación: Desarrollo e implementación a escala de simulación de un algoritmo basado en métodos de campos potenciales utilizando visión para la planificación y seguimiento de trayectorias de un robot móvil con tracción diferencial con ayuda del Matlab.

La tesis se encuentra organizada de la siguiente forma: una introducción general en la que se exponen las principales motivaciones que llevaron a la realización de esta investigación y en la cual se encuentra, además, la fundamentación del diseño metodológico de la misma. Dos capítulos que constan de introducciones y conclusiones parciales para una mejor comprensión de los objetivos de los mismos, estos a su vez, se encuentran organizados por epígrafes. Finalmente se dan las conclusiones generales, recomendaciones y bibliografía utilizada para el desarrollo de esta investigación.

CAPÍTULO I. CARACTERIZACIÓN DE LOS ROBOTS MÓVILES Y ANÁLISIS DE LOS MÉTODOS PARA LA PLANIFICACIÓN Y SEGUIMIENTO DE TRAYECTORIAS.

Introducción

En este capítulo se abordan conceptos generales de la robótica móvil, la cual constituye el área de interés en el desarrollo de este trabajo. Primeramente, se muestran las diferentes configuraciones de robots móviles con ruedas, para luego analizar las diferentes configuraciones cinemáticas presentes en estos tipos de robots, y los principales elementos que los conforman. Adicionalmente se revisan los aspectos relacionados con la planeación de trayectorias y los principales métodos para lograrla. Por último, se mencionan los adelantos y métodos actuales utilizados en el seguimiento de trayectorias, en los cuales la visión constituye el elemento sensorial más potente utilizado para lograr satisfactoriamente este objetivo.

1.1 Robots móviles. Clasificación.

Existen diferentes clasificaciones de los robots, las cuales pueden ser por su geometría, método de control o según su función, dentro de esta última categoría se encuentran los robots manipuladores y robots móviles, los primeros enfocados a aplicaciones de manufactura y los segundos empleados en aplicaciones de exploración [4, 6, 8, 16].

Los robots móviles brindan la posibilidad de navegar en distintos terrenos y tienen aplicaciones como: exploración minera, exploración planetaria, misiones de búsqueda y rescate de personas, limpieza de desechos peligrosos, automatización de procesos, vigilancia, reconocimiento de terreno, y también son utilizados como plataformas móviles que incorporan un brazo manipulador.

En el caso de los robots móviles se presentan las siguientes categorías según su forma de locomoción, tal como se muestra en la figura 1.1:



Figura 1.1. Categorías de los robots móviles según forma de locomoción.

Los robots móviles terrestres se pueden clasificar por el tipo de locomoción utilizado, en general, los tres medios de movimiento son: por ruedas, por patas y orugas (figura 1.2) [24]. Cabe señalar que, aunque la locomoción por patas y orugas han sido ampliamente estudiadas, el mayor desarrollo se presenta en los Robots Móviles con Ruedas (RMR).

Dentro de los atributos más relevantes de los RMR, destacan su eficiencia en cuanto a energía en superficies lisas y firmes, a la vez que no causan desgaste en la superficie donde se mueven y requieren un número menor de partes y menos complejas, en comparación con los robots de patas y de orugas, lo que permite que su construcción sea más sencilla, además de poseer buena capacidad de carga. No obstante, están limitados a desplazarse sobre terrenos relativamente planos ya que tienden a presentar problemas si las diferencias en el terreno son mayores al radio de las ruedas.

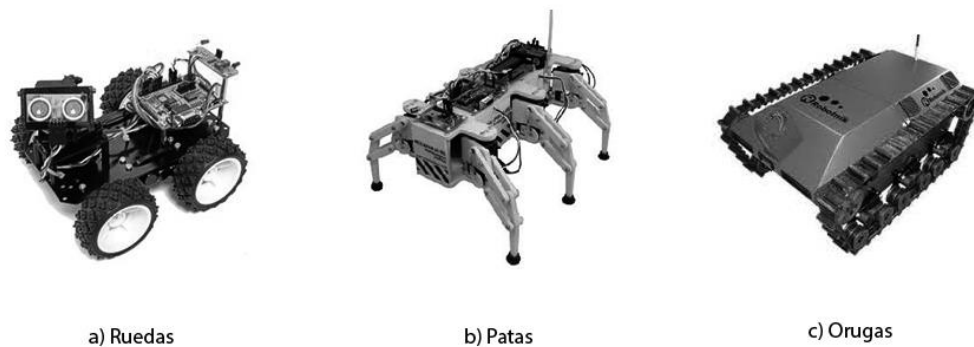


Figura 1.2. Robots móviles terrestres. Clasificación según forma de locomoción.

Son precisamente estos argumentos los que motivan el análisis de este tipo de robots, y surge la necesidad, en primera instancia, de tener una definición que satisfaga el contexto de los RMR.

De esta manera, se puede definir un robot móvil de ruedas como un sistema electromecánico controlado, que utiliza como locomoción ruedas de algún tipo, y que es capaz de trasladarse de forma autónoma a una meta preestablecida en un determinado espacio de trabajo [16].

1.1.1 Evolución de los robots móviles.

Con la llegada de nuevas tecnologías de planificación y razonamiento automático, de 1966 a 1972 se desarrolló en el Stanford Research Institute (SRI) el primer robot móvil llamado Shakey, que era una plataforma móvil independiente controlada por visión mediante una cámara y un detector táctil. A partir de ese momento, la investigación y diseño de robots móviles (que contaron con características muy diferentes entre ellos) creció de manera exponencial.

A principios de la década del setenta, el desarrollo de la robótica y la construcción de robots para disímiles funciones fue en aumento, en el Jet Propulsion Laboratory (JPL) se construyó el Lunar Rover diseñado particularmente para la exploración planetaria, así como modelos destinados para estudiar las diferentes características según sus tipologías, entre los cuales estaban el robot Raibert de una sola pata, desarrollado en el Massachusetts Institute of Technology (MIT), en el cual fue construido también un prototipo unicycle (una sola rueda).

Años más tarde, en 1994, el Instituto de robótica CMU desarrolló el robot Dante II, un sistema de seis patas. En 1996 también en el CMU, se desarrolló el robot Gyrover, un mecanismo ausente de ruedas y patas basado en el funcionamiento del giroscopio. Ese mismo año se desarrolló en el MIT el Spring Flamingo, un robot que emulaba el movimiento de un flamenco. Por su parte, la NASA en 1997 envió a Marte un robot móvil teleoperado llamado Sojourner rover, dedicado a enviar fotografías del entorno de dicho planeta. Ese mismo año, la empresa japonesa

HONDA, dio a conocer el robot P3, el primer humanoide capaz de imitar movimientos del cuerpo humano. Al siguiente año, se desarrolla en la universidad Waseda en Japón, el WABIAN R-III, un robot humanoide. En 1999 en el CMU, Zeglin propuso un nuevo diseño de robot con una pata llamado Bow Leg Hopper, un diseño que permite almacenar la energía potencial de la pata.

Desde 2004 hasta 2010 los robots teleoperados Spirit rover y Opportunity rover, exploraron la superficie del planeta Marte en busca de mantos acuíferos.

La compañía de robótica Boston Dynamics, propiedad de Google, presentó en febrero del 2015 a SPOT, un autómeta cuadrúpedo. Se trata de un robot todoterreno con infinidad de funciones. Una de las principales características que destaca de Spot es su capacidad para resistir, mantener o recuperar su postura frente a empujones o incluso patadas o golpes fuertes de forma simple, una tarea que representa un hito en la robótica moderna. Sube, baja, trepa incluso corre. No es fácil hacerle perder el equilibrio y puede levantar más peso de los 72 kilos que pesa él, personas u objetos.

Los robots aquí mencionados, son únicamente una porción de los tantos que se han diseñado, sin embargo, es posible notar que las aplicaciones de estos son vastas y que las mismas son ilimitadas debido al desarrollo cada vez más vertiginoso de la tecnología.

1.1.2 Elementos de los robots móviles.

La estructura de un robot es el conjunto de elementos mecánicos que le dan forma y soportan los demás elementos que lo componen. Los robots pueden mostrar muchos tipos de estructuras dependiendo del fin para el que estén diseñados: brazo robótico articulado, vehículo para desplazamiento de cargas, paneles solares que siguen automáticamente al Sol, etc.

Pero todos ellos están formados fundamentalmente por los siguientes componentes: sensores, sistemas de control, actuadores, herramientas y elementos que suministran energía.

SENSORES

Los sensores son los encargados de recoger la información del entorno y enviarla al sistema de control para su procesamiento. Se pueden clasificar en dos tipos dependiendo de la función que realicen.

Sensores externos: Son los que sirven para tomar datos del entorno del robot, como, por ejemplo, detectar objetos (finales de carrera, sensores infrarrojos y ultrasonidos), niveles de iluminación (LDR), temperatura (NTC o PTC), etc.

Sensores internos: Sirven para controlar el propio funcionamiento del robot, como velocidad de los motores, posición de elementos móviles, la fuerza ejercida, etc.

SISTEMAS DE CONTROL

Los sistemas de control son los encargados de analizar la información que les mandan los sensores, tomar decisiones y dar órdenes para que las realicen los actuadores.

Estos sistemas de control pueden realizarse de dos formas:

Mediante un circuito electrónico que puede ser programable: Este sistema de control permite construir pequeños robots móviles sin necesidad de cables de conexión con un ordenador.

Mediante ordenador: Este es más utilizado en máquinas que no realizan desplazamientos, ya que la conexión por cable con el ordenador dificultaría su movilidad.

ACTUADORES

Son los elementos encargados de promocionar el movimiento a las articulaciones del robot. Dependiendo de la forma en la que reciban la energía para realizar el movimiento, se puede hacer la siguiente clasificación:

Actuadores eléctricos: Son los motores eléctricos que utilizan la energía eléctrica para el que el robot ejecute sus movimientos. Se emplean para robots de tamaño mediano, que no requieren de tanta velocidad ni potencia como los robots diseñados para funcionar con impulsión hidráulica.

- Los motores eléctricos de corriente continua se utilizan para proporcionar movimientos giratorios en los que no se requiere mucha precisión.
- Los motores paso a paso permiten controlar de forma precisa el ángulo de giro del motor, haciendo que el motor se coloque en una posición determinada. Para el control de estos motores se requiere un circuito electrónico de control o un ordenador.

Actuadores hidráulicos y neumáticos: Son los solenoides, formados por una bobina de hilo de cobre esmaltado, en cuyo interior se mueve un núcleo de hierro cuando se hace circular una corriente por el bobinado. Con los solenoides se consiguen movimientos lineales cortos, y se pueden utilizar para enclavar o liberar determinadas partes móviles de la máquina.

PINZAS Y HERRAMIENTAS

Los robots manipuladores suelen llevar algún dispositivo que se une a la muñeca del robot para realizar una tarea determinada. Se pueden dividir en dos tipos: las pinzas, diseñadas para la manipulación, transporte y unión de objetos, y las herramientas, cada una de ellas diseñada para una función específica.

SUMINISTRO DE ENERGIA

Los robots son máquinas electromagnéticas; por lo tanto, todas ellas necesitan un suministro de energía eléctrica, al menos, para el sistema de control. La energía eléctrica también proporcionará alimentación para los motores eléctricos y electromagnéticos.

Si los robots constan de motores de explosión, será necesario disponer de un suministro de gasolina o gasoil. Cuando el robot incluya actuadores hidráulicos o neumáticos, será preciso disponer también de un suministro de aceite o de aire, respectivamente.

1.1.3 Sensores y actuadores en los RMR.

Un robot es, por definición, una máquina capaz de interactuar con su entorno. Si es móvil, a menos que se mueva en un espacio absolutamente acotado y preparado para él, deberá ser capaz de adaptar sus movimientos y sus acciones de interacción en base a las características físicas de los ambientes con los que se encuentre y los objetos que hay en ellos.

Para lograr esta capacidad de adaptación, lo primero que necesitan los robots es tener conocimiento del entorno. Para conocer el medio, los seres vivos disponemos de un sistema sensorial. Los robots no pueden ser menos: deben poseer sensores que les permitan saber dónde están, cómo es el lugar en el que están, a qué condiciones físicas se enfrentan, dónde están los objetos con los que deben interactuar, sus parámetros físicos, etc.

Para esto se utilizan diversos tipos de sensores (o captadores), con un grado de complejidad y sofisticación que varía desde algunos bastante simples a otros con altos niveles de sofisticación de hardware y más aún de complejidad de programación.

Existe una amplia variedad de dispositivos diseñados para percibir la información externa de una magnitud física y transformarla en un valor electrónico que sea

posible introducir al circuito de control, de modo que el robot sea capaz de cuantificarla y reaccionar en consecuencia.

Un sensor consta de algún elemento sensible a una magnitud física -como por ejemplo la intensidad o color de la luz, temperatura, presión, magnetismo, humedad- y debe ser capaz, por sus propias características, o por medio de dispositivos intermedios, de transformar esa magnitud física en un cambio eléctrico que se pueda alimentar en un circuito que la utilice directamente, o si no en una etapa previa que la condicione (amplificando, filtrando, etc.), para que finalmente se la pueda utilizar para el control del robot.

Dentro de las magnitudes físicas que son necesarias medir para que un robot tenga algún conocimiento del entorno:

- Luz (con su gama de espectro: visible, infrarroja, ultravioleta)
- Sonido y ultrasonido
- Gravedad (inclinación, posición)
- Temperatura
- Humedad
- Presión y/o fuerza
- Velocidad
- Magnetismo
- Ubicación
- Proximidad
- Distancia

Relativo a los actuadores utilizados para dotar de movimiento a los RMR, es común que se utilicen motores. Existe una gama bastante amplia dependiendo de su empleo, los más utilizados en la robótica móvil son los de corriente directa (CD), por el hecho de que su modelo matemático es lineal, lo que facilita su control, específicamente los de imán permanente debido a que el voltaje de control es aplicado al circuito de armadura y el circuito de campo es excitado de manera independiente.

Los motores de CD de imán permanente, se presentan en dos tipos: con escobillas y sin escobillas. Ambos tipos brindan ventajas semejantes, sin embargo, los motores sin escobillas tienen algunas ventajas significativas sobre los motores con escobillas, como, por ejemplo:

- a) al no contar con escobillas, no se requiere el reemplazo de éstas ni mantenimiento por residuos originados de las mismas.
- b) no presentan chispas que las escobillas generan, de esta forma se pueden considerar más seguros en ambientes con vapores o líquidos inflamables.
- c) la interferencia causada por la conmutación mecánica de las escobillas se minimiza considerablemente mediante una conmutación electrónica.
- d) los motores sin escobillas alcanzan velocidades de hasta 50,000 rpm comparadas con las 5,000 rpm aproximadamente, máximas de los motores con escobillas.

A pesar de que estas ventajas parecieran tender la balanza a favor de los motores sin escobillas, existen desventajas cruciales que pueden cambiar la tendencia:

- a) en los motores sin escobillas no se puede invertir el sentido de giro cambiando la polaridad de sus terminales, esto agrega complejidad y costo a su manejo.
- b) los motores sin escobillas son más caros.
- c) se requiere un sistema adicional para la conmutación electrónica.
- d) el controlador de movimiento para un motor sin escobillas es más costoso y complejo que el de su equivalente con escobillas. Al igual que en el arreglo cinemático, cuando se modela un motor de CD se asumen algunas consideraciones, de esta forma se establece que la única fricción presente es la viscosa, aunque en la práctica se involucran otros tipos de fricción no lineales, sin embargo, la suposición es válida al elegir un motor cuyo efecto de las fricciones no lineales sea muy pequeño.

Estos elementos hacen de los motores de CD sin escobilla sean utilizables principalmente en aplicaciones en las cuales el sentido de giro nunca cambie y en las que se necesite el menor ruido posible. Dejando a los motores de CD con

escobillas el resto de las aplicaciones, principalmente como actuadores en plataformas móviles.

A este grupo se suman también los motores de paso, que son gobernados por impulsos procedentes de sistemas lógicos, presentan las ventajas de contar con alta precisión y repetitividad en cuanto al posicionamiento. Entre sus principales aplicaciones se destacan como motor de frecuencia variable, motor de corriente continua sin escobillas, servomotores y motores controlados digitalmente, ampliamente usados en temas de robótica.

1.1.4 Configuraciones cinemáticas de los RMR.

Existen diferentes configuraciones cinemáticas para los RMR, las que dependen principalmente de la aplicación hacia dónde va enfocado, no obstante, de manera general se tienen las siguientes: Ackerman, triciclo clásico, tracción diferencial, skid steer, síncrona y tracción omnidireccional [7, 8] (ver figura 1.3). Dependiendo de la configuración cinemática, los RMR utilizan cuatro tipos de ruedas para su locomoción, las cuales pueden ser: convencionales, tipo castor, ruedas de bolas y omnidireccionales, observadas en la figura 1.4.

En el marco de las configuraciones cinemáticas posibles y las ruedas que estas utilizan, los RMR documentados en la literatura utilizan comúnmente la configuración de tracción diferencial, (figura 1.3c), donde se utilizan ruedas convencionales (figura 1.4a), como ruedas motrices y una o dos ruedas tipo castor, de bola, u omnidireccionales, (figuras 1.4b, 1.4c, 1.4d), respectivamente, para proveer de estabilidad al móvil.

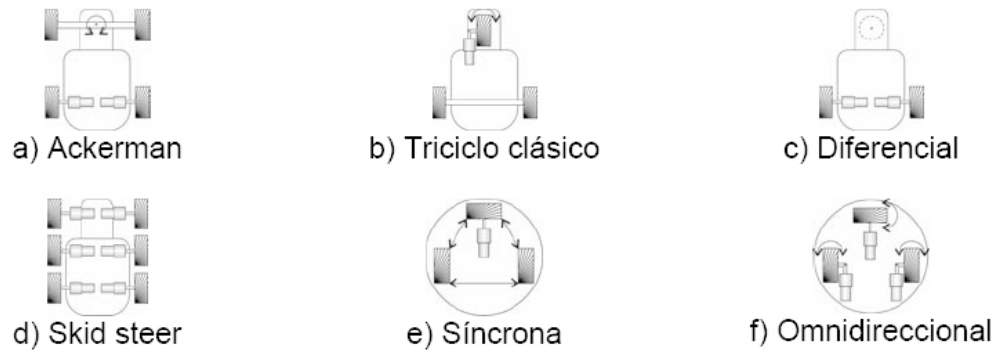


Figura 1.3. Configuración de los RMR.



Figura 1.4. Tipos de ruedas.

TRACCIÓN DIFERENCIAL

Es uno de los esquemas más sencillos, consiste de dos ruedas en un eje común, donde cada rueda se controla independientemente. Este tipo de configuración permite realizar desplazamientos en forma recta, de arco o girar sobre su propio eje. En esta configuración se puede utilizar una o dos ruedas adicionales para mantener el balance. En la figura 1.5 se observa el esquema de este tipo de tracción.

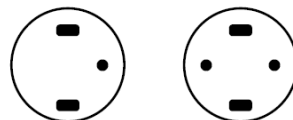


Figura 1.5. Esquema de un robot móvil de tracción diferencial.

TRACCIÓN SINCRÓNICA

En esta configuración todas las ruedas se mueven en forma síncrona para dar vuelta y avanzar, las ruedas están dispuestas de forma que siempre apuntan en la misma dirección. Para dar vuelta las ruedas giran sobre su eje vertical. Cuando giran las ruedas la dirección del chasis se mantiene. En ciertas aplicaciones es necesario un mecanismo adicional para mantener el frente del chasis en la misma dirección de las ruedas. En la figura 1.6 se observa el esquema de este tipo de tracción.

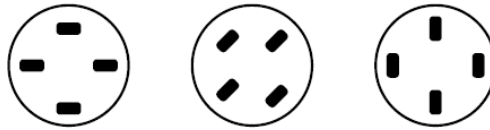


Figura 1.6. Esquema del robot móvil de tracción síncrona.

TRACCIÓN TIPO TRICICLO

Este esquema consiste en dos ruedas encargadas de suministrarle tracción al móvil mientras se tiene una rueda para la dirección que normalmente no tiene tracción. Esta configuración presenta buena estabilidad y simplicidad mecánica, facilitando la descripción de trayectorias rectas. En la figura 1.7 se observa la representación de este tipo de tracción

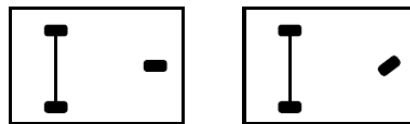


Figura 1.7. Esquema de tracción tipo triciclo.

TIPO ACKERMAN

Es similar al tipo triciclo, consiste en dos ruedas de tracción y dos ruedas para dirección. Aunque tiene mayor complejidad mecánica que el triciclo por el acoplamiento entre las dos ruedas de dirección, presenta buena estabilidad y facilita

describir trayectorias rectas. En la figura 1.8 se puede apreciar el esquema de este tipo de tracción.

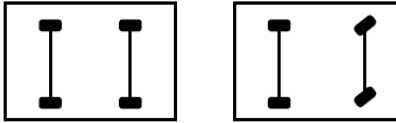


Figura 1.8. Esquema de tracción tipo ackerman.

1.1.5 Restricciones cinemáticas

Los robots móviles pueden clasificarse en holonómicos o no holonómicos dependiendo del número de grados de libertad efectivos del robot, los cuales están condicionados al desacople que se presenta entre los movimientos dados por los actuadores. Un robot que se desplaza sobre una superficie plana presenta tres grados de libertad (x , y , θ) dos de posición y uno de orientación. Idealmente sin importar la posición inicial se debería posicionar el robot en cualquier punto y orientación en el espacio de trabajo, sin embargo, si la orientación y la posición están acopladas el robot para realizar un giro debe realizar movimientos hacia adelante y hacia atrás por lo cual no puede ir directamente a la posición y orientación deseada.

El robot de tracción diferencial y robot síncrono son ejemplos de robots holonómicos. En la figura 1.9 se presenta el ejemplo de un movimiento realizado por un robot de tracción diferencial, donde es posible apreciar que se puede posicionar con facilidad el robot en el lugar deseado.

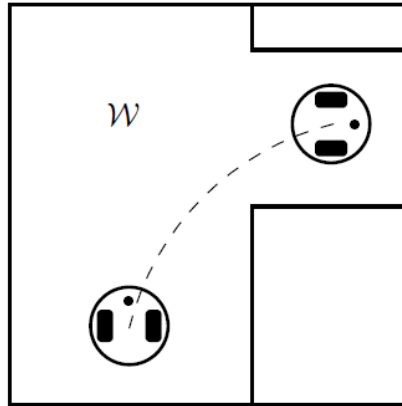


Figura 1.9. Robot móvil de tracción diferencial realizando una trayectoria.

Ejemplos de robots no holonómico son el robot tipo triciclo y el robot tipo Ackerman [21]. En la figura 1.10 se observa que al robot tipo triciclo se le dificulta llegar a la posición final ya que para lograr la orientación debe moverse hacia el frente y hacia atrás.

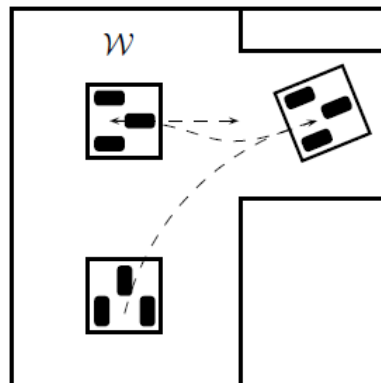


Figura 1.10. Robot móvil tipo triciclo realizando una trayectoria.

1.2 Modelado cinemático y control de robots móviles.

Tanto en los robots manipuladores como en la robótica móvil existen puntos de interés común: el modelado cinemático, el modelado dinámico, el control (arquitecturas, algoritmos...), la planificación, el reconocimiento del entorno, etc.

En el caso de robots manipuladores existe ya una abundante bibliografía que aborda los aspectos anteriores, mientras que para robots móviles todavía se están realizando un considerable número de investigaciones y desarrollos al respecto.

Para el control del vehículo o robot móvil algunos autores han planteado métodos geométricos [19] y otros la utilización de herramientas de la teoría de control clásica: aproximación lineal [3]; linealización exacta [3], [5], [13], [14]; controlabilidad [8]-[10]; estabilidad por Lyapunov [21]; control adaptativo [11], [25], [26]; control predictivo [2]; lógica borrosa [1]; etc.

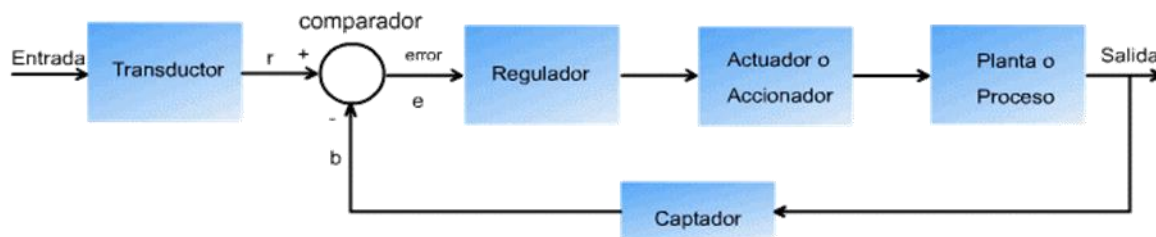


Figura 1.11. Esquema clásico de lazo de control en un robot móvil.

Uno de los principales problemas a enfrentar por los RMR son precisamente las ruedas, al no poder salvar grandes obstáculos; para esto se debe de tener en cuenta en el diseño el radio de las ruedas, su disposición en el móvil y la parte que proporcionará el movimiento, es decir cuáles van a ser las ruedas motrices. Los sistemas de los que se compone un RMR se resumen en dos partes fundamentales: un arreglo cinemático y un sistema de actuadores. Entendiendo por arreglo cinemático al análisis del movimiento sin tener en cuenta las fuerzas que lo producen, ni las que se generan; y por la parte de actuadores como aquellos recursos que proporcionan el movimiento.

La configuración por tracción diferencial se caracteriza porque el movimiento se consigue con dos ruedas, acopladas cada una a su propio motor, teniendo normalmente una o dos ruedas de apoyo. De esta forma se logra que el robot avance en línea recta fijando ambos motores a la misma velocidad, también se le puede hacer girar en una u otra dirección cuando se aplican velocidades diferentes,

y se logra que gire sobre su propio eje cuando las velocidades tienen igual magnitud con sentidos opuestos.

Con el objeto de hacer más tratable el problema del modelado en las configuraciones cinemáticas, se suelen establecer algunas suposiciones de diseño y de operación.

Dentro de las suposiciones de diseño generalmente se toman tres. La primera va dirigida a considerar que las partes dinámicas del RMR son insignificantes, que no contiene partes flexibles, de esta manera pueden aplicarse mecanismos de cuerpo rígido para el modelado cinemático. La segunda limita que la rueda tenga a lo más un eslabón de dirección, con la finalidad de reducir la complejidad del modelado. La tercera es asumir que todos los ejes de dirección son perpendiculares a la superficie, de esta manera se reducen todos los movimientos a un solo plano.

Respecto a las suposiciones de operación, al igual que en las de diseño, se toman tres. Una de ellas descarta toda irregularidad de la superficie donde se mueve el RMR. Otra, considera que la fricción de traslación en el punto de contacto de la rueda con la superficie donde se mueve, es lo suficientemente grande para que no exista un desplazamiento de traslación del móvil. Como complemento a lo anterior, una tercera suposición de operación establece que la fricción rotacional en el punto de contacto de la rueda con la superficie donde se mueve, es lo suficientemente pequeña para que exista un desplazamiento rotatorio. Aunque las suposiciones mencionadas son realistas, el deslizamiento que ocurre en el punto de contacto de las ruedas con la superficie se ha convertido en un tópico importante debido a las repercusiones que tiene sobre el móvil.

Por otro lado, el seguimiento de línea es una aplicación habitual en robótica móvil, utilizándose normalmente cables o sensores ópticos. Con los últimos avances tecnológicos se ha ido incrementando el uso de sistemas de visión para esta aplicación. La visión tiene algunas ventajas respecto a los otros métodos, aunque con el inconveniente del coste computacional asociado al tratamiento de imagen: umbralización, segmentación, detección de contornos, etc. Para mejorar dicho

tratamiento se utilizan herramientas como la transformada de Hough., transformada de Fourier, algoritmo de Canny, entre otros.

1.3 Planificación y seguimiento de trayectorias.

Para lograr que un robot móvil autónomo diseñado para interiores, que se encuentra dentro de un entorno estructurado con superficie horizontal, cambie de una posición inicial a otra posición final con movimiento controlado, se debe diseñar un algoritmo que genere referencias al sistema de control de manera automática (figura 1.12) [20, 19, 18]. El lazo de realimentación deberá obtener datos, por medio del subsistema de adquisición de datos, a partir de los sensores propioceptivos (internos) y exteroceptivos (externos). El robot necesitará entonces percibirse a sí mismo y percibir su entorno.

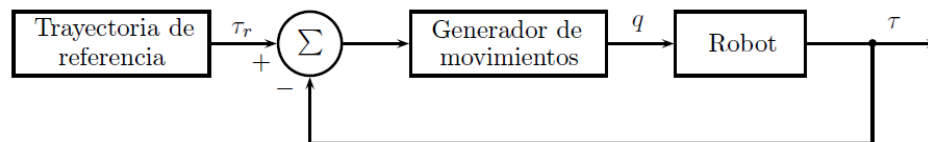


Figura 1.12. Esquema del sistema de seguimiento de trayectoria.

Para planificar el curso global de un robot móvil no sólo debe tenerse en cuenta el camino que debe recorrer, sino también la velocidad a la que lo hará, ya que existen diferentes limitaciones de velocidad que afectan significativamente a su movimiento. En el ámbito de la Robótica, la unión de las referencias espaciales y temporales que debe seguir un vehículo se denomina trayectoria, aunque no hay un consenso total a la hora de utilizar este término, y muchas veces se emplea para referirse únicamente al camino geométrico del robot.

La autonomía que el sistema de control brinda al robot, le permite realizar lo que se denomina “navegación autónoma”. Su objetivo es hacer que los robots se muevan de manera independiente en los entornos en los cuales se desempeñan, se orienten y elijan la ruta adecuada que les permita desplazarse mientras

completan la tarea asignada, sin colisiones ni otros percances. El enfoque más utilizado en la literatura para estudiar la navegación autónoma, consiste en dividirla en tres problemas, que se expresan respectivamente con tres preguntas:

1. Localización: ¿dónde estoy?
2. Planificación de Tareas: ¿hacia dónde quiero ir?
3. Planificación del Movimiento: ¿cómo puedo llegar allí?

El problema de la localización consiste en conocer en todo momento las coordenadas del robot respecto a un cierto sistema fijo de referencia.

La planificación de tareas se encarga de decidir qué acciones y en qué orden se han de realizar para completar una misión. Este es el razonamiento de más alto nivel exigido a un robot.

El objetivo de la planificación del movimiento es la generación automática de las acciones necesarias para que el robot se desplace entre dos posiciones de su entorno de trabajo evitando los obstáculos para no colisionar. El problema de la planificación se plantea de manera diferente según se disponga o no, de manera previa a la ejecución del movimiento, de un modelo del entorno de trabajo del robot. Durante la planificación, la información acerca de las características del entorno de trabajo puede estar consignada a través de un mapa del mismo, previamente obtenido. Esto permite, que el problema de la planificación se formule como un problema geométrico de cálculo de trayectorias entre dos puntos. Lo anterior constituye la base de la planificación de movimiento con información completa, conocida como planificación basada en mapa o planificación de trayectorias.

Cuando el robot va a desarrollar su misión en un área de trabajo desconocida, es decir sin información previa, entonces es necesario que realice movimientos exploratorios con el fin de levantar un mapa del entorno [15]. Sin embargo, generar el movimiento de un robot con el propósito de hacer un mapa del entorno, constituye en sí mismo un problema bastante complejo, ya que la reconstrucción de una descripción interna (o modelo) del entorno que rodea al robot, plantea a su vez varios problemas característicos:

- Un problema de percepción, que consiste en integrar información de diferentes fuentes obtenidas en diferentes instantes de tiempo a través de los sensores.
- La incertidumbre en la posición del robot hace que el problema de la localización permanezca fuertemente acoplado al de la reconstrucción del mapa, exigiendo una solución combinada de ambos.
- Adicionalmente se debe tener en cuenta el costo computacional que conllevan tanto la construcción y actualización del mapa, como su uso en la planificación de rutas.

Los algoritmos locales o con información incompleta tienen solo datos restringidos sobre el entorno de trabajo y son proporcionados por los sensores del robot en tiempo real. La información proporcionada por los sensores se puede utilizar en una estrategia diferente, que consiste en encaminarse directamente hacia el objetivo y solo cuando algún obstáculo se interponga, realizar una maniobra para evadirlo. El resultado de aplicar esta estrategia es también diferente, ya que no se puede producir con anterioridad una trayectoria explícita entre el punto de origen y el punto de destino. Los algoritmos con información incompleta, según sean convergentes o no, dan lugar a los métodos heurísticos o reactivos, y a los de planificación basada en sensores. En general a la navegación que prescinde de la planeación de rutas, se le ha denominado navegación reactiva [23]. Por su parte, en la navegación no reactiva, que hace uso de mapas en los cuales se contiene la información del entorno, es posible preestablecer una trayectoria completa desde el punto inicial al final, evitando a la vez la colisión con obstáculos. Aunque esta forma de navegación basa su desarrollo en consignas ya establecidas y en información previamente obtenida no limita el uso de sensores en el robot móvil para prever cualquier eventualidad ante cambios en las condiciones del terreno o la presencia inesperada de nuevos objetos.

Convencionalmente el problema de generación de trayectorias para un robot móvil indica que, dado un robot y un ambiente, se puede determinar una trayectoria entre dos puntos específicos si el robot está libre de colisiones y si satisface cierto criterio

de desempeño [12]. Considerando los métodos desarrollados para la planeación de trayectorias se puede proponer la siguiente clasificación:

1. Determinísticos:

- Basados en grafos:
 - Grafos de visibilidad
 - Diagramas de Voronoi
 - Modelado del espacio libre
 - Descomposición en celdas
- Campos potenciales artificiales.

2. Probabilísticos y aleatorios (algoritmos de búsqueda aleatoria):

- Planeador aleatorio de trayectorias (Randomized Path Planner RPP).
- Mapas probabilísticos (Probabilistic Road Maps PRM).
- Árboles de exploración rápida (Rapidly Exploring Random Tree RRT).
- Basados en optimización:
 - Algoritmos genéticos (Genetic Algorithms GA).
 - Colonia de hormigas (Ant Colony Optimization ACO).
 - Enjambre de partículas (Particle Swarm Optimization PSO).
 - Quimiotaxis bacteriana (Bacterial Chemotaxis BC).

1.3.1 Planificación basada en grafos

Estos métodos consisten en construir un grafo que interconecte el origen con el destino y posteriormente encontrar el camino más adecuado considerando algún índice de desempeño que por lo general es la distancia más corta.

Grafos de visibilidad

Propuesta por Nils J. Nilsson, consiste en un enfoque geométrico para construir el grafo suponiendo un entorno bidimensional donde los obstáculos se pueden

representar por polígonos. El grafo se determina aplicando el concepto de visibilidad el cual consiste en determinar si dos puntos sobre el espacio se pueden unir mediante un segmento de recta que no presente interferencia con ningún obstáculo. Se genera un grafo de conectividad entre vértices visibles y se escoge la mejor ruta. Presenta el inconveniente de generar rutas muy cercanas a los obstáculos.

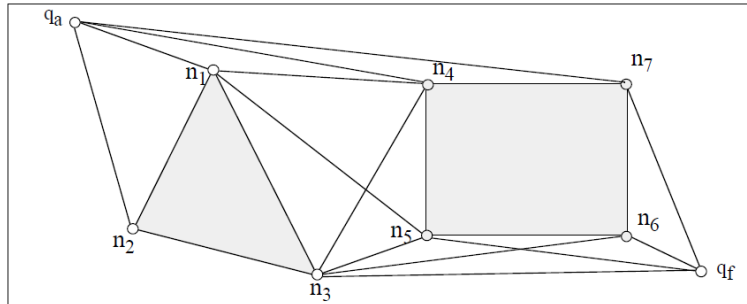


Figura 1.13. Grafo de visibilidad en un entorno de dos obstáculos.

Diagramas de Voronoi

Con este método las líneas de la ruta se localizan lo más alejado posible de los obstáculos buscando maximizar la distancia entre obstáculos y la ruta. El grafo se forma de segmentos rectilíneos y parabólicos dependiendo la forma de los obstáculos que se encuentran a igual distancia de los dos obstáculos más próximos, como se muestra en la siguiente figura.

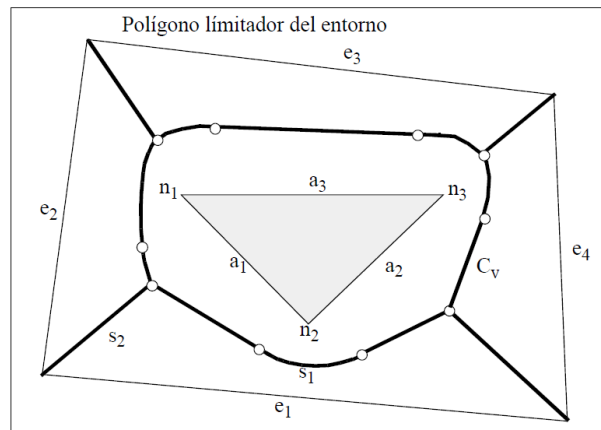


Figura 1.14. Retracción del espacio libre en un diagrama de Voronoi.

Modelado del espacio libre

Método desarrollado por Rodney A. Rrooks, generalmente se aplica a espacios de trabajo con obstáculos poligonales (figura 1.15). La construcción del grafo se realiza mediante cilindros rectilíneos generalizados (CRG). Al emplear los CRG se busca que el móvil se desplace de la forma más lejana de los obstáculos. La ruta desde la configuración inicial hasta la final está compuesta por una serie de CRG interconectados.

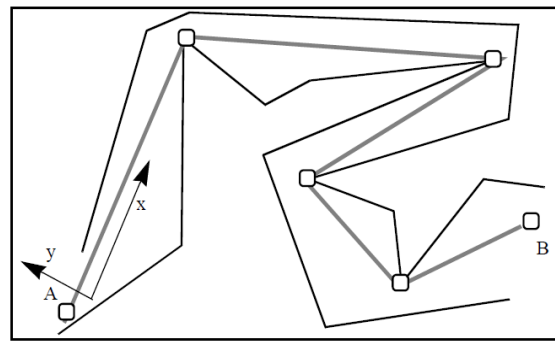


Figura 1.15. Planificación con el espacio libre de obstáculos modelado mediante cadenas

Descomposición en celdas

Este método consiste en descomponer el espacio libre en un conjunto de celdas desde el punto inicial hasta el final, por lo tanto, el problema consiste en encontrar una sucesión de celdas que no presente discontinuidades. Mediante un grafo es posible determinar la conectividad de las celdas (figura 1.16).

Los métodos empleados para la descomposición de celdas pueden ser exactos o aproximados. Los métodos exactos consisten en descomponer el espacio libre en celdas donde la unión de estas corresponde al espacio libre. En los métodos aproximados se construyen celdas de una forma establecida donde su unión no es exactamente el espacio libre. Con los métodos exactos se puede garantizar encontrar un camino si existe, lo cual no se puede garantizar con los métodos aproximados. La ventaja que tienen los métodos aproximados es su facilidad de implementación.

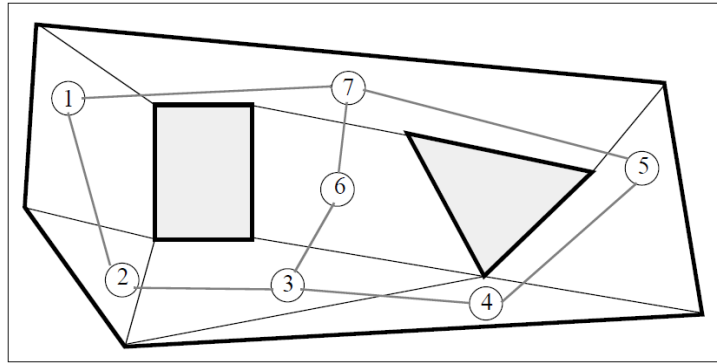


Figura 1.16. Descomposición en celdas y grafo de conectividad

Los distintos métodos basados en este principio, se distinguen por la forma en la cual realizan la descomposición en celdas y como se construye el grafo de conectividad. El método más sencillo de descomposición del espacio libre del entorno en celdas resulta el denominado descomposición trapezoidal [21]. Este método se basa en la construcción de segmentos rectilíneos paralelos al eje Y del sistema global F_g a partir de los vértices de cada uno de los elementos del entorno. El final del segmento queda delimitado por el primer corte de la línea con un elemento del entorno. Esta descomposición es la mostrada en la figura 1.17.

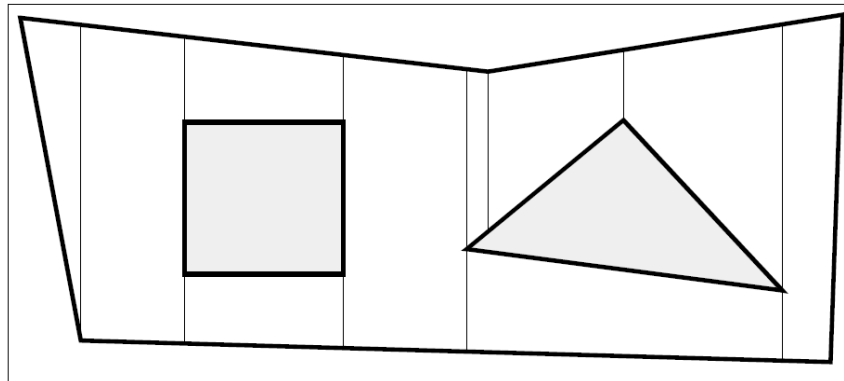


Figura 1.17. Descomposición trapezoidal del espacio libre

El grafo de conectividad se construye por medio de la unión de los puntos medios de los segmentos verticales definidos (Figura 1.18).

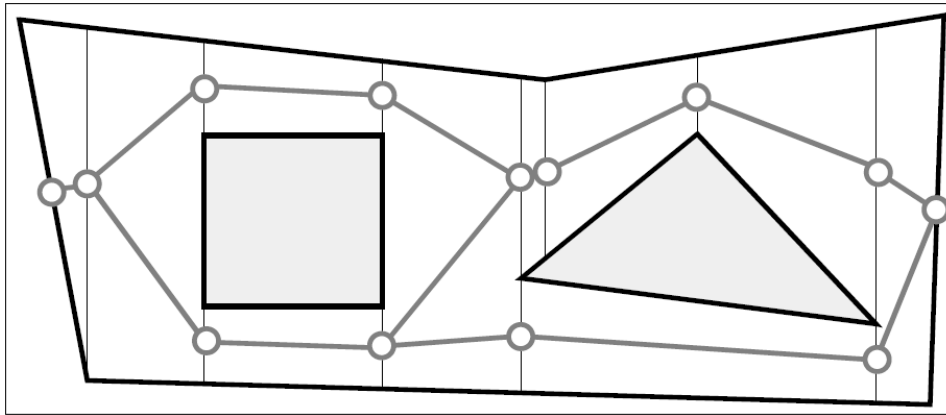


Figura 1.18. Grafo de conectividad de una descomposición trapezoidal

Este tipo de enfoque se presta a muchas variantes, por ejemplo, la utilización de varios niveles de resolución para una búsqueda jerarquizada, o bien el uso de celdas en tres dimensiones para la planificación de caminos en espacios tridimensionales

1.3.2 Planeación de trayectorias empleando campos potenciales

El método de campos potenciales es una de las técnicas más populares en la generación de trayectorias para robots móviles por ser una forma elegante en la solución de este problema. De los primeros desarrollos se tiene el presentado por Khatib [12], aquí se propone que los obstáculos y el robot tienen una carga eléctrica del mismo signo con la finalidad de generar una repulsión mientras que la meta tiene asociada una carga eléctrica de signo opuesto para atraer al robot al punto destino (figura 1.19). Convencionalmente los potenciales atractivos y repulsivos se generan de forma separada y el potencial total del espacio de trabajo es obtenido como la suma de estos.

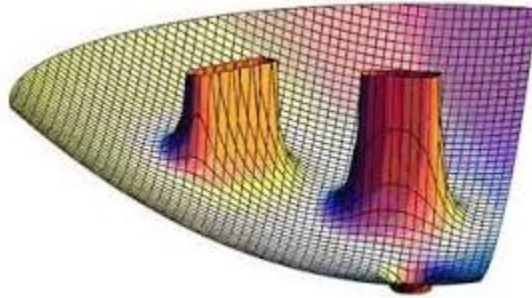


Figura 1.19. Ejemplo de Campo Potencial Artificial con 2 obstáculos.

Para generar la trayectoria con estos potenciales solo se requiere el cálculo de gradientes, sin embargo, los campos potenciales generados pueden ser no convexos presentando mínimos locales lo cual dejaría caer al robot en una trampa. Otra desventaja consiste en la estimación de los parámetros del campo potencial para lograr una adecuada evasión de obstáculos. Modificaciones al planteamiento original de campos potenciales se realizan considerando las anteriores limitaciones, Volpe y Khosla proponen funciones potenciales supercuadráticas, donde se utiliza un amplio conjunto de formas desde rectángulos hasta elipses empleando fórmulas supercuadráticas.

Una propuesta basada en la solución de la ecuación de Laplace para la transferencia de calor es propuesta de forma separada por Connolly y adicionalmente por Kim y Khosla, este método no presenta mínimos locales para configuraciones simples y se genera mediante funciones potenciales armónicas (Harmonic Potential Functions). Para el caso donde se tienen obstáculos en movimiento S.S. Ge y Y.J. Cui, proponen un nuevo conjunto de funciones potenciales las cuales dependen tanto de la posición como de la velocidad de los obstáculos.

Un problema adicional identificado tanto por S.S. Ge y Y.J. Cui como por R. Volpe y P. Khosla, aparece cuando el campo potencial de un obstáculo encierra el objetivo lo cual ocurre cuando estos dos se encuentran muy cercanos, este problema es denominado GNRON (Goals Non Reachable With Obstacle Nearby), los anteriores

autores en sus trabajos proponen diferentes tipos de campos potenciales para solucionar este problema.

Los métodos anteriores debido a su característica de adición de campos asociados a los obstáculos se denominaron campos potenciales locales. En contraste con los campos potenciales globales donde se considera la totalidad de los obstáculos como también el objetivo para construir el campo potencial como un todo y no como la adición de partes, con esto se busca eliminar el problema asociado a los mínimos locales. Bajo este nuevo enfoque aparecen las funciones de navegación (Navigation Function), propuestas por Rimon y Koditschek las cuales son explícitamente diseñadas para presentar un único mínimo local en el objetivo puesto que los demás puntos de equilibrio son enmascarados (rellenados). Para construir las funciones de navegación es necesario conocer completamente el espacio de trabajo.

1.3.3 Algoritmos de búsqueda aleatoria

Este tipo de métodos se pueden considerar como locales puesto que solo exploran una parte del espacio de configuraciones y están basados en movimientos aleatorios para determinar la respectiva secuencia de configuraciones para que el robot pase de la configuración inicial a la final.

Planeación aleatoria de trayectorias

El algoritmo de planeación aleatoria de trayectorias (Randomized Path Planner RPP) fue desarrollado por Jean-Claude Latombe y Jerome Barraquand en 1991. Este algoritmo emplea movimientos Brownianos para poder salir de mínimos locales.

Mapas probabilísticos

Este método fue propuesto en 1992 de forma separada por Jean-Claude Latombe y Lydia E. Kavraki en la Universidad de Stanford y en la Universidad de Utrecht desarrollado por los investigadores Mark H. Overmars y Peer Svestka.

Los mapas probabilísticos (Probabilistic Road Map Method PRM) buscan crear un mapa generado de forma aleatoria libre de colisiones el cual conecta de forma rápida el móvil con el objetivo.

Arboles de exploración rápida

Una alternativa en la generación aleatoria de trayectorias son los arboles de exploración rápida (Rapidly Exploring Random Tree RRT), este método fue propuesto por Steven M. LaValle en 1998. En este método se crean árboles que se expanden de forma aleatoria para tener una conexión desde el punto de partida hasta el objetivo.

Métodos basados en optimización estocástica

Estos métodos utilizan técnicas de optimización estocástica por lo general bioinspiradas. Una primera aplicación emplea algoritmos genéticos en la planeación de trayectorias. Este tipo de enfoque es útil en espacios de búsqueda muy amplios. Otra estrategia probabilística, consiste en emplear la técnica de optimización basada en colonias de hormigas.

Trabajos adicionales para la planeación de trayectorias con métodos de optimización bioinspirada consisten en la aplicación del algoritmo de optimización basado en quimiotaxis bacteriana y optimización basada en comportamientos de enjambres Particle Swarm Optimization (PSO).

Un trabajo donde se examina de forma general la planeación de trayectorias empleando técnicas evolutivas se observa en la tesis doctoral Planificación y Gestión de Trayectorias basadas en técnicas evolutivas, de la Universidad de Washington.

Comportamiento de enjambres

En la naturaleza se pueden apreciar diferentes comportamientos los cuales han sido de atención para su descripción de forma analítica, en particular el comportamiento

de congregaciones de individuos resulta ser una temática interesante por los diferentes patrones que forman los individuos y que están dados por la interacción de los mismos.

El comportamiento de enjambres ha sido empleado con éxito en problemas de optimización, logrando convergencia en problemas multiobjetivo de alta complejidad. La planeación de trayectorias por campos potenciales puede ser visto como un problema de optimización donde se busca conducir una partícula, o grupo de partículas, a un mínimo global o meta. Desafortunadamente y como es común en problemas de optimización, existen complicaciones ya que al generar los campos potenciales se presentan mínimos locales los cuales puede hacer que la trayectoria nunca converja a la meta.

Por lo general el enfoque de enjambre se ha empleado para la coordinación de un grupo de robots. También se han reportado aplicaciones en las cuales los enjambres se emplean en la planeación de trayectorias para poder evadir mínimos locales. Los trabajos más relevantes con este enfoque son los desarrollados en [9], sin embargo por el momento no se han empleado ampliamente modelos de partículas activas Brownianas.

Una vez puestos en práctica los diferentes métodos para la obtención de la trayectoria, se hace necesario contar con técnicas para seguirla. Esta tarea consiste en determinar las acciones a realizar en los actuadores para seguir la trayectoria deseada, lo que se conoce como control de movimiento. Según las técnicas empleadas en esta operación surgen varios métodos.

1.3.4 Métodos de Seguimiento

Una vez realizada la trayectoria que debe seguir el robot móvil para lograr el objetivo final, es necesario implementar un algoritmo para que el robot realice lo más fielmente posible esta trayectoria de forma autónoma para alcanzar el punto de destino o final de la trayectoria.

Entre los métodos para el seguimiento de trayectoria se encuentra el conocido como ajuste de polinomios de quinto orden (Quintic Polynomial Fit). En el cual se trata de encontrar una curva que enlace la posición actual del vehículo con el punto objetivo. Otra técnica recientemente empleada es mediante linealización de entrada-salida la cual impone que las variables de estado tiendan asintóticamente a la trayectoria deseada, generando una dinámica remanente asociada a una variable de estado, la cual resulta ser estable.

Sin embargo, uno de los métodos más empleados en el seguimiento de trayectoria es el de persecución pura (pure pursuit), éste genera arcos entre el punto de desplazamiento del móvil y los puntos de la trayectoria a seguir, los arcos se generan de 10 a 15 veces por segundo lo que resulta en un seguimiento suave y con muy buenos resultados.

Seguimiento mediante persecución pura.

Considerándose un sistema de referencia local asociado al movimiento del vehículo, tal como se muestra en la figura 1.20. Se supone que, en el intervalo de control, la curvatura es constante, describiendo el vehículo un arco de circunferencia.

Del análisis de la figura se deducen las relaciones:

$$r = \Delta x + d \tag{1.1}$$

$$d^2 + (\Delta y)^2 = r^2 \tag{1.2}$$

Siendo r el radio de curvatura del vehículo.

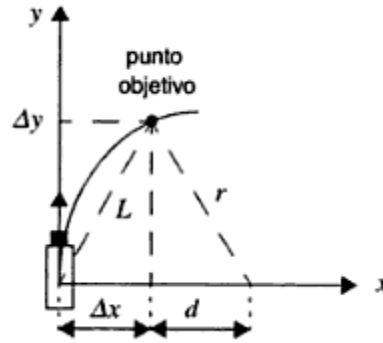


Figura 1.20. Seguimiento de caminos mediante persecución pura.

Despejando d en la primera y sustituyendo en la segunda se obtiene:

$$(r - \Delta x)^2 + (\Delta y)^2 = r^2 \quad (1.3)$$

De donde el radio de curvatura necesario para que el vehículo se desplace Δx , Δy es:

$$r = \frac{(\Delta x)^2 + (\Delta y)^2}{2\Delta x} \quad (1.4)$$

Por tanto, la curvatura que es necesario suministrar al vehículo es:

$$\gamma_r = \frac{1}{r} = -\frac{2(\Delta x)}{L^2} \quad (1.5)$$

Donde el signo viene dado por el sentido de giro necesario para alcanzar el punto objetivo en la figura 1.20, L es la distancia a la que se encuentra el punto objetivo y Δx es el desplazamiento lateral.

La expresión anterior constituye la ley de control de persecución pura ("pure pursuit"). Obsérvese como es una ley de control proporcional al error lateral (Δx) con respecto al punto objetivo. La constante de proporcionalidad (ganancia) varía con la inversa del cuadrado de L .

En la figura puede verse también que la curvatura de la persecución pura es la inversa del radio de una de las circunferencias que pasa por la posición actual del vehículo y por el punto objetivo.

El programa para la aplicación de esta ley de control es muy sencillo. Basta determinar el punto del camino que se encuentra a una distancia previamente definida L , y calcular el error lateral (Δx) con respecto a la posición actual del centro de guiado del vehículo. Si las coordenadas están en un sistema global, es necesario tener en cuenta la orientación del vehículo para obtener (Δx). En efecto, si el vehículo está en las coordenadas globales (x, y) con orientación θ y el punto objetivo sobre el camino está en las coordenadas globales (x_{ob}, y_{ob}) entonces se tendrá:

$$\Delta x = (x_{ob} - x) \cos \theta + (y_{ob} - y) \sin \theta \quad (1.6)$$

1.4 Navegación basada en visión.

La visión por computador fue un tema bastante destacado durante la década de 1980 y la primera mitad de 1990, y sus ventajas fueron inmediatamente aprovechadas para mejorar las capacidades de navegación de los robots. La facilidad con la que “vemos”, llevó a pensar a los primeros investigadores en inteligencia artificial, por 1960, que hacer que una computadora interpretara imágenes era relativamente fácil, pero no resultó así, y muchos años de investigación han demostrado que es un problema muy complejo. Sin embargo, en los últimos años hay avances considerables, básicamente por 3 factores:

- El desarrollo tecnológico en las capacidades de procesamiento y de memoria en las computadoras, que facilita el almacenamiento y procesamiento de las imágenes.
- Los avances teóricos en los principios y algoritmos para el procesamiento y análisis de Imágenes.

- La creciente necesidad del procesamiento automático de imágenes, que se capturan y almacenan en grandes cantidades en diversos dominios, como en medicina, seguridad, tránsito de vehículos, etc.

1.4.1 Visión artificial

La Visión Artificial, también llamada Visión por Computador, es una rama de la Inteligencia Artificial que recoge el conjunto de técnicas que permiten a una máquina reconocer la estructura y propiedades de las imágenes que recibe tal y como lo haría un ser humano. Para conseguir esto la Visión Artificial utiliza, entre otras, técnicas de procesado de imágenes, de reconocimiento de patrones, aprendizaje estadístico y teoría de grafos.

Un área muy ligada a la de visión computacional es la de procesamiento de imágenes. Aunque ambos campos tienen mucho en común, el objetivo final es diferente. El objetivo del procesamiento de imágenes es mejorar la calidad de las imágenes para su posterior utilización o interpretación, por ejemplo:

- remover defectos,
- remover problemas por movimiento o desenfoque,
- mejorar ciertas propiedades como color, contraste, estructura, etc.
- agregar “colores falsos” a imágenes monocromáticas.
- Mientras que el objetivo de la visión computacional es extraer características de una imagen para su descripción e interpretación por la computadora. Por ejemplo:
 - determinar la localización y tipo de objetos en la imagen

La visión es un recurso más que los robots pueden utilizar para inferir el estado de su entorno. Esta entrega un flujo continuo de imágenes, pero es difícil extraer de ese flujo información directamente útil para la acción del robot. Las imágenes que la cámara proporciona se pueden capturar en un ordenador y poner disponibles a los programas de control.

Los objetivos típicos de la visión artificial incluyen:

- La detección, segmentación, localización y reconocimiento de ciertos objetos en imágenes (por ejemplo, caras humanas).
- La evaluación de los resultados (por ejemplo, segmentación, registro).
- Registro de diferentes imágenes de una misma escena u objeto, es decir, hacer concordar un mismo objeto en diversas imágenes.
- Seguimiento de un objeto en una secuencia de imágenes.
- Mapeo de una escena para generar un modelo tridimensional de la escena; este modelo podría ser usado por un robot para navegar por la escena.
- Estimación de las posturas tridimensionales de humanos.
- Búsqueda de imágenes digitales por su contenido.

La visión por computador es un proceso que incluye en el mismo varias etapas, entre las principales se encuentran las representadas en la figura 1.21.

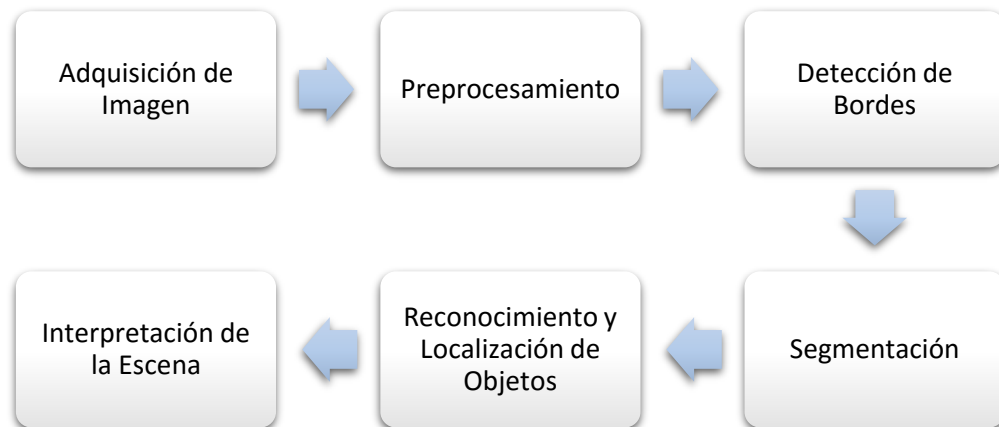


Figura 1.21. Etapas de un Sistema de Visión por Computador

La adquisición de la imagen, es el proceso a través del cual se obtiene una señal eléctrica conteniendo la información visual de la imagen y su conversión A/D que permite tenerla disponible en la memoria de la PC.

El preprocesamiento incluye técnicas como reducción de ruidos y realce de detalles

La detección de bordes suministra una valiosa información sobre las fronteras de los objetos y puede ser utilizada para segmentar la imagen, reconocer objetos, etc.

La mayoría de las técnicas para detectar bordes emplean operadores locales basados en distintas aproximaciones discretas de la primera y segunda derivada de los niveles de grises de la imagen.

La segmentación es el proceso que divide una imagen en regiones significativas que sean de interés para cada aplicación en particular.

El reconocimiento es el proceso que clasifica estos objetos a partir de características obtenidas.

La interpretación le otorga un significado simbólico a un conjunto de objetos reconocidos y a sus relaciones geométricas y topológicas.

En el caso particular de la robótica la visión se utiliza como elemento para cerrar el lazo de control en tareas de interacción inteligente de un robot con su entorno de trabajo y tendría el proceso una etapa adicional denominada comunicación Robot - Visión (figura 1.22).

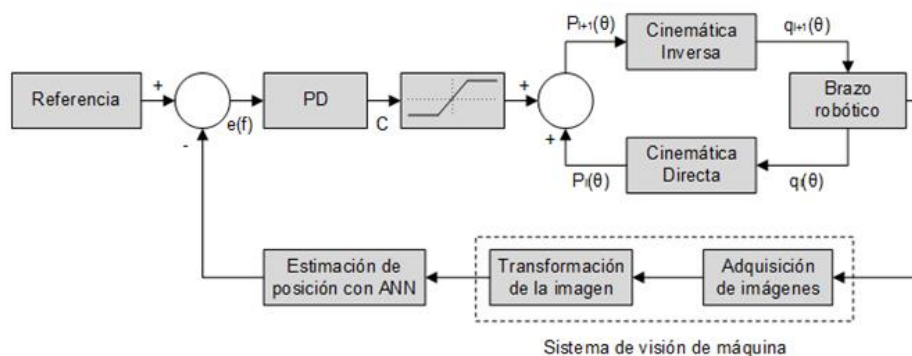


Figura 1.22. Lazo de control de un robot que incorpora visión.

Por tal motivo, la misión de un sistema de visión para robot consiste en proporcionar, a partir de la información visual contenida en una imagen (o serie de imágenes) del entorno significativo del robot, una descripción de dicho entorno, entendiéndose por descripción al vector de atributos que dentro de un espacio de caracterización permite la diferenciación entre clases de objetos para su reconocimiento de manera automática y relativamente inteligente.

1.4.2 Métodos utilizados para la extracción de características de objetos.

Códigos de Cadena

Es un tipo de estructura de datos para representar el contorno de un objeto en una imagen binaria mediante una secuencia de segmentos conectados consecutivamente, de longitud y orientación específica, que conectan píxeles adyacentes.

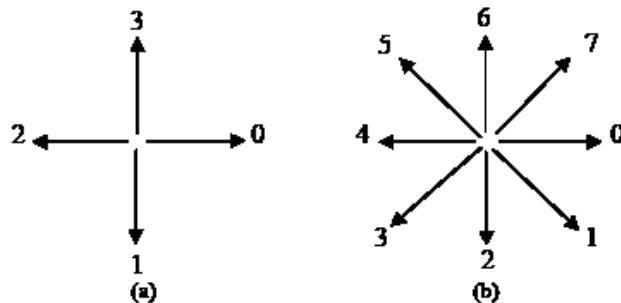


Figura 1.23. Direcciones para entornos de a) 4 vecinos b) 8 vecinos.

La conexión de los segmentos se lleva cabo en entornos de 4 ó 8 vecinos (figura 1.23). Cuando se usa un entorno de 4 vecinos se tienen 4 orientaciones para los segmentos y se utilizan los números 0, 1, 2 y 3. Si se usa un entorno de 8 vecinos se tienen 8 orientaciones para los segmentos y se utilizan los números 0, 1, 2, 3, 4, 5, 6 y 7. Ejemplo en la siguiente figura.

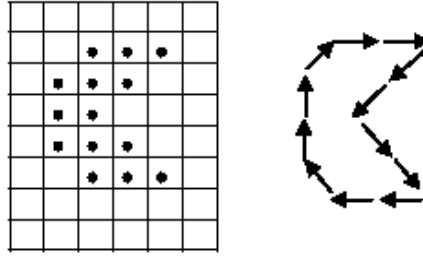


Figura 1.24. Contorno de un objeto y su código de cadena 003311445667.

Es importante notar que el código de cadena depende del punto de partida, es decir que no es invariante a rotación. Este inconveniente se puede eliminar si se considera el código como una lista circular en el caso de contornos cerrados. Además, como es conveniente que tenga la propiedad de unicidad entre todas las cadenas posibles según el punto de partida que se haya tomado, se puede elegir aquella que corresponda al menor número entero.

Firmas

Es una representación de un contorno mediante una función real unidimensional.

Una de las formas más simples de definir una firma es a través de la distancia desde un punto interior (puede ser el centroide del contorno) a cada uno de los puntos del contorno, como una función del ángulo, tal y como se muestra a continuación:

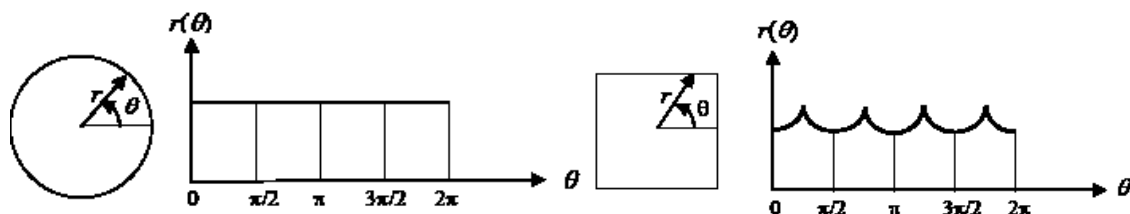


Figura 1.25. Firmas generadas para una circunferencia y para un cuadrado.

Sólo tiene sentido usar las firmas cuando el vector que se extiende desde el origen corta en un solo punto al contorno. Es invariante frente a traslaciones, pero no frente a rotaciones o cambios de escala.

Para resolver el inconveniente de que no sea invariante frente a rotaciones, se puede comenzar la representación por el ángulo para el cual la distancia entre el centroide y el punto del contorno es máxima.

Para resolver el inconveniente de que no sea invariante frente a cambio de escala, se consigue dividiendo la función por la distancia máxima al centroide, de forma que la distancia máxima sea uno.

Inconvenientes de las firmas:

1. Es muy sensible a la posición del centroide.

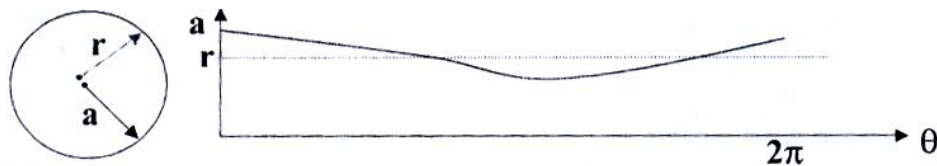


Figura 1.26. Influencia de la posición del centroide.

2. Las concavidades pueden dar lugar a una representación multielevada para algunos ángulos.

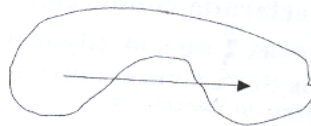


Figura 1.27. Varias distancias para un mismo ángulo.

Aproximaciones poligonales

Se trata de aproximar de la mejor manera posible el contorno de un objeto mediante una curva de tramos lineales que constituye un polígono, y donde los vértices de dicho polígono son una representación del contorno del objeto.

Como criterio para evaluar la calidad del ajuste se puede elegir el criterio de mínimos cuadrados.

Suponer la curva que se muestra en la siguiente figura, que va del punto x_1 al punto x_N y sean x_1, x_2, \dots, x_{N-1} las coordenadas de $N-2$ puntos intermedios de la curva. Si d_i es el punto del segmento $[x_1, x_N]$ más próximo al punto x_i , $i=2,3,\dots,N-1$, entonces $|x_i - d_i|$ es el error de aproximación de la curva por el segmento lineal $[x_1, x_N]$ correspondiente al píxel x_i .

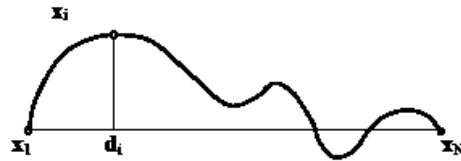


Figura 1.28. Aproximación poligonal.

La aproximación poligonal consistirá en determinar los vértices del polígono de manera que el error total sea mínimo. Sin embargo, encontrar la solución óptima para este problema requiere un costo computacional muy alto.

Para ello hay técnicas alternativas que suministran buenas soluciones y son más rápidas y eficientes.

Las técnicas de partición consisten en ir dividiendo de forma recurrente los tramos de la curva representados por un segmento en dos tramos, representados a su vez cada uno por un segmento. De manera que la medida local de error sea mínima.

Si se utiliza el criterio mínimax, es decir, minimizar el error dado por la expresión $\max_{2 \leq i \leq N-1} |x_i - d_i|$, entonces para cada tramo de la curva representado por un segmento, se determina el punto de la curva más alejado del segmento y se toma dicho punto como nuevo vértice.

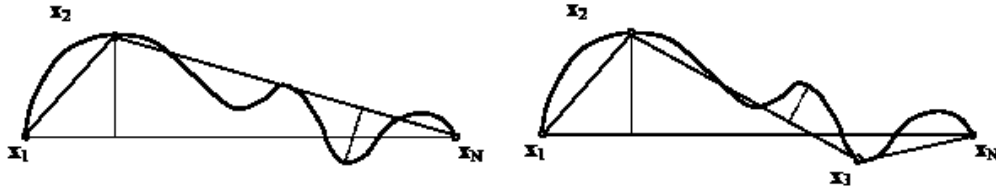


Figura 1.29. Aproximación poligonal por partición sucesiva.

El proceso acaba cuando el error máximo por píxel es menor que cierta cantidad prefijada. Los puntos x_1 y x_N pueden ser los 2 puntos más alejados del contorno. Dichos puntos dividen el contorno en dos partes y el algoritmo se aplica a cada una de ellas.

Obsérvese que en cada caso el error del píxel es igual o menos que el del paso anterior. Conforme se van introduciendo nuevos vértices se mejora la aproximación.

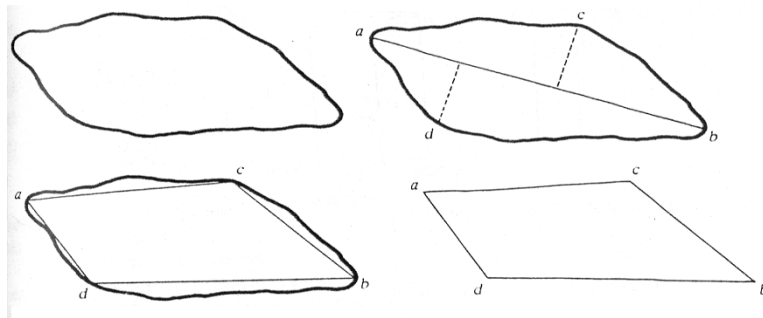


Figura 1.30. Aproximación poligonal para un contorno cerrado por partición sucesiva.

Las técnicas de incorporación operan al contrario, parten de un punto x_1 de la curva y van recorriendo los píxeles de la curva en el sentido de las manecillas del reloj.

Cuando se encuentra el primer píxel digamos x_i , para que el error de x_1 a x_i supere un umbral T (prefijado), se declara dicho punto como vértice del polígono y se repite el proceso comenzando ahora en x_i . Así se genera una secuencia de vértices que constituye la aproximación poligonal.

Esta técnica tiene el inconveniente de que los vértices no suelen ser puntos de inflexión, como ocurre con las técnicas de partición, generando vértices que no coinciden con las esquinas de la curva.

1.5 Conclusiones del capítulo

De la bibliografía consultada y los métodos mencionados para el control de la navegación de robots móviles se llega a las siguientes conclusiones:

1. Queda demostrada la sencillez y agilidad del robot con ruedas tipo diferencial escogido para el seguimiento de la trayectoria.
2. Se utilizará como sensor externo una cámara digital para captar una imagen del espacio de tarea del robot.
3. Para generar la trayectoria a seguir por el robot se utilizará el método de los campos potenciales.
4. Para el seguimiento de trayectoria se utilizará el método de persecución pura.
5. El código de cadena es un método sencillo de implementar para obtener una descripción del contorno de un objeto, pero también se podrá utilizar para determinar los puntos de referencia en la trayectoria a seguir.

CAPÍTULO 2. GENERACIÓN Y SEGUIMIENTO DE TRAYECTORIA PARA UN ROBOT MÓVIL BASADO EN CAMPOS POTENCIALES UTILIZANDO VISIÓN.

Introducción

En este capítulo se propone un algoritmo para la generación de la trayectoria de un robot móvil. El algoritmo propuesto se basa en los métodos de campos potenciales y el uso de una cámara digital para la adquisición de la imagen que representa el plano de trabajo. Se muestra la obtención del modelo cinemático de un robot móvil de tracción diferencial, por ser el tipo de robot utilizado para esta investigación y la validación del modelo mediante los resultados de la simulación. Se utilizan criterios para generar y optimizar la trayectoria generada. Se utiliza el método del código de cadena para determinar los waypoints de la trayectoria generada. Se corroboran los resultados mediante simulación a través del toolbox de Robótica del Matlab, utilizando funciones propias de esta herramienta mediante el método de persecución pura para el seguimiento de la trayectoria generada. Por último, se realiza una comparación del método desarrollado con el método propuesto en [9].

2.1 Fundamentos teóricos de la propuesta.

En el capítulo anterior se analizaron diferentes métodos para la generación de trayectoria en robot móviles, así como métodos para el seguimiento de la trayectoria generada. También se mencionaron varias técnicas utilizadas para describir objetos presentes en imágenes digitales.

Constituyen aspectos teóricos del fundamento de la propuesta la generación de una trayectoria de un punto inicial a otro final evadiendo obstáculos utilizando el método de campos potenciales [25], el método del código de cadena [26] para la obtención de los waypoints de la trayectoria, así como el método de persecución pura para el seguimiento de la trayectoria generada.

El método de campos potenciales permite generar una trayectoria entre el punto de inicio y final dando un valor diferente a cada objeto sobre el área de trabajo. El

objetivo o punto de llegada se considera como un imán de polaridad contraria a la del vehículo, es decir que tiene fuerza de atracción mientras que los obstáculos se comportan como imanes de igual polaridad generando fuerzas repulsivas.

El método del código de cadena permite la representación del contorno de un objeto presente en la imagen, este se utilizará para determinar los puntos de interés en la trayectoria generada, que servirán de referencia al controlador diseñado según el método de persecución pura.

2.2 Cinemática de un robot móvil.

Desde el punto de vista de la cinemática, la principal diferencia entre un robot manipulador y un robot móvil estriba en la naturaleza y disposición de sus articulaciones. La primera, se suele modelar en forma de cadena cinemática abierta, compuesta de una alternancia de sólidos rígidos con articulaciones de un solo grado de libertad (prismático o de revolución). Por el contrario, la estructura cinemática de un robot móvil terrestre, se puede considerar como un conjunto de cadenas cinemáticas cerradas, cuyo número de grados de libertad está dado por la cantidad de miembros que estén en contacto con el suelo. En el caso de RMR está dada por la cantidad de ruedas en contacto con el suelo.

Normalmente, se consideran las siguientes limitaciones para la construcción del modelo cinemático de un mini robot móvil RMR:

- El robot se mueve sobre una superficie plana.
- No existen elementos flexibles en la estructura del robot (incluidas las ruedas).
- Se considera que las ruedas poseen un eje de direccionamiento, que siempre es perpendicular al suelo.
- Se desprecia todo tipo de fricción en los elementos móviles del vehículo contra el suelo.

Un rasgo especial de los robots móviles es su naturaleza no holonómica, en contraste con los robots manipuladores. Las restricciones holonómicas son aquellas que son expresadas como un sistema de ecuaciones algebraicas con las variables de posición, tanto de traslación como rotacionales, las que hacen que el sistema sea integrable sin involucrar ninguna variable de velocidad. También se puede constituir un sistema de ecuaciones con variables de velocidad que logren integrarse, así de conducir el sistema de restricciones a uno con variables de posición.

Si el sistema de restricciones de velocidad no es integrable, se dice que las restricciones son no holonómicas. Por tanto, si un sistema mecánico está sujeto sólo a restricciones holonómicas, se dice, que el sistema es holonómico; de lo contrario, es no holonómico. Los manipuladores compuestos de pares prismáticos y de revolución son ejemplos de sistemas holonómicos, mientras que los robots móviles normalmente constituyen sistemas no holonómicos.

En general, los robots móviles pueden colocarse en dos grupos, los que poseen 2 grados de libertad (gdl) o 3 grados de libertad. Una de las definiciones está dada por el número de articulaciones o ruedas energizadas. Por ejemplo, un robot móvil de 3 gdl es un vehículo con tres ruedas motrices y una de éstas puede ser direccional o todas son ruedas direccionales. Por otro lado, un robot móvil de 2 gdl puede ser un vehículo con tres ruedas, pero sólo dos son motrices y la rueda restante es no energizada, este es el caso del robot de configuración diferencial.

Cuando un robot móvil se desplaza sobre una superficie plana normalmente tiene tres grados de libertad dependiendo de su configuración, estos grados de libertad corresponden a la posición del robot en el plano (x, y) y su orientación θ , por lo cual, en un modelo cinemático es de interés la representación de la posición y orientación del robot en función de las acciones de los actuadores dando lugar a la cinemática directa e inversa del robot. Para robots con ruedas el desplazamiento se produce por el contacto de las ruedas con la superficie de tal forma que se considera que la rueda no presenta deslizamiento lateral ni frontal.

Con la cinemática directa de un robot móvil se puede determinar la posición final del robot conociendo la posición inicial y los movimientos realizados. Mediante esta descripción se puede pasar del espacio de configuraciones al espacio de trabajo. La cinemática inversa de un robot móvil permite establecer la serie de movimientos que el robot debe realizar para desplazarse de la posición inicial a la final. Mediante esta cinemática se puede pasar del espacio de trabajo al espacio de configuraciones.

2.2.1 Modelo cinemático de un robot móvil de tracción diferencial.

Para realizar el análisis cinemático del robot móvil de tracción diferencial en primer lugar se determina su velocidad lineal y angular considerando las velocidades lineales de las ruedas. En la figura 2.1 se pueden observar las velocidades asociadas al robot móvil de tracción diferencial.

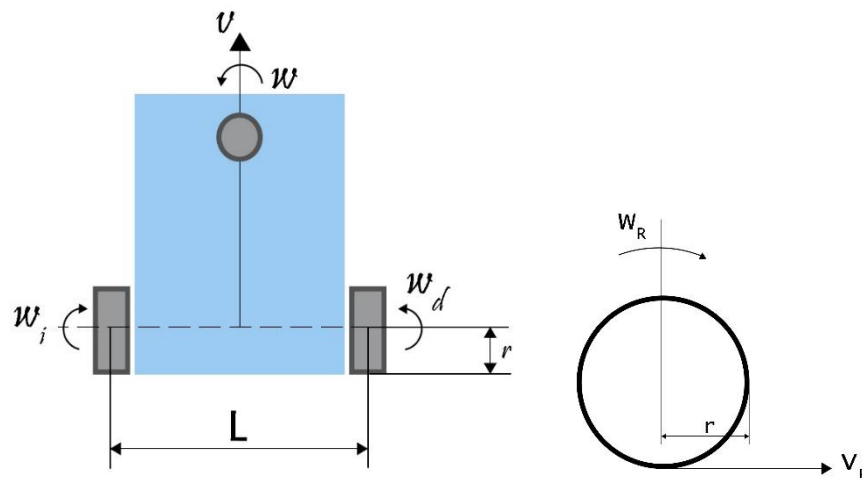


Figura 2.1. Diagrama del robot móvil de configuración diferencial, con las velocidades asociadas a la rueda.

Para realizar el modelado del robot, es necesario conocer las condiciones físicas del sistema, en especial las dimensiones del robot. Las medidas que interesan en el proceso de modelado son: la distancia entre las ruedas y el radio de las mismas. Como se denota en la figura 2.1 la distancia entre las ruedas se va a denominar L y el radio de las mismas será r .

Saber con exactitud la posición y orientación del robot con respecto al sistema global, son las incógnitas que se tienen en un robot móvil. Para poder lograr un movimiento controlado, se deben tener dominio sobre V_d la velocidad lineal de la rueda derecha y V_i la velocidad lineal de la rueda izquierda.

El objetivo de modelar el robot es buscar una relación directa de cómo afectan las entradas V_d y V_i a los estados del sistema X, Y, θ .

El robot puede trasladarse con velocidad lineal v y rotar con una velocidad angular w , como se pudo observar en la figura 2.1.

Para que el robot se desplace en línea recta la velocidad de sus ruedas debe ser igual, por lo tanto, se puede definir la velocidad lineal del robot, como el promedio de las velocidades de las ruedas.

$$v = \frac{v_d + v_i}{2} \quad (2.1)$$

Una representación de estas velocidades es la mostrada en la Figura 2.2.

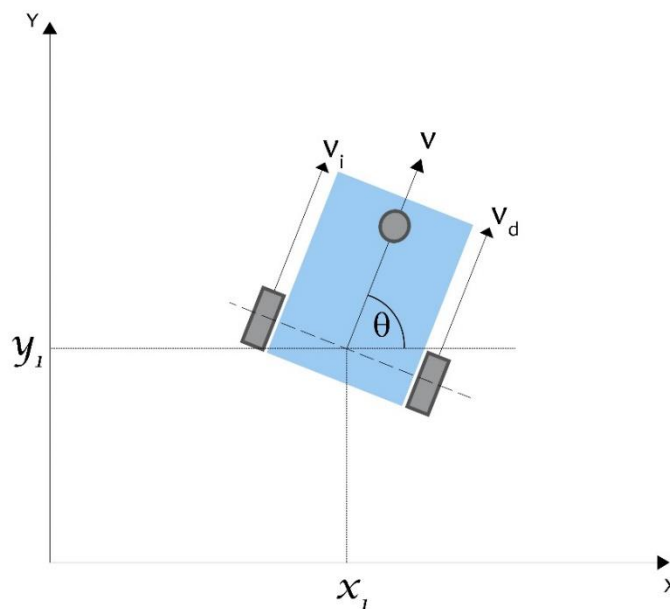


Figura 2.2. Diagrama de las velocidades en el robot móvil.

A su vez la velocidad lineal de cada rueda es proporcional a la velocidad angular de estas, la relación viene dada por la siguiente expresión:

$$v_d = r * w_d \quad v_i = r * w_i \quad (2.2)$$

Sustituyendo las expresiones de (2.2) en (2.1), se obtiene:

$$v = \frac{r(w_d + w_i)}{2} \quad (2.3)$$

Para que el robot tenga un movimiento de rotación sobre su mismo centro de masa, las velocidades de sus ruedas deben tener la misma magnitud, pero signo diferente, por ende, se puede definir la velocidad angular como la diferencia de la velocidad de sus ruedas sobre la longitud que hay entre ellas.

$$w = \frac{v_d - v_i}{L} \quad (2.4)$$

Este movimiento al igual que la velocidad lineal es proporcional al radio de las ruedas. Lo que se demuestra al sustituir las expresiones de (2.2) en (2.4), quedando:

$$w = \frac{r(w_d - w_i)}{L} \quad (2.5)$$

Teniendo las velocidades que se presentan en el desplazamiento del robot se procede a integrarlas en un primer concepto de movimiento del robot, para obtener unas ecuaciones que definan la dinámica del movimiento del robot en cada eje.

$$\dot{x} = -v \sin \theta$$

$$\dot{y} = v \cos \theta \quad (2.6)$$

$$\dot{\theta} = w$$

En (2.6) se obtiene el modelo del robot con configuración diferencial que se utilizará en el desarrollo de este trabajo. Expresando en forma matricial, se obtiene la matriz

de rotación, ya que en todo momento el robot no va a estar alineado con el eje Y global.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\sin \theta & 0 \\ \cos \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (2.7)$$

Sustituyendo los valores de las velocidades lineales y de rotación:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\sin \theta & 0 \\ \cos \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{r(w_d+w_i)}{2} \\ \frac{r(w_d-w_i)}{L} \end{bmatrix} \quad (2.8)$$

Luego de tener el modelo de la cinemática directa del robot se pueden obtener las ecuaciones que determinen el movimiento del robot. Para esto se multiplican la matriz de rotación y el vector de velocidades del robot, obteniendo de esta manera una ecuación matricial que representa las velocidades en X, Y y angular del robot en el sistema global. Se opera en la ecuación 2.8 para obtener el modelo de una forma más clara, como se muestra en 2.9, quedando establecidas las velocidades angulares de cada rueda como entradas del sistema y las diferentes velocidades como las salidas.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\frac{r}{2} \sin \theta & -\frac{r}{2} \sin \theta \\ \frac{r}{2} \cos \theta & \frac{r}{2} \cos \theta \\ -\frac{r}{L} & \frac{r}{L} \end{bmatrix} \begin{bmatrix} w_i \\ w_d \end{bmatrix} \quad (2.9)$$

Partiendo del modelo obtenido se realiza una simulación en el Simulink de Matlab para comprobar el comportamiento ante diferentes situaciones. El modelo queda representado en la Figura 2.3.

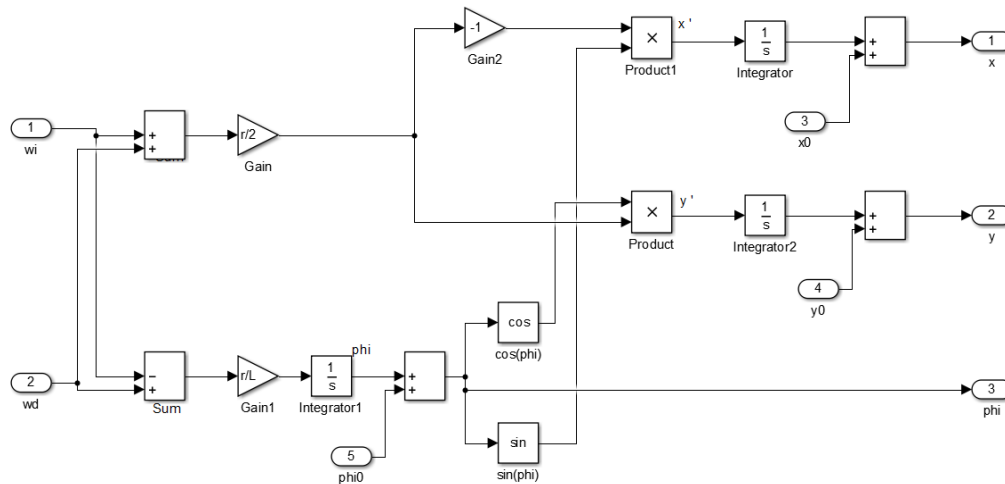


Figura 2.3. Esquema del Modelo Cinemático del robot diferencial en el Simulink del Matlab.

Ante diferentes valores de velocidades en las ruedas del robot diferencial, se obtienen comportamientos esperados. Se realiza una primera simulación con valores de velocidades iguales y en el mismo sentido para ambas ruedas a lo que el robot responde desplazándose en línea recta (figura 2.4), moviéndose a lo largo del eje Y, sin desplazamiento en X y sin alterar su orientación.

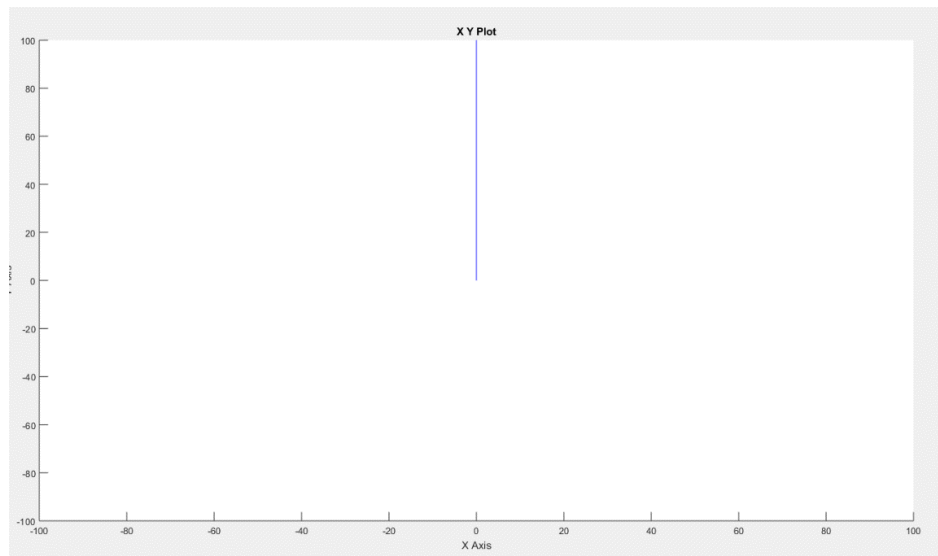


Figura 2.4. Respuesta del modelo del robot ante velocidades iguales en el mismo sentido.

Manteniendo los valores de velocidad, pero cambiando el sentido de una de ellas, obtenemos el comportamiento reflejado en la figura 2.5.

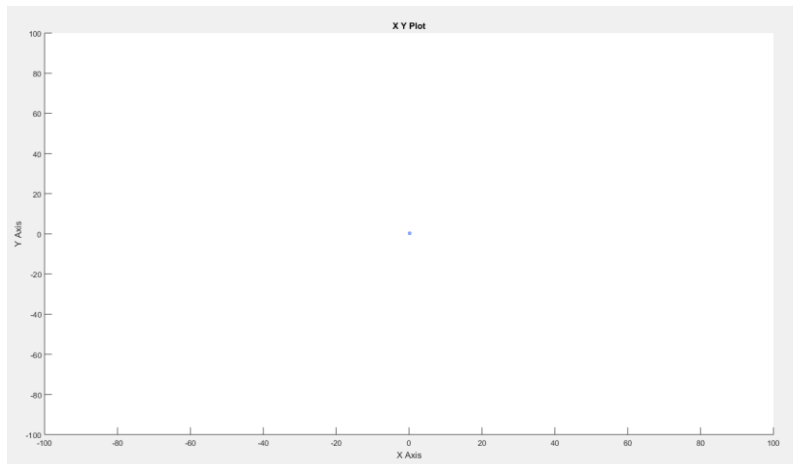


Figura 2.5. Respuesta del modelo del robot ante velocidades iguales en sentido contrario.

Esto es debido a que al estar cada rueda girando en sentidos diferentes el robot rota sobre su eje. Manteniendo su posición y variando su orientación. Seguido se realiza una última prueba, esta consiste en mantener velocidades en un mismo sentido, pero con magnitudes diferentes, esto permite tener un movimiento rotacional con desplazamiento circular. La respuesta obtenida se muestra en la

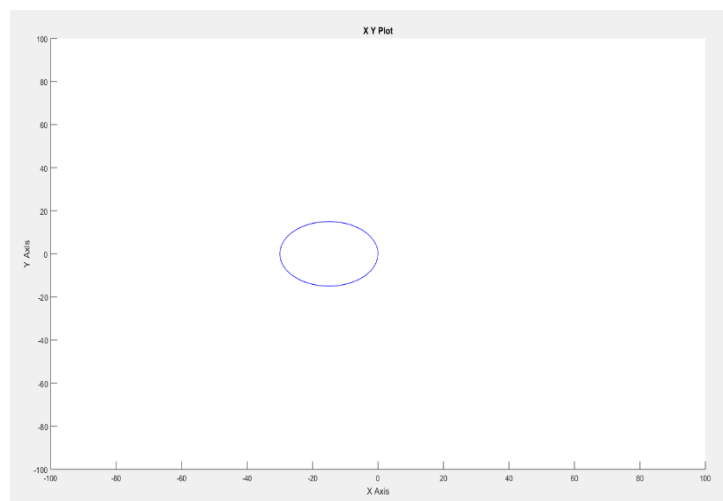


Figura 2.6. Respuesta del modelo ante velocidades diferentes en las ruedas y en el mismo sentido.

De manera similar se simuló, pero con una relación menor que el caso anterior entre las magnitudes de las velocidades, es de esperar que la amplitud del arco que genera la trayectoria cambie. En este caso el círculo que describe la trayectoria es más amplia (figura 2.7).

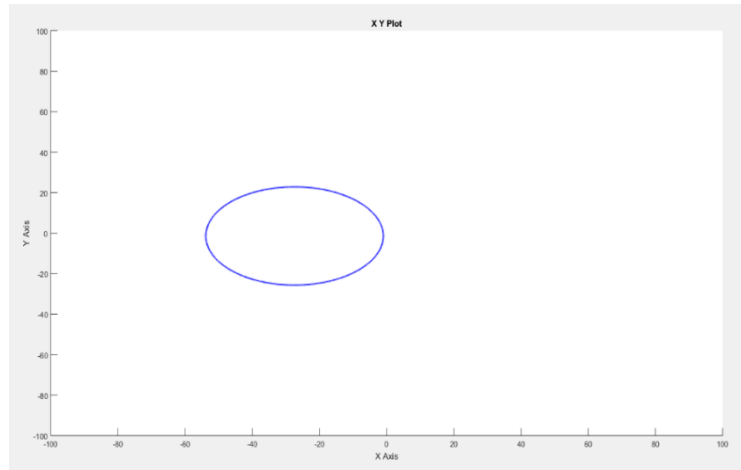


Figura 2.7. Respuesta del modelo ante velocidades diferentes en las ruedas y en el mismo sentido, con relaciones entre estas diferente.

Validado el modelo mediante las simulaciones realizadas con el Simulink de Matlab, se realizará un método para la generación de trayectorias evadiendo los obstáculos presentes en el camino.

2.3 Método de Campos Potenciales Artificiales.

El método utilizado para el planeamiento de trayectorias es el de campos potenciales artificiales, que se auxilia del uso de visión por computadora para obtener una representación aérea del plano de trabajo.

Este método trata al robot representado como una partícula sometida a diferentes fuerzas [25]. Por un lado, existe una fuerza de atracción hacia la configuración destino, y por otro una serie de fuerzas de repulsión destinadas a evitar la colisión con los obstáculos. Para definir estas fuerzas se considera un campo de potencial asociado al escenario, cuyo valor depende de su proximidad a los obstáculos y de la distancia al punto de destino (figura 2.8).

El planificador genera una trayectoria en virtud de un vector gradiente derivado del campo potencial. Se calcula dicho vector en el origen y se agrega a la trayectoria un segmento de tamaño fijo en la dirección marcada por el gradiente. En el extremo de dicho segmento (final de la trayectoria actual), se vuelve a calcular el gradiente. Nuevamente se añade otro segmento en la dirección del último vector gradiente calculado. De este modo se procede iterativamente construyendo la trayectoria o bien guiando al robot continuamente según el sentido del vector gradiente.

Esta técnica permite que el robot siga siempre la dirección que minimiza el valor del campo potencial. El objetivo es alcanzar el mínimo absoluto que estará situado en la configuración destino.

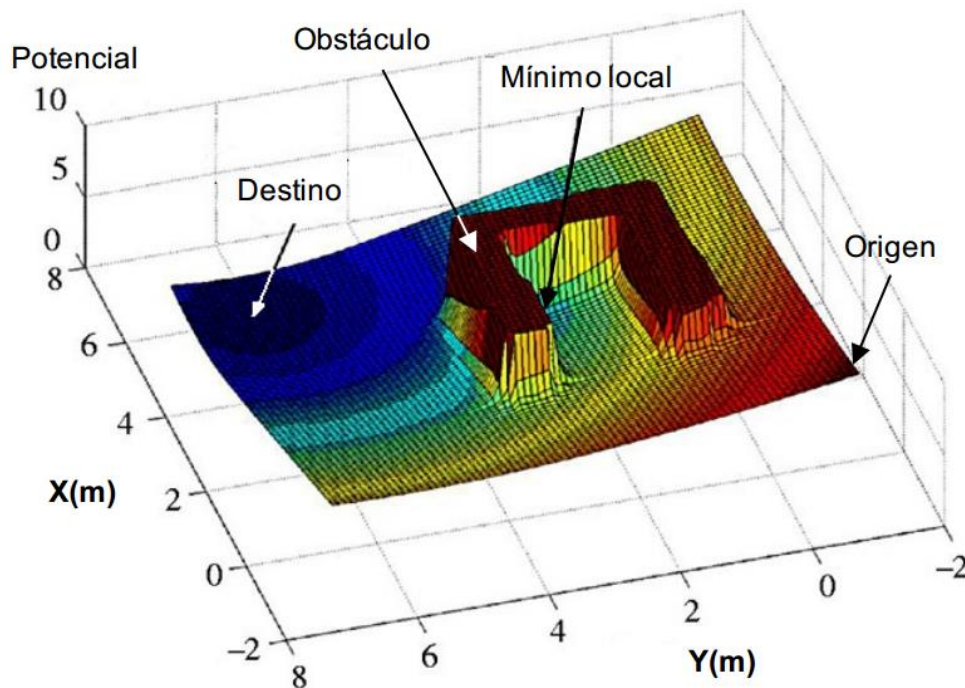


Figura 2.8. Campo de potenciales creado en un entorno.

La definición de este campo de potencial (U) puede realizarse de diversas formas. Una de ellas consiste en diseñar el campo como la suma de dos: uno que genere una fuerza de atracción sobre la configuración destino (U_{atr}), y otro que genere fuerzas de repulsión en torno a los obstáculos (U_{rep}). Así el campo de potencial en un punto $q \in C$ (definiendo a C como el plano de trabajo) resulta:

$$U(q) = U_{atr}(q) + U_{rep}(q) \quad (2.10)$$

El campo de potenciales responsable de la fuerza de atracción suele definirse como una función parabólica:

$$U_{atr} = k\rho_{goal}^2(q) \quad (2.11)$$

Donde k es un factor de escala, $\rho_{goal}(q)$ es la distancia euclídea $\|q - q_{goal}\|$ y q_{goal} es la configuración destino.

La fuerza de atracción derivada de este campo de potencial será [25]:

$$\vec{F}_{atr} = -\vec{\nabla}U_{atr}(q) = -2k * (q - q_{goal}) \quad (2.12)$$

Como puede observarse la fuerza de atracción consiste en un vector cuyo módulo es proporcional a la distancia entre el punto considerado y el objetivo, y cuya dirección apunta siempre hacia la configuración final. En cuanto al campo de repulsión suele definirse como la suma de los campos generados por cada obstáculo:

$$U_{rep}(q) = \sum_{k=1}^r U_{CB_k}(q) \quad (2.13)$$

Un ejemplo de campo de potencial de repulsión asociado a un obstáculo podría ser el siguiente [25]:

$$U_{CB_k}(q) = \begin{cases} \frac{1}{2}n \left(\frac{1}{\rho_k(q)} - \frac{1}{\rho_0} \right)^2 & si \quad \rho_k(q) \leq \rho_0 \\ 0 & si \quad \rho_k(q) > \rho_0 \end{cases} \quad (2.14)$$

Donde n es un factor de escala positivo, $\rho_k(q)$ resulta ser la distancia de q al obstáculo- C identificado con el valor k . Por ejemplo:

$$\rho_k(q) = \min_{q' \in CB_k} \|q - q'\| \quad (2.15)$$

Por último ρ_0 es una constante positiva denominada “distancia de influencia del obstáculo-C”. Así el campo de potencial repulsivo asociado al obstáculo-C correspondiente es siempre positivo o nulo, tiende a infinito conforme se acerca al obstáculo-C y se hace nulo más allá de ρ_0 . La fuerza de repulsión asociada a cada obstáculo resulta [25]:

$$\vec{F}_{CB_k}(q) = -\vec{\nabla}U_{CB_k}(q) = \begin{cases} n \left(\frac{1}{\rho_k(q)} - \frac{1}{\rho_0} \right) \frac{1}{\rho_k^2(q)} \vec{\nabla}\rho_k(q) & \text{si } \rho_k(q) \leq \rho_0 \\ 0 & \text{si } \rho_k(q) > \rho_0 \end{cases} \quad (2.16)$$

El conjunto de todas estas fuerzas de repulsión sería:

$$\vec{F}_{rep}(q) = \sum_{k=1}^r \vec{F}_{CB_k}(q) \quad (2.17)$$

Y por tanto el vehículo se moverá en función de la siguiente fuerza resultante:

$$\vec{F}_{res} = \vec{F}_{rep} + \vec{F}_{atr} \quad (2.18)$$

El planificador por tanto generará una trayectoria propia de una partícula sometida a \vec{F}_{res} .

Sin embargo, el campo resultante puede tener una serie de mínimos locales (figura 2.8). El problema surge cuando el sistema alcanza un mínimo local. En este caso la resultante de todas las fuerzas es nula, pero no es la posición que se desea alcanzar. Para resolver esto en las diferentes bibliografías se recurren a diversos métodos, entre ellos los probabilísticos, los cuales presentan algunos inconvenientes, como son el costo computacional y el tiempo que se emplea en su desarrollo. Como variante, la solución que adopta en este trabajo consiste en eliminar los mínimos locales del plano utilizando el procesamiento de imágenes, modificando la forma de los obstáculos que puedan representar en el plano un mínimo local. Esta modificación se realiza sin comprometer la región que ocupa el obstáculo, para no crear falsas áreas libres de obstáculos.

2.3.1 Implementación del algoritmo.

La implementación del algoritmo para la generación de trayectorias se realiza utilizando Matlab®, en su versión 2015a. El algoritmo básicamente consiste en 4 etapas, las cuales se describen en la figura 2.9. Las etapas brindan una mejor comprensión del flujo del programa, estas deben realizarse en el orden establecido ya que cada una depende de la anterior. Como resultado se debe contar con un vector en el cual están descritas las coordenadas que utilizará el seguidor para ir guiando al robot móvil sobre el plano.

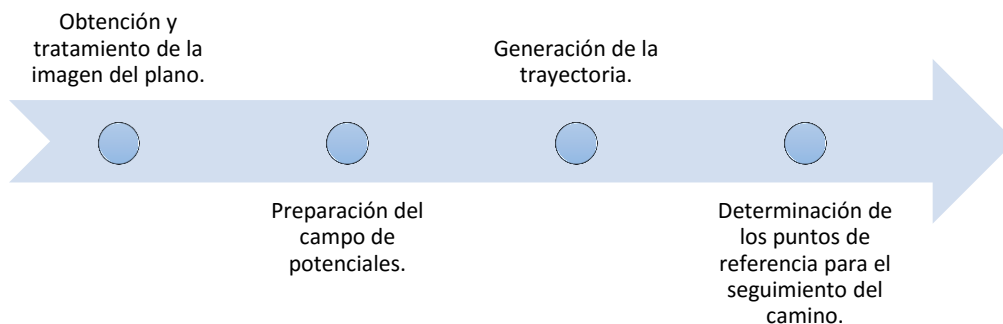


Figura 2.9. Etapas de implementación del algoritmo.

2.3.2 Obtención y tratamiento de la imagen del plano.

Como se planteó anteriormente, el algoritmo hace uso de la visión por computadora para obtener la información del plano sobre el cual se desplazará el vehículo. La imagen es captada por una cámara (ver anexo 1) incorporada a la computadora donde se ejecuta el Matlab®, esto es posible gracias al toolbox de adquisición de imágenes que incorpora el software de la MathWorks®.

Previamente configurado, el tamaño de la imagen capturada se fija en 640x480 píxeles, esto nos proporciona una imagen digital de 640 píxeles de ancho y 480 píxeles de alto. La selección de la resolución obedece a que es la máxima que brinda la cámara, aunque una imagen de mayor resolución siempre brinda más información, para la aplicación en cuestión la resolución seleccionada es suficiente.

Estableciendo un espacio de trabajo de 4x3 metros (tamaño de una habitación promedio), se obtiene para la resolución fijada, que cada pixel representaría un cuadrado de 6.25mm de lado, como se muestra en la figura 2.10. O sea que por cada pixel que se recorra el vehículo se desplaza 6.25mm en el plano. Estos datos y el tamaño del vehículo son factores a tener en cuenta para evitar colisiones laterales del vehículo con los objetos o con los bordes del plano. También son datos que se emplean a la hora de generar las coordenadas para el seguidor. Esto se hace necesario debido a la diferencia que existe en la ubicación del origen de una imagen y de un plano cartesiano.

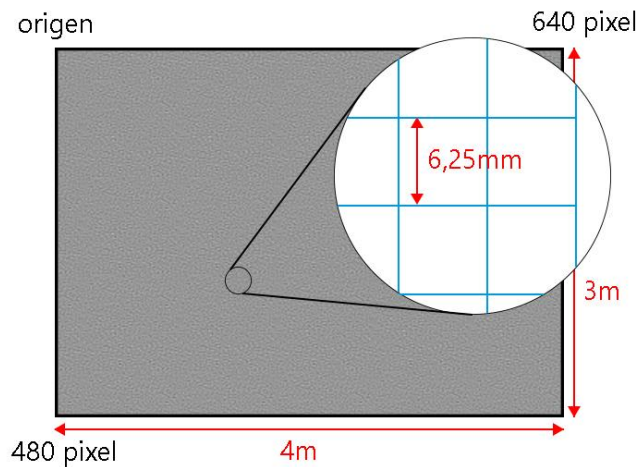


Figura 2.10. Dimensión de un pixel en el plano.

La imagen adquirida es guardada en escala de grises, ya que cualquier objeto en la imagen se considera como un obstáculo, para su detección no es necesario el reconocimiento de los colores presentes, ya que no es de interés clasificarlos sino conocer su ubicación. La condición que si debe cumplir la imagen es el color blanco del fondo, el color de los obstáculos pudiera variar ya que al convertir la imagen en un formato de escala de grises estos son presentados en tonos oscuros. Un ejemplo es el mostrado en la figura 2.11, donde la imagen resultante en escala de grises conserva la forma de cada obstáculo eliminando la información de color.

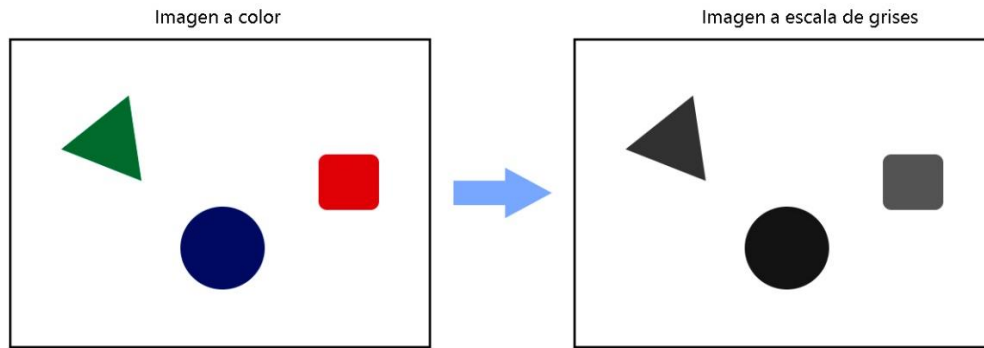


Figura 2.11. Imagen a color llevada a escala de grises.

Luego de obtener la imagen, en esta etapa se realiza un barrido el cual tiene como objetivo limpiar el fondo de áreas de poca luz o manchas claras que pudieran presentarse en el momento de la adquisición, básicamente consiste en llevar los índices de gris a sus extremos opuestos, es decir los tonos oscuros a negros y aquellos relativamente claros a blancos. Esto se hace necesario ya que el algoritmo de planeación trabajará solo con píxeles negros y blancos, o sea una imagen en blanco y negro.



Figura 2.12. Umbral seleccionado para la escala de grises.

Una vez emparejado a un mismo valor todos los índices de los diferentes obstáculos se obtiene la siguiente imagen (figura 2.13).

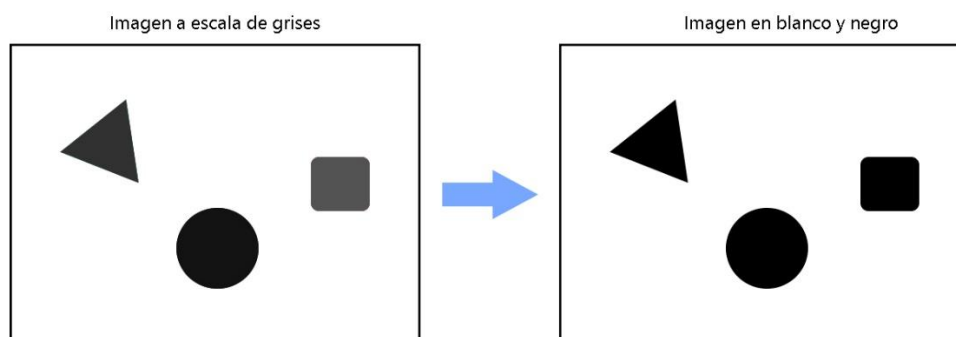


Figura 2.13. Imagen en escala de grises a la que se le aplica el umbral.

Un aspecto que afecta el método de campos potenciales artificiales es la aparición de mínimos locales, descritos en el epígrafe anterior. El tratamiento propuesto en el trabajo consiste en cerrar las cavidades o ángulos que puedan llevar al robot a uno de estos espacios donde queda atrapado (figura 2.8). Esta estrategia se completa en tres pasos, los cuales hacen uso de las funciones *bwconvhull* y *imdilate* del Matlab®.

La función *bwconvhull* cubre al objeto al que se le aplique con una región que no deja en el ángulos cóncavos, útil para eliminar cavidades que pudieran representar mínimos locales, ejemplo en la figura 2.14. Esta función puede aplicarse a varios objetos y tratarlos como un todo o bien a cada uno por separado. En este caso se tratan los objetos individualmente.

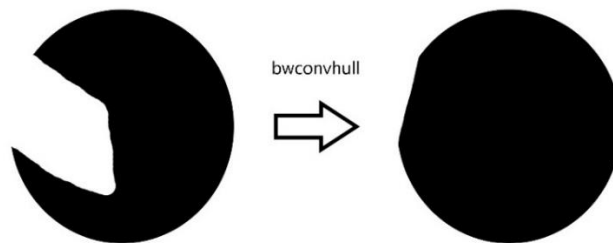


Figura 2.14. Función bwconvhull aplicada a un objeto.

Por otra parte, la función *imdilate* realiza la dilatación de los objetos presentes, esta es capaz de dilatar los objetos a los cuales se les aplique, tal que estos aumentan su tamaño, pero siguen conservando su forma (figura 2.15).

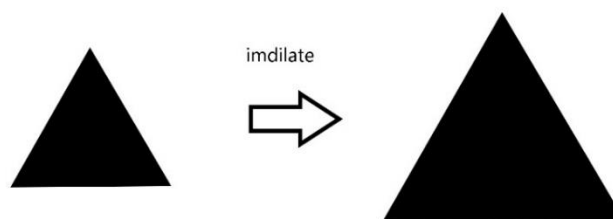


Figura 2.15. Función imdilate aplicada a un objeto.

La combinación de estas dos funciones y la manera en la que se aplican brindan una vía de solución al problema de los mínimos locales. Para esto primero se cubren los posibles ángulos o regiones cóncavas de los objetos en la imagen con una primera aplicación de la función *bwconvhull*, luego los objetos son dilatados 10 píxeles en todos los sentidos, esto se hace para brindar un margen entre la trayectoria y el obstáculo, se establece ese valor de dilatación porque es el que satisface las dimensiones del vehículo que se utiliza. Luego de la dilatación aparece un posible problema, el cual es la unión que pudiera existir entre dos objetos anteriormente separados que debido a su cercanía se unen al ser dilatados, para esto se realiza una segunda aplicación de la función *bwconvhull*, que garantiza la eliminación de cualquier cavidad o ángulo cóncavo resultante de la posible unión entre dos obstáculos. Una vez realizados estos tres pasos la imagen está en condiciones para el uso en el planeador de la trayectoria.

En la siguiente imagen se muestra un ejemplo en el cual los mínimos locales presentes en el plano son suprimidos utilizando la solución antes propuesta.

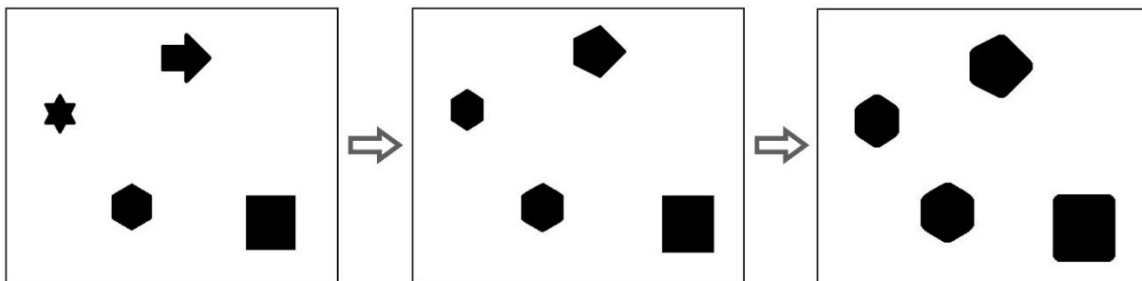


Figura 2.16. Supresión de mínimos locales utilizando bwconvhull y imdilate.

La necesidad de realizar una segunda aplicación de la función *bwconvhull* se demuestra en la siguiente figura, en la cual los obstáculos al ser dilatados se unen formando un nuevo obstáculo con posibles mínimos locales.

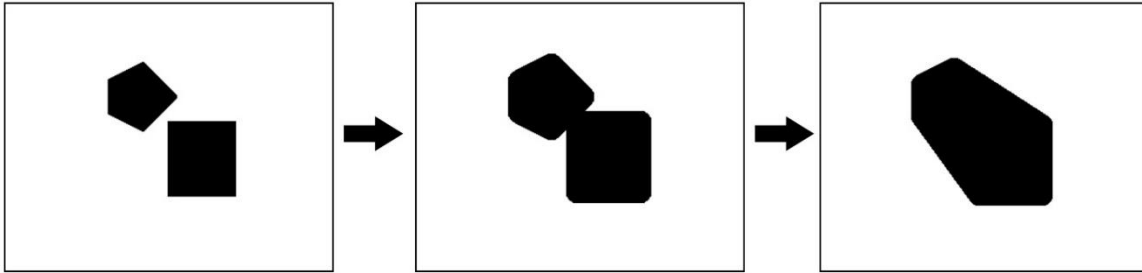


Figura 2.17. Segunda aplicación de la función *bwconvhull*.

Es preciso resaltar que la solución antes expuesta no asegura en un cien por ciento un plano sin mínimos locales, ya que estos pueden aparecer de la unión entre un obstáculo y el borde del plano, caso en el cual las funciones quedan sin efecto, debido a que el obstáculo en esta situación no presenta ángulos cóncavos (figura 2.18). Por lo que las regiones definidas como *m* y *m1* quedan como posibles mínimos locales en la trayectoria del vehículo.

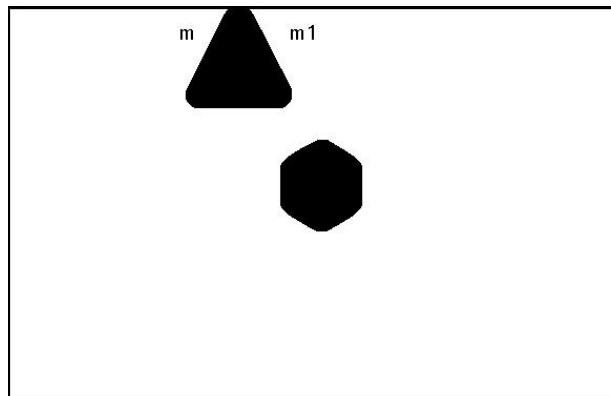


Figura 2.18. Mínimo local formado entre el borde del plano y un obstáculo.

2.3.3 Preparación del campo de potenciales.

El método trata el ambiente de trabajo como un campo de fuerzas, dando un valor diferente a cada objeto sobre el área de trabajo. El objetivo o punto de llegada se considera como un imán de polaridad contraria a la del vehículo, es decir que tiene fuerza de atracción mientras que los obstáculos se comportan como imanes de igual polaridad generando fuerzas repulsivas. La implementación de este método se

realizó usando una matriz de discretización del terreno, de tal manera que cada punto sobre la matriz es un área del plano.

Básicamente la planificación se puede trabajar de la siguiente manera:

- Inicializar variables.
- Leer y ubicar puntos importantes del terreno sobre la matriz.
- Definir potencial para el punto de llegada.
- Definir potencial para cada obstáculo.
- Mover el móvil desde el punto de inicio al punto de llegada siguiendo el potencial menor o igual y guardar ese movimiento hasta encontrar el punto de llegada.

Para lo cual es necesario tener las siguientes consideraciones:

- Existe una discretización del terreno.
- Cada punto del plano se considera afectado por un valor numérico que indica la relación de este con el punto de llegada. De otra manera cada celda considerada posee un potencial que indica si la trayectoria puede o no pasar por allí.
- De acuerdo con la distribución numérica asignada, la trayectoria se hace siguiendo un potencial descendente
- El algoritmo es ajustable al área de trabajo. Con esto se trata de generar algo de robustez lo cual permite usar el mismo programa para diferentes tamaños de plano. En este caso el área de trabajo quedaría definida por las características del sistema de visión tales como la altura de la cámara y lente usados.
- El algoritmo trabaja off-line, ya que la información de la ubicación de los obstáculos se pone al principio del programa, con lo cual sólo se encuentra un camino sin poder revisar si algún obstáculo ha entrado a la trayectoria planeada.

2.3.3.2 Creación del potencial alrededor del punto de llegada.

Para poder generar un potencial que haga converger todos los caminos al punto de llegada, se le asigna a este punto un valor numérico igual a “0” y a partir de este se generan valores numéricos ascendentes hasta cubrir la totalidad del plano, como se observa en la figura 2.20. Hasta ahora el único punto que queda sin potencial es el de inicio, el cual se representa con un valor superior al mayor potencial del plano.

4	4	4	4	4	4	4	4	4
4	3	3	3	3	3	3	3	4
4	3	2	2	2	2	2	3	4
4	3	2	1	1	1	2	3	4
4	3	2	1	0	1	2	3	4
4	3	2	1	1	1	2	3	4
4	3	2	2	2	2	2	3	4
4	3	3	3	3	3	3	3	4
4	4	4	4	4	4	4	4	4

Figura 2.20. Potencial alrededor del punto de llegada.

Con lo que se asegura que desde cualquier punto siguiendo una secuencia descendente de números se converge al punto de llegada.

2.3.3.3 Creación del potencial alrededor de los obstáculos.

Para incluir los obstáculos es necesario tener en cuenta las siguientes características:

- El obstáculo posee un área efectiva propia de sus dimensiones, a través de la cual no se puede realizar una trayectoria porque habría colisión. Éste se representa en el campo con un índice mucho mayor que el máximo alcanzado por el potencial del campo generado.
- Los obstáculos son calcados en el plano en lugar de representarse por un punto. Esto posibilita una representación exacta de su forma además de ser

lo que posibilita utilizar el método antes expuesto para suprimir mínimos locales.

- Los obstáculos muy dimensionados pueden llevar al fallo del algoritmo, pues podría ser incapaz de bordearlo, además de poder provocar mínimos locales con los bordes del plano.

En algunas literaturas los obstáculos son representados como puntos en el plano que son envueltos por un potencial descendente (figura 2.21a) para contrarrestar el inconveniente que puede representar el que estos tengan forma irregular diferente al área de discretización, esto impediría que una trayectoria fuera descrita en su cercanía por considerarse esta una zona de riesgo. Esta técnica no es necesaria con la implementación realizada la cual considera la forma exacta del obstáculo en el momento de asignar el valor a las celdas que lo componen (figura 2.21b). A su vez el uso de un campo descendente es remplazado por el efecto que produce la función *imdilate* en los bordes del obstáculo, simulando con esto la fuerza repulsora.

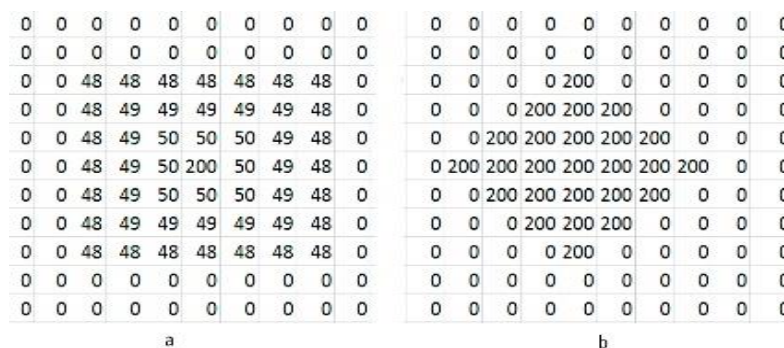


Figura 2.21. Campo de potencial alrededor de un obstáculo.

Una vez colocados los obstáculos es necesario establecer los límites del campo en el plano, esto se realiza asignando un valor superior al máximo encontrado en el plano, pero al mismo tiempo inferior al usado para los obstáculos. Esta es una forma de asegurar que el móvil nunca salga del área de trabajo y corresponde más a un método de seguridad que de necesidad, ya que, si el robot posee sensores de proximidad, estos pueden evitar las colisiones con las paredes de la orilla del plano.

Un ejemplo de un campo de potenciales con los obstáculos representados se muestra en la siguiente figura (figura 2.22).

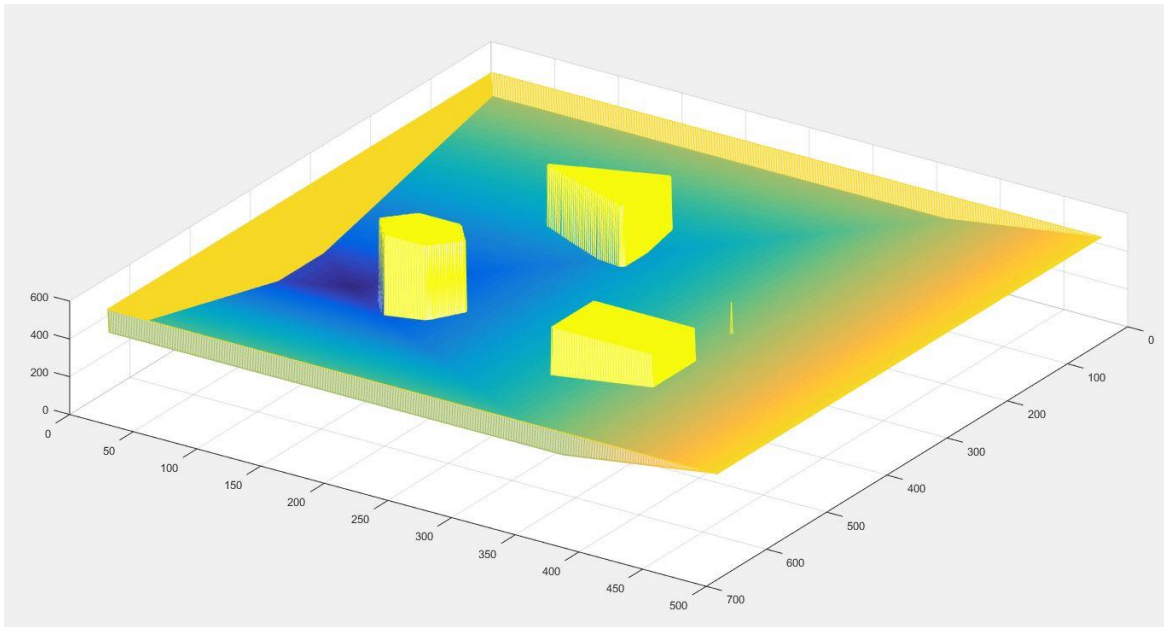


Figura 2.22. Vista general del campo de potenciales.

2.3.4 Generación de la trayectoria.

Para la generación de la trayectoria el programa se va desplazando por las celdas de la matriz de campo siguiendo los índices de menor valor en sus celdas vecinas. El movimiento se realiza de celda a celda y el ciclo de encuesta sigue el patrón presentado en la figura 2.23, esto contenido en un ciclo hasta que el valor de la celda corresponda a '0', lo que significa que se ha llegado al destino.

En caso de encontrarse una iteración en la cual no se obtenga un valor menor al actual el programa buscará en su lugar uno equivalente y se moverá a esa celda en la cual volverá a preguntar por un índice menor. Esto es útil en el caso que el obstáculo bloquee el camino y no quede más remedio que bordearlo, caso que aparece cuando la trayectoria se encuentra con el obstáculo perpendicularmente.

A continuación, algunos ejemplos de trayectorias generadas usando el método antes explicado. Los puntos I y F, corresponden al inicio y final de la trayectoria respectivamente.

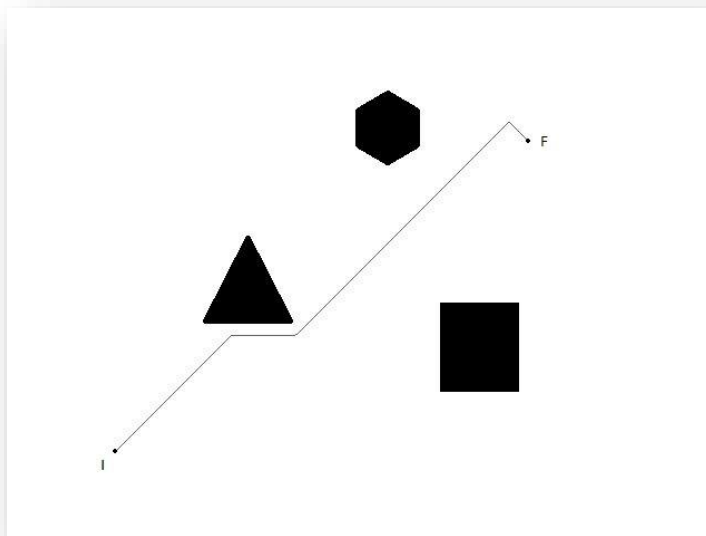


Figura 2.23. Trayectoria generada utilizando el método de campos potenciales artificiales.

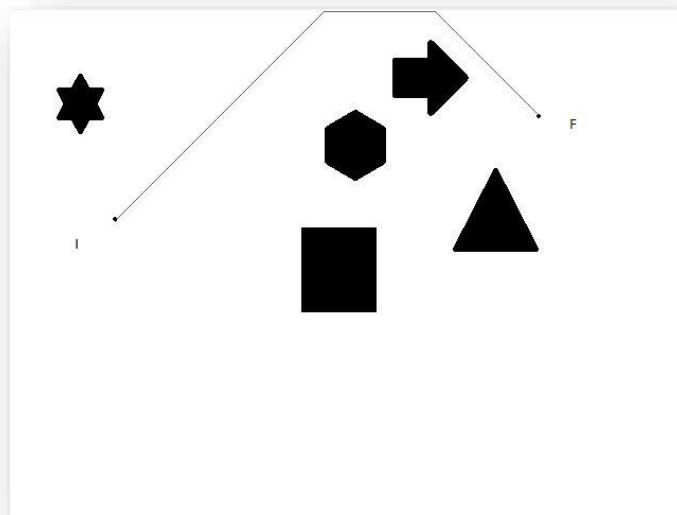


Figura 2.24. Trayectoria generada utilizando el método de campos potenciales artificiales.

Del análisis de las trayectorias logradas se destaca un problema de optimización sobre todo en lograr un camino más directo al objetivo y por ende más corto. En la figura 2.23 se llega al punto objetivo, pero con un cambio brusco en la orientación cerca del destino, al igual que en la figura 2.24 en la cual el camino trazado es sin lugar a dudas el menos directo.

Este resultado se debe al orden en que el programa analiza las celdas vecinas a su posición actual. El orden establecido de esta encuesta se realiza comenzando por la celda superior (fil-1,col), y continua su recorrido según la figura 2.25 hasta encontrar un valor menor para desplazarse a esa celda.

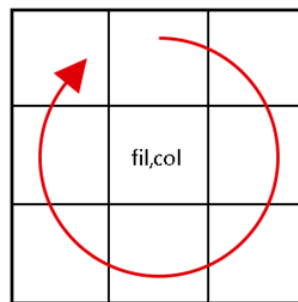


Figura 2.25. Orden de encuesta de las celdas vecinas.

Esto ocasiona que exista una cierta preferencia a la hora de escoger el mejor valor en casos en los que se encuentran varios valores menores e iguales entre ellos en la vecindad de la celda actual. Por ejemplo, en la figura 2.16 la celda a la cual se desplazará el programa no es necesariamente la más cercana al destino, lo que devolvería una trayectoria más larga.

Para resolver este inconveniente se introduce un paso antes de generar la trayectoria, y consiste en conocer la orientación del inicio respecto al destino, y usar esa información para escoger la celda que mejor nos lleve al punto destino.

2.3.5 Utilizando la orientación para mejorar la trayectoria generada.

El uso de la orientación busca mejorar no sólo la trayectoria resultante sino también el costo computacional del algoritmo debido a que la cantidad de celdas a encuestar disminuye, además de lograr trayectorias más suaves y directas.

Conociendo los puntos de partida y destino en el plano, se puede conocer el ángulo formado entre la línea recta que los une y el eje de referencia X. Para lo cual se recurre al concepto de la pendiente y su relación con el ángulo mencionado.

La pendiente se define como la inclinación de un elemento ideal natural o constructivo respecto de la horizontal. El ángulo a , definido tal como aparece en la figura 2.26, se llama ángulo de inclinación de la recta respecto al eje X. La tangente trigonométrica del ángulo de inclinación a se llama coeficiente angular de la recta y se designa usualmente con la letra k , entonces:

$$k = \tan a \quad (2.19)$$

En realidad, el coeficiente angular y la pendiente tienen el mismo significado geométrico. En la ecuación $y = kx + b$ que involucra el coeficiente angular y la ordenada en el origen: k es el coeficiente angular y b la ordenada en el origen.

PENDIENTE DE LA RECTA

La pendiente de una recta en un sistema de representación rectangular (de un plano cartesiano), suele estar representada por la letra m , y está definida como la diferencia en el eje Y dividido por la diferencia en el eje X para dos puntos distintos en una recta. Esto se describe en la siguiente ecuación:

$$m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1} \quad (2.20)$$

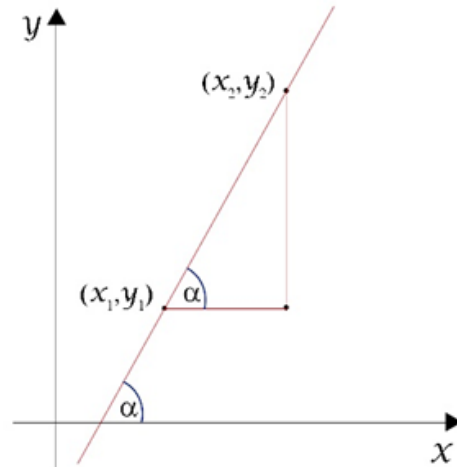


Figura 2.26. Representación de una recta en el plano cartesiano.

El ángulo a que se forma entre la recta y el eje X, se define:

$$m = \tan a \quad (2.21)$$

Equivalente a:

$$a = \tan^{-1} \frac{y_2 - y_1}{x_2 - x_1} \quad (2.22)$$

Una vez conocido el ángulo de inclinación se tiene información de la orientación a seguir por la trayectoria, esta información determina la primera celda que será encuestada en busca de los valores que conduzca al destino. Para determinarla, la vecindad de la celda origen se divide en 8 rangos que se determinan según el valor del ángulo de inclinación de la celda con mayor probabilidad de contar con el menor valor y a su vez la mejor orientada.

			<i>Angulo</i>	<i>Celda</i>
5	6	7	$338^\circ - 22^\circ$	0
			$23^\circ - 67^\circ$	7
			$68^\circ - 112^\circ$	6
4	●	0	$113^\circ - 157^\circ$	5
			$158^\circ - 202^\circ$	4
			$203^\circ - 247^\circ$	3
3	2	1	$248^\circ - 292^\circ$	2
			$293^\circ - 337^\circ$	1

Figura 2.27. Rangos en los que se divide la 8 vecindad.

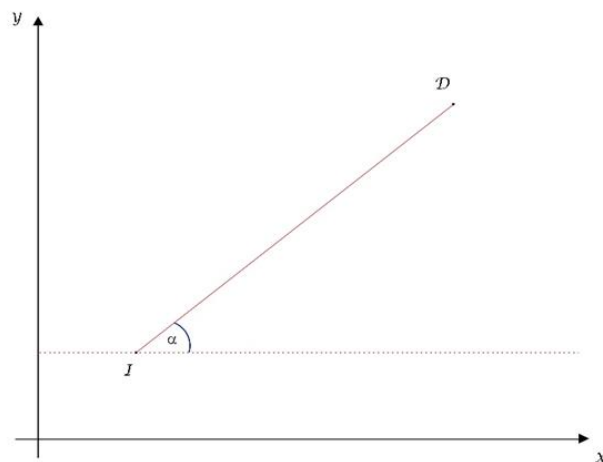


Figura 2.28. Ejemplo de orientación calculada en el plano.

El cálculo de esta orientación no se realiza una sola vez en el programa, por el contrario, esta se recalcula cada 20 iteraciones del algoritmo, para mantener actualizado la orientación en casos que puedan ocurrir desvíos producto de la evasión de algún obstáculo. Casos en los cuales la orientación cambia, variando la primera celda a ser encuestada. Con este punto añadido, en las siguientes figuras 2.29 y 2.30, se muestran los resultados obtenidos al aplicarlo en los ejemplos anteriores presentados en las figuras 2.23 y 2.24.

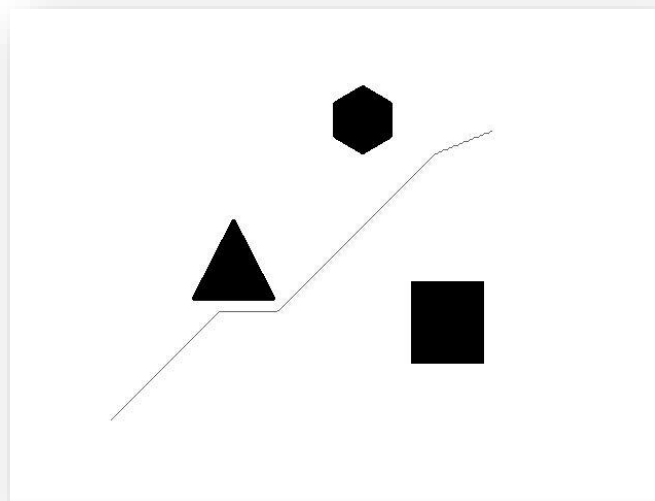


Figura 2.29. Trayectoria generada añadiendo el uso de la orientación en el programa.

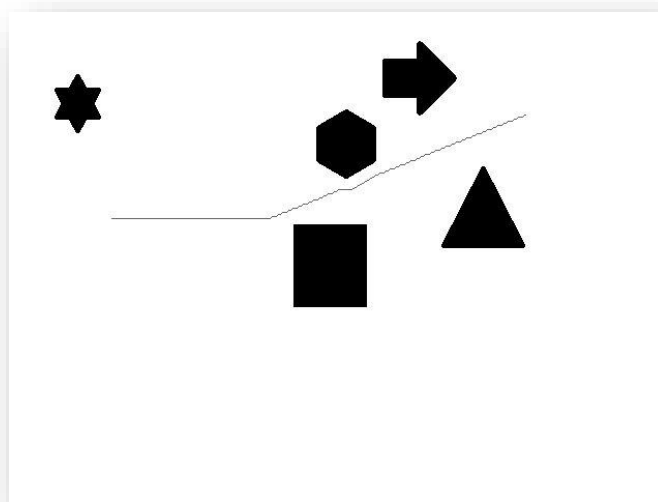


Figura 2.30. Trayectoria generada añadiendo el uso de la orientación en el programa.

La propuesta de utilizar la orientación para lograr trayectorias más efectivas demuestra su validez en los ejemplos antes mostrados, en los cuales las trayectorias muestran mejores características, la llegada al destino se hace de forma más suave sin cambios bruscos en la orientación, que asegura trayectorias más

fácilmente seguidas por los robots móviles. Es evidente también en el segundo ejemplo el apreciar una trayectoria más corta y fácilmente reproducible.

El algoritmo va generando el código de cadenas al mismo tiempo que genera la trayectoria. Este código de cadenas se utilizará para determinar las referencias que necesitará el seguidor de trayectorias.

2.4 Seguidor de trayectorias.

El uso de un seguidor de trayectorias tiene como objetivo validar hasta cierto punto cuan realizables son las trayectorias logradas por un robot móvil. Para lo cual se utilizó el toolbox de robótica del Matlab®, específicamente el algoritmo de seguimiento de trayectorias PurePursuit. Éste computa la velocidad angular que moverá el robot desde su posición actual hasta alcanzar el punto de mira hacia adelante del robot, definido por el parámetro LookAheadDistance. La velocidad lineal se asume constante, aunque con posibilidad de modificarse. Este seguidor no necesita información de cada uno de los puntos que pertenecen a la trayectoria, utiliza solo puntos característicos (waypoints) como puntos de referencia para ir desplazándose.

El algoritmo no es un controlador tradicional, pero actúa como un seguidor para propósitos de seguimiento de caminos, sus parámetros lo definen los datos proporcionados según las especificaciones del robot móvil. Dadas la posición y orientación del robot como entradas, la función step puede ser usada para calcular las velocidades lineal y angular del robot.

La propiedad LookAheadDistance (distancia de mira hacia adelante) es la propiedad de sintonía principal para el controlador PurePursuit. La distancia de mira hacia adelante significa hasta qué punto a lo largo del camino el robot debe desplazarse desde la ubicación actual, para calcular los comandos de velocidad angular. La siguiente figura muestra un ejemplo de un robot y el punto de mira hacia adelante (Look Ahead).

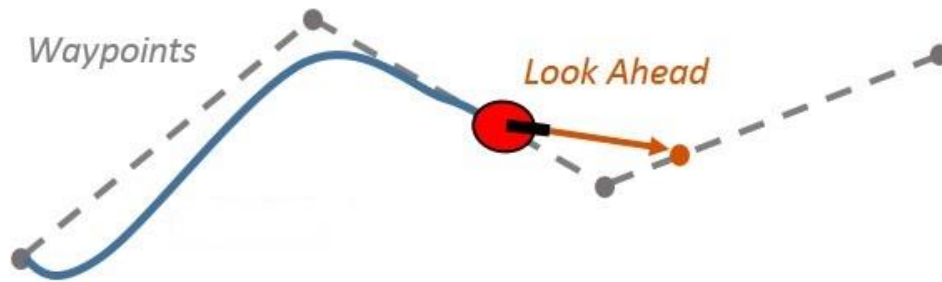


Figura 2.31. Distancia de mira hacia adelante (Look Ahead)

El efecto del cambio de este parámetro puede variar la forma en que el robot sigue el camino. Existiendo dos objetivos principales: recuperar el camino y el mantenimiento de la vía. Con el fin de recuperar rápidamente la ruta entre waypoints, una pequeña distancia de mira hacia adelante hará que el robot se mueva rápidamente hacia el camino. Sin embargo, como puede verse en la siguiente figura 2.32, el robot sobrepasa la ruta y oscila a lo largo de la trayectoria deseada. A fin de reducir las oscilaciones a lo largo de la trayectoria, se puede elegir una distancia de mira hacia adelante más grande, sin embargo, podría resultar en curvaturas más grandes cerca de las esquinas.

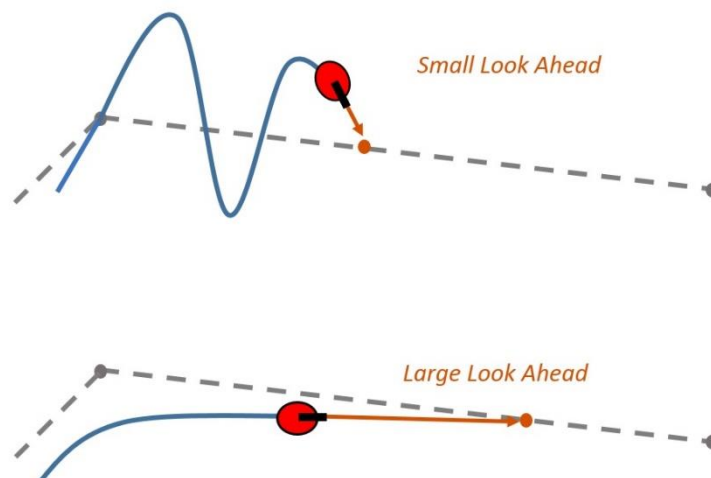


Figura 2.32. Efectos de cambios en el valor de mira hacia adelante.

La propiedad LookAheadDistance debe ser sintonizado para su aplicación y según las características del robot. Diferentes velocidades lineales y angulares afectarán

a esta respuesta, debiendo ser considerados para el controlador de seguimiento de camino.

Limitaciones del método de persecución pura.

Hay algunas limitaciones a tener en cuenta acerca de este algoritmo de persecución pura:

- Como puede observarse, el controlador no puede seguir exactamente trayectorias directas entre waypoints. Los parámetros deben estar sintonizados para optimizar el rendimiento y para que converja a la trayectoria en el tiempo.
- El algoritmo PurePursuit no estabiliza el robot en un punto. En su aplicación, un umbral de distancia para una ubicación objetivo debe ser aplicado, con el objetivo de detener el robot cerca de la meta deseada.

2.5 Análisis de resultados.

Una vez obtenidos los resultados del método implementado, estos se comparan con los obtenidos utilizando otra variación del campo potencial, la cual consiste en integrar a este la teoría de enjambres de partículas brownianas [9], un método probabilístico. El método consiste en una solución robusta al problema de mínimos locales. Se analizan dos casos con tipos de obstáculos diferentes.

En el primer caso, con dos obstáculos en el plano el algoritmo implementado obtiene una buena respuesta la cual es en gran medida superior a la obtenida en [9]. En la cual se observa que el método planteado en [9] es más susceptible a la cercanía de los campos de potenciales generados por los obstáculos. Lo que afecta el curso estable de la trayectoria a través de estos.

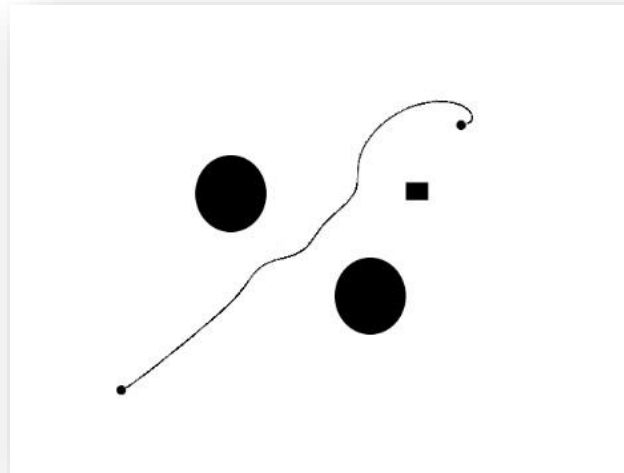


Figura 2.33. Respuesta obtenida utilizando el método planteado en [9].

Por otra parte, los resultados de utilizar en las mismas condiciones el método implementado en el trabajo arroja la siguiente respuesta, según muestra la figura

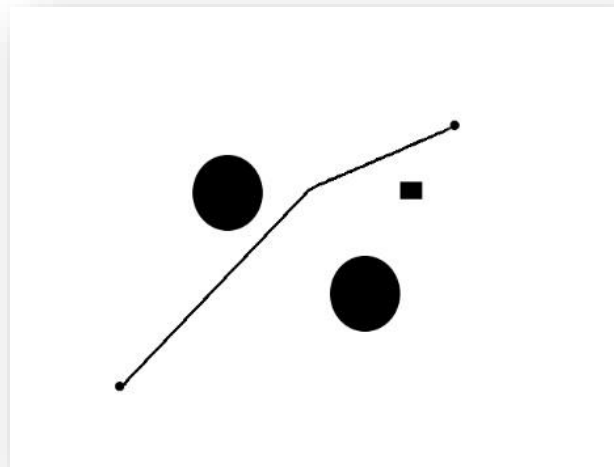


Figura 2.34. Respuesta obtenida utilizando el método propuesto.

Se muestra una mayor inmunidad al efecto de los potenciales que no representan riesgo de colisión con el obstáculo. Por consecuencia el camino tiende a ser más directo y por lo tanto más corto.

Para el segundo caso, el objeto considerado es de tipo barrera (figura 2.35), el método planteado en [9] logra vencer el mínimo local gracias al uso de las partículas autopropulsadas brownianas.

El método propuesto en el trabajo falla al bordear el obstáculo debido al tamaño de este (figura 2.36).

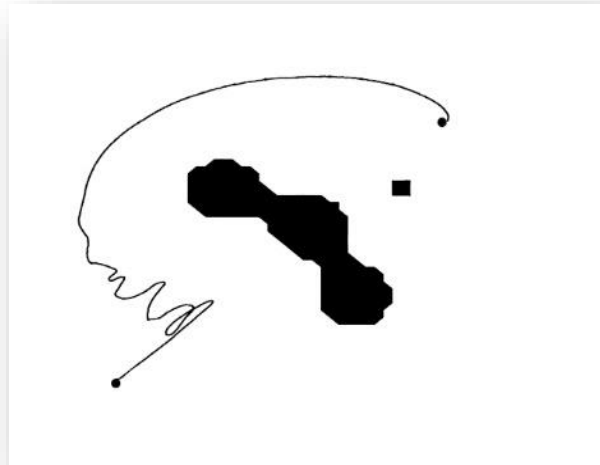


Figura 2.35. Respuesta obtenida utilizando el método planteado en [9].

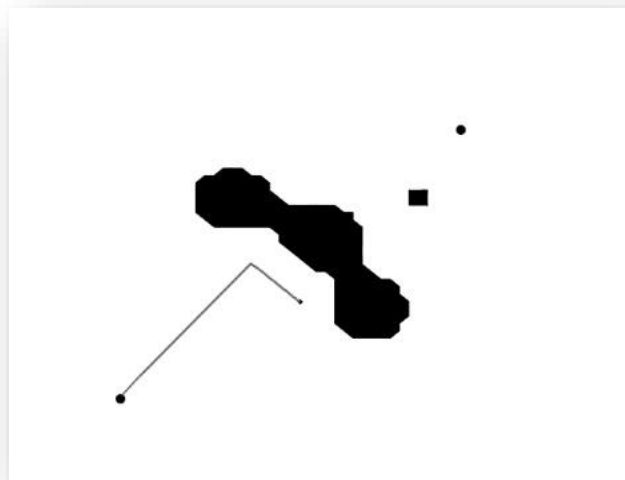


Figura 2.36. Respuesta obtenida utilizando el método propuesto.

Ante el fallo del algoritmo se analizaron dos casos, uno disminuyendo el tamaño del objeto tipo barrera, y otro modificando el punto de inicio de la trayectoria. Para ambos casos el algoritmo responde satisfactoriamente, logrando evadir el obstáculo con una trayectoria rápida y suave. Ver figuras 2.37 y 2.38.

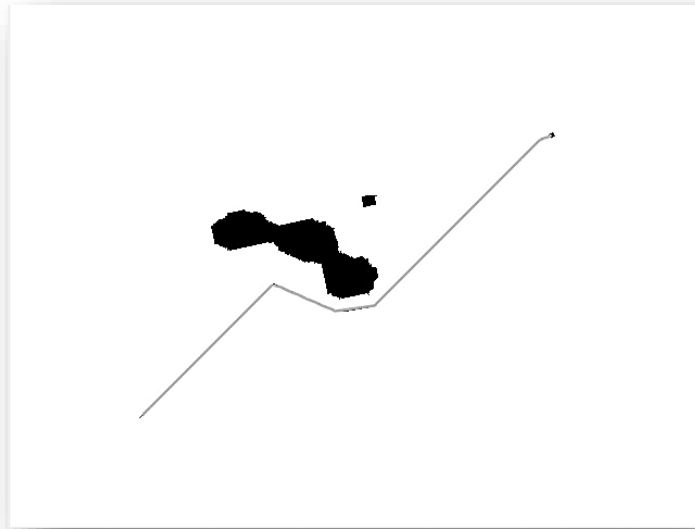


Figura 2.37. Trayectoria obtenida disminuyendo el tamaño de la barrera.

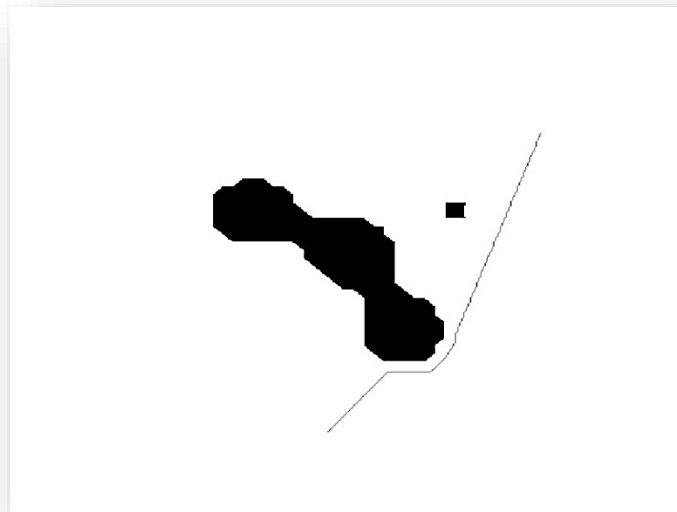


Figura 2.38. Trayectoria obtenida modificando el punto de inicio de la trayectoria.

2.6 Conclusiones del capítulo

1. Se obtuvo el modelo cinemático para un robot móvil de configuración diferencial, validado mediante simulación.
2. Se desarrolló un algoritmo para la generación de trayectorias, basado en campos potenciales, utilizando visión.
3. Se utilizaron técnicas de procesamiento de imágenes para obtener los waypoints de la trayectoria como referencias al algoritmo de control de persecución pura.
4. El algoritmo propuesto implementa una técnica para eliminar en gran medida los mínimos locales presentes en el plano.
5. Se comparó el método desarrollado con otro reportado en la literatura, demostrando sus ventajas y desventajas.

CONCLUSIONES

En el análisis de la literatura consultada se evidenciaron las potencialidades de la visión por computadora en aplicaciones de robótica y se caracterizaron varias técnicas para la descripción de objetos presentes en las imágenes digitales.

Se realizó un estudio de las diferentes configuraciones cinemáticas en robots móviles, escogiendo la configuración de tracción diferencial para el desarrollo del trabajo.

Se analizaron diferentes técnicas para la generación de trayectorias, escogiendo la basada en campos potenciales artificiales.

El método desarrollado utiliza la visión artificial para obtener una representación digital del plano, y se utilizan funciones del toolbox de procesamiento de imágenes del Matlab para eliminar los mínimos locales. El algoritmo es aplicable tanto a imágenes en blanco y negro, así como a color. Se implementa una técnica para optimizar la trayectoria generada, y al mismo tiempo se obtiene el código de cadena que permite determinar los waypoints de la trayectoria haciéndolo más rápido.

Los resultados demostraron la eficiencia del algoritmo al trabajar con imágenes en las cuales se presentan varios obstáculos. En algunos casos ventajoso sobre algunos algoritmos de mayor complejidad, pero sensible a las dimensiones de los objetos que generen campos muy grandes a su alrededor, situaciones en las cuales se pueden cambiar los puntos de inicio y final deseados para la trayectoria, con el fin de garantizar el éxito

RECOMENDACIONES

1. Seguir desarrollando el algoritmo para evitar fallas ante la presencia de objetos de grandes dimensiones en el plano.
2. Utilizar otros controladores en el método de seguimiento, pudiéndose pensar en alguno más clásico, por ejemplo, PID.
3. Dotar al método de capacidad para trabajar con obstáculos dinámicos, ya que actualmente este solo lee la información del plano al inicio.
4. Implementar el método en alguna aplicación práctica.

BIBLIOGRAFÍA

- [1] A. Ollero, A. García-Cerezo, J. Martínez, “Fuzzy supervisory path tracking of mobile robots”, *Control Engineering Practice* 2 (2) pp. 313- 319, Pergamon Press, 1994.
- [2] A. Ollero and O. Amidi, “Predictive path-tracking of mobile robots. Applications to the CMU Navlab”, *Proceedings of 5th International Conference on Advanced Robotics, Robots in Unstructured Environments (ICAR '91)*, vol. 2, pp. 1081-1086, 1991.
- [3] B. d'Andréa-Novel, G. Campion, G. Bastin, “Control of nonholonomic wheeled mobile robots by state feedback linearization”, *The International Journal of Robotics Research*, vol. 14 no. 6, pp. 543-559, 1995.
- [4] Bräunl, T., *Embedded robotics: Mobile robot design and applications with embedded systems*, (Springer-Verlag, New York, 2006).
- [5] C.S. Tzafestas and S.G. Tzafestas, “Full-state modeling, motion planning and control of mobile manipulators”, *Studies in Informatics and Control*, vol. 10 no. 2, pp 109-128, 2001.
- [6] C. Samson, “Control of chained systems. Application to path following and time-varying point-stabilization of mobile robots”, *IEEE Transactions on Robotics and Automation*, vol. 40 no. 1, pp. 64-77, 1995.
- [7] Domingo Juan, *Robótica, Apuntes para la asignatura*, Universidad de Valencia, 2001.
- [8] De Paiva, E.C., Azinheira, J.R., Bueno, S.S. (2010). “Controle de trajetoria para veiculos terrestres de exterior”. XI Congresso Brasileiro de Automatica. Bonito, MS, Brasil, Setembro 2010.

- [9] Espitia C. Helbert E. (2011). Propuesta de un algoritmo para la planeación de trayectorias de robots móviles empleando campos potenciales y enjambres de partículas activas brownianas. Universidad Nacional de Colombia.
- [10] González, R. (1987). Digital Image Processing. Second Edition. Addison-Wesley Publishing Company. Estados Unidos.
- [11] K. Inoue, K. Otsuka, M. Sugimoto, N. Murakami, "Estimation of place of Tractor and Adaptative Control Method of Autonomous Tractor using INS and GPS", In Proceedings of Biorobotics, Valencia – Spain, pp. 27-32, 1997.
- [12] Khatib O., Real-time obstacle avoidance for manipulators and mobile robots, International Journal of Robotic Research, Volume 5, 1986.
- [13] K. Park, H. Chung, J.G. Lee, "Point stabilization of mobile robots via state space exact feedback linearization", Proc of the IEEE International Conference on Robotics and Automation, Detroit, pp. 2626-2631, 1999.
- [14] M. O'Connor, T. Bell, G. Elkaim, B. Parkinson, "Automatic steering of farm vehicles using GPS", Proc. of 3 rd Int. Conference on Precision Agriculture pp. 767-778, 1996.
- [15] Oliveira, E., Schneebeli, H., Sarcinelli-Filho, M. (2007). An optical flow-based sensing system for reactive mobile robot navigation. Revista Controle & Automação, Vol.18, No.3, Julio, agosto y septiembre.
- [16] Ollero Baturone Aníbal, Robótica manipuladores y robots móviles, Marcombo, 2001
- [17] Passino, K.M., "Bacterial Foraging Optimization," Int. J. Swarm Intelligence Research, Vol. 1, No. 1, pp. 1-16, Jan.-March, 2010

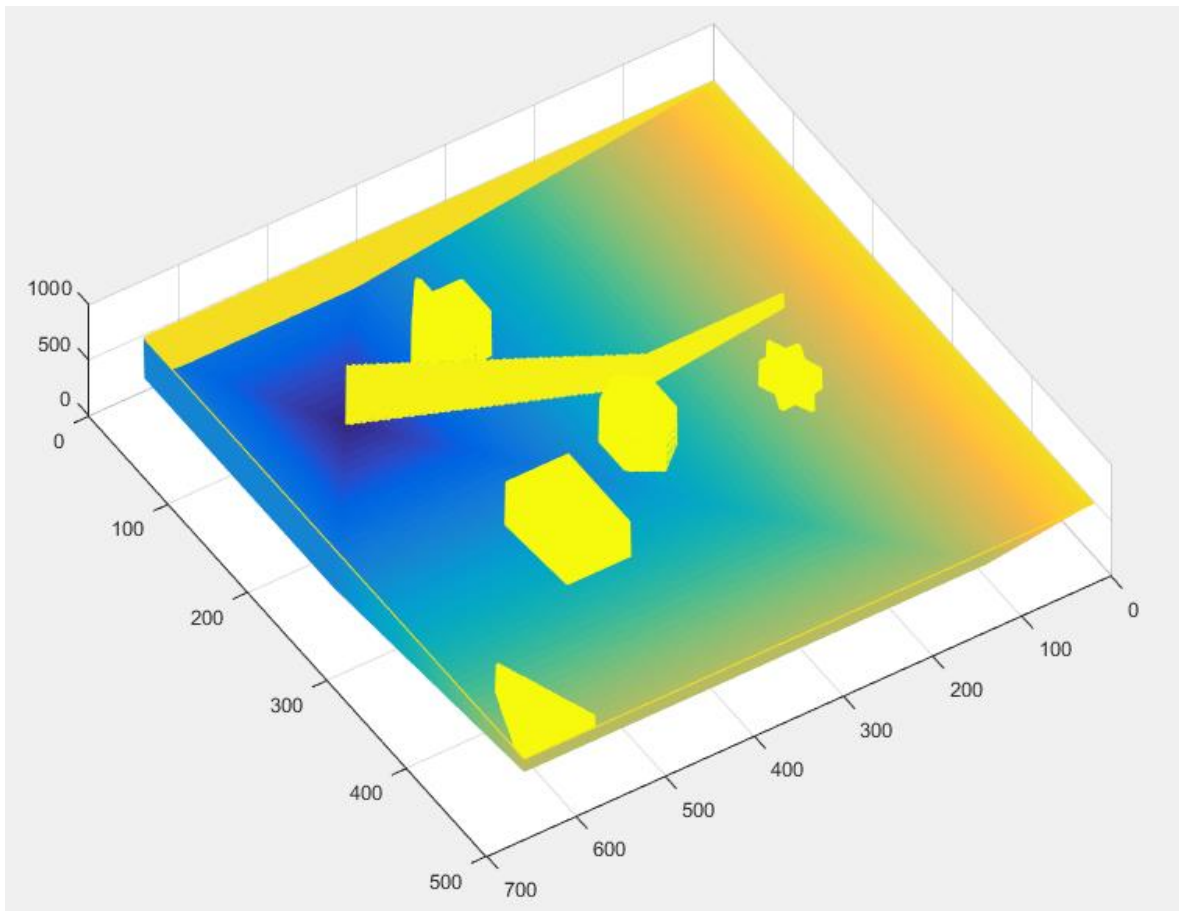
- [18] R. Hernández. “Proposta de uma plataforma de testes para o desenvolvimento de veículos autônomos”, M. Sc. Dissertação, Faculdade de Engenharia Mecânica, Univ. Estadual de Campinas, 2012.
- [19] Rajamani, R. (2006). “Vehicle Dynamics and Control”. [S.I.]: Springer, 2006.
- [20] R. M. Murray and S. Sastry, “ No holonomic motion planning: Steering using sinusoids”, IEEE Transactions on Automatic Control, vol. 38 no. 5, pp. 700-716, 1993.
- [21] S. E. Lyshevski and A. Nazarov, “Lateral maneuvering of ground vehicles: Modeling and Control”, Proc. of the American Control Conference, Chicago, Illinois, pp. 110-114, 2000.
- [22] Snider J. M. (2009). “Automatic Steering Methods for Autonomous Automobile Path Tracking”. Robotics Institute Carnegie Mellon University CMU-RI-TR-09-08.
- [23] S. Monaco and D. Normand-Cyrot, “Multirate sampling and zero dynamics: from linear to nonlinear, Nonlinear Synthesis”, C. I. Byrnes and A. Kurzhansky Eds., Birkauer, pp. 200-213, 1991.
- [24] Siegwart, R. and Nourbakhsh, R., Introduction to Autonomous Mobile Robots, (The MIT Press, Cambridge Massachusetts, 2004).
- [25] T. Fukao, H. Nakagawa, N. Adachi, “Adaptative tracking control of a nonholonomic mobile robot”, IEEE Transactions on Robotics and Automation, vol. 16 no. 5, pp. 609-615, 2000.
- [26] W.E. Dixon, D.M. Dawson, E. Zergeroglu, A. Behal, “Adaptative tracking control of a wheeled mobile robot via an uncalibrated camera system”, IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, vol. 31 no. 3, pp. 341-352, 2001.

ANEXOS

Anexo 1. Representación del plano luego de ser tratado digitalmente.



Anexo 2. Campo de potenciales generado, con la trayectoria generada.



Anexo 3. Especificaciones de la cámara utilizada para la adquisición de la imagen del plano.



Brand	Trust
Resolution	640x480
Built-in Microphone	Yes
Video recording capable	Yes
Item Weight	240 g
Product Dimensions	6.3 x 6.3 x 10.2 cm
Series	Portable Webcam WB-3100P