

## Capítulo 3

### Exploración del Ambiente Desconocido

#### 3.1 Desarrollo de algoritmos propuestos con base en los vehículos de Braitenberg

El problema que se trata en este trabajo es el de controlar un robot móvil en un ambiente desconocido con la finalidad de explorarlo y construir un mapa de él, para realizar esto, partimos del vehículo con comportamiento explorador de Braitenberg descrito en el capítulo anterior, debido a que la implementación de estas ideas es relativamente sencilla y se adapta muy bien a las características y limitaciones sensoriales del robot Khepera que cuenta con 8 sensores infrarojos de proximidad, 6 de los cuales están dispuestos de la siguiente manera: 2 sensores enfrente, 2 sensores a 45 grados, y 2 mas a 90 grados en ambos lados del robot. Los 2 restantes en la parte trasera como se muestra en la figura siguiente.

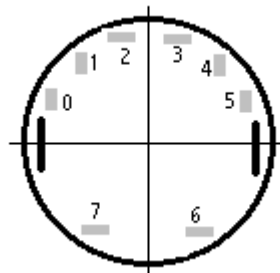


Figura 3.1 Posición de los sensores de proximidad del robot Khepera

El vehículo explorador de Braitenberg cuyo comportamiento es el de evitar obstáculos, siendo atraído por espacios libres que pueden ser conocidos o no, considera solo dos sensores en la parte delantera del agente autónomo, por lo cual la percepción del sensor izquierdo del vehículo de Braitenberg se calculó con la percepción de los sensores 0, 1 y 2 del robot Khepera, de igual manera, la percepción del sensor derecho se calculó con las aportaciones de los sensores 3, 4 y 5.

La contribución de cada uno de los sensores depende de un factor de sensibilidad que está relacionado con la posición de éstos, lo cual nos permite modificar el comportamiento del robot en el sentido de que puede tener mayor o menor sensibilidad y reaccionar de manera diferente ante obstáculos encontrados al frente o a un lado.

Las siguientes fórmulas describen el comportamiento de los motores izquierdo y derecho del robot Khepera con el comportamiento explorador de Braitenberg.

$$(3.1) \text{ motor\_derecho} = \text{velocidad de crucero} + \text{contribución sensor 0} + \text{contribución sensor 1} + \text{contribución sensor 2}$$

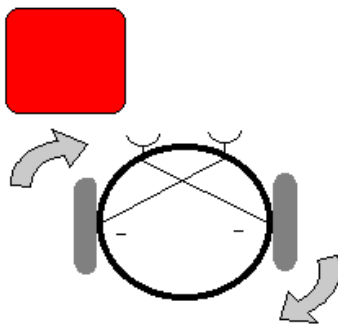
$$(3.2) \text{ motor\_izquierdo} = \text{velocidad de crucero} + \text{contribución sensor 3} + \text{contribución sensor 4} + \text{contribución sensor 5}$$

El máximo valor proporcionado por los sensores en el robot Khepera es de 1024 y el mínimo es 0, la contribución de un sensor determinado, por ejemplo el sensor 1, es igual a la percepción del sensor 1 multiplicado por un factor, en este caso se escogió como factor 1/1000, todo esto multiplicado por la sensibilidad asociada a ese sensor. Los factores de sensibilidad pueden variar, pero para el caso del comportamiento explorador deberán ser negativos, por lo que si el robot percibe un obstáculo a un lado de él, girará al lado contrario con una velocidad proporcional a lo percibido por los sensores próximos al obstáculo.

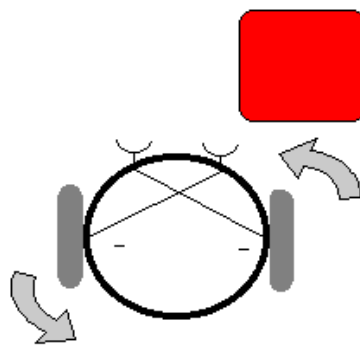
Ubicación de sensores	Factor de sensibilidad
Frontales	-17
Laterales 45 grados	-11
Laterales 90 grados	-7

Tabla 3.1 Factores de sensibilidad utilizados en el algoritmo propuesto

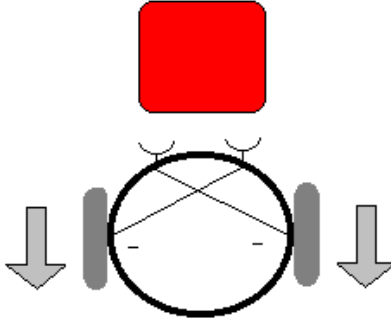
Uno de los problemas detectados al implementar el vehículo explorador de Britenberg en el robot Khepera, es el hecho de que los sensores infrarrojos de proximidad tienen un alcance limitado, por lo que las decisiones de cambio de rumbo para explorar nuevas áreas se realizan una vez que el robot está lo suficientemente cerca del obstáculo como para que los sensores perciban la presencia de éste (menos de 1.5 cm). Este problema es típico en robots que cuentan con este tipo de sensores, sonares y táctiles, por esto, se vio la necesidad de equipar al robot Khepera con un sistema de visión artificial que le permitiera tomar decisiones más acertadas en la tarea de exploración del ambiente.



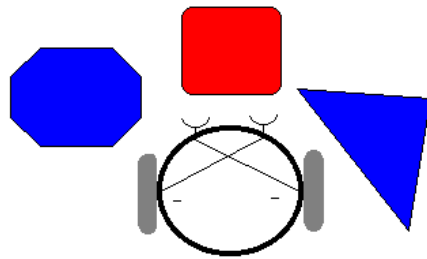
*Figura 3.2 Acción del robot al detectar obstáculo a la izquierda*



*Figura 3.3 Acción del robot al detectar obstáculo a la derecha*

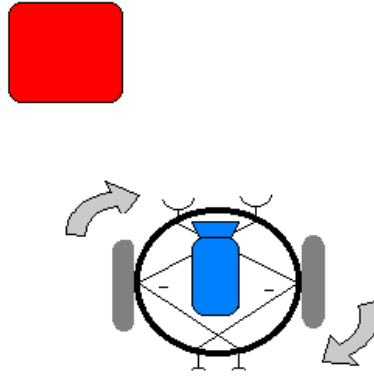


*Figura 3.4 Acción del robot al detectar obstáculo enfrente*

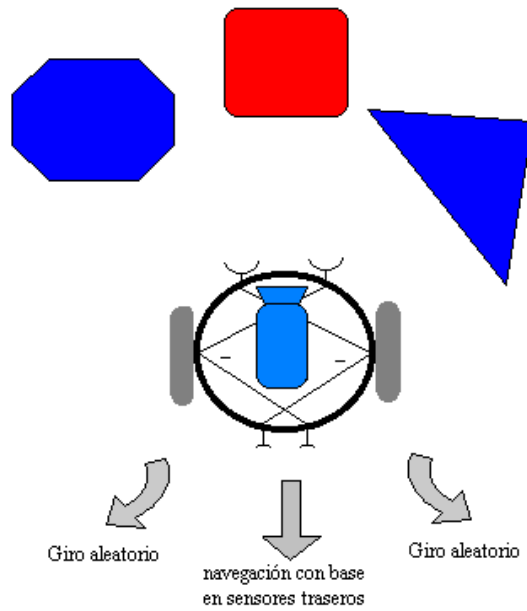


*Figura 3.5 Situación que provoca que el robot se detenga*

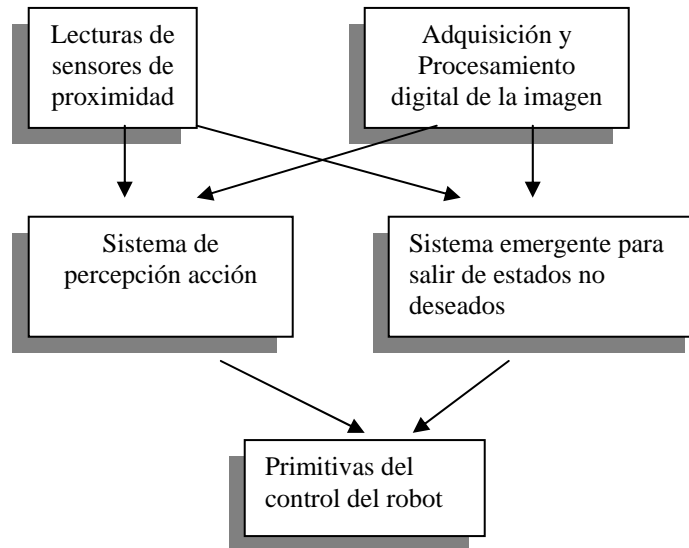
Otro problema que se detectó al implementar el vehículo explorador de Braitenberg es el hecho de que el robot puede detenerse (fig 3.5) y no continuar más con el trabajo de exploración, esto es cuando la atracción hacia lugares abiertos y la repulsión hacia los obstáculos se nivela, para resolver este problema, se implementó un mecanismo que detecta el problema y desactiva el algoritmo principal de navegación, para utilizar otro algoritmo similar pero ahora usando los sensores de proximidad 6 y 7 del robot khepera, es decir los sensores traseros, hasta salir de la posición que originó el conflicto, y una vez que se logró salir de esta porción del ambiente, que generalmente es un lugar sin salida, se realiza un giro aleatorio del robot y se regresa nuevamente el control al algoritmo principal para continuar con la tarea de exploración como se ve en la figura 3.7



*Figura 3.6 El robot detecta el obstáculo con la cámara antes de que sus sensores de proximidad los perciban*



*Figura 3.7 Solución al problema de nivelación de percepción de los sensores*



*Figura 3.8 Diagrama de bloques del sistema de exploración propuesto*

### **3.2 Desarrollo de algoritmos de visión para el robot Khepera**

El sensor principal para el sistema de visión del robot Khepera es una cámara digital de video modelo WebCam 3. Fue necesario construir un soporte de acrílico para la cámara que pudiera ser fácilmente ensamblado sobre el robot móvil, el resultado de adaptar la cámara se muestra en la figura siguiente.



*Figura 3.9 Robot Khepera con la cámara acoplada*

Para la adquisición de las imágenes se utilizó el software de la cámara QuickCam 3, el cual permite tomar fotografías de manera constante en un intervalo de tiempo definido, dicho intervalo puede ser como mínimo un segundo, esto limita la velocidad de reacción del robot debido a que la comunicación entre el software de la cámara y el sistema de control del robot, es por medio de un archivo de imagen en formato bmp. Sin embargo, se obtuvo una mejora en el tiempo necesario para la toma de decisiones sobre el cambio de rumbo hacia áreas libres, ya que estas decisiones se toman mucho antes de que los sensores de proximidad perciban un obstáculo y que el sistema de visión pudo ya identificar. A continuación se muestra una fotografía típica de un ambiente tomada por la cámara del robot.



*Figura 3.10 Vista del ambiente desde la perspectiva del robot*

En la fotografía anterior se pueden percibir dos obstáculos color rojo y parte de uno color azul, se restringió el problema a tener solo obstáculos de estos dos colores y las paredes blancas. En el ambiente existen más obstáculos rojos que azules, por tanto, es necesario que el sistema de visión robótico determine si existen obstáculos de este color en su trayectoria. Para esto, fue necesario segmentar la imagen y extraer solo porciones rojas de ella que representan a los obstáculos. Uno de los problemas encontrados al tratar de segmentar el color rojo, es que la imagen presenta heterogeneidad en los tonos de este color debido a la iluminación del entorno, por ello fue necesario utilizar un filtro que permitiera segmentar diversas tonalidades de rojos.

### 3.2.1 Segmentación de la imagen

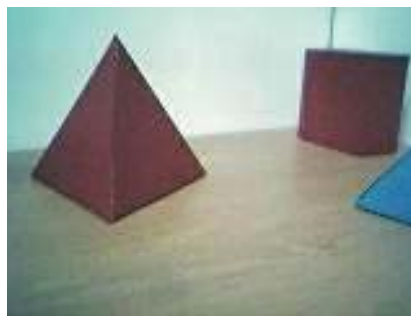
Al realizar diferentes pruebas de segmentación aplicando las componentes  $I1I2I3$  se observó que la componente  $I2$  presentaba mejores resultados, ya que en una imagen de color con múltiples colores al aplicarle la segmentación  $I2$  se obtuvieron diferentes tonalidades de grises para cada uno de los colores de la imagen, lo que permitía segmentar más fácilmente objetos de distintos colores.

La componente  $I2$  (fórmula 2.2) solo involucra las bandas de color rojo y azul sin incluir la banda verde, lo cual aunque muestra mejores resultados que al segmentar con las otras dos componentes, la información es incompleta en imágenes con color RGB. En [13] se propone una modificación a la componente  $I2$ , dando como resultado las fórmulas:

$$(3.3) I2_{GR} = \frac{G - R}{2}$$

$$(3.4) I2_{BG} = \frac{B - G}{2}$$

Para nuestro problema específico requeríamos segmentar porciones rojas en las imágenes que corresponden a los obstáculos en el ambiente, a continuación se muestra el proceso de segmentación utilizando la fórmula 3.3 para una imagen típica procesada por el sistema de visión del robot.



*Figura 3.11 Imagen original*





Figura. 3.12 Imagen resultante al aplicar la fórmula (3.3)

Como se puede ver en la figura 3.12 los objetos en color rojo, se muestra en tonalidades claras y los objetos azules tienen tonalidades aún más oscuras que el resto de la imagen. El siguiente paso fue la reducción en tamaño de la imagen, originalmente las fotografías tienen un tamaño de 160 x 120 píxeles y se redujo a una imagen de 40 x 30 píxeles. El método para reducir la imagen consiste en obtener un conjunto de ventanas siguiendo los puntos siguientes:

- Se tomaron ventanas de 5 x 5 píxeles y el promedio de los 25 píxeles forman el valor del nivel de gris de un píxel de la imagen reducida.
- Las ventanas de 5 x 5 píxeles tienen un traslape de una columna de 1 x 5 píxeles en el sentido horizontal y una fila de 5 x 1 píxeles en el sentido vertical.

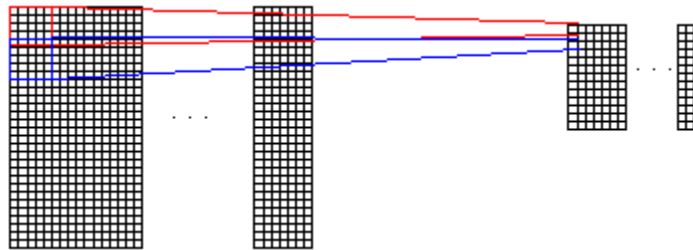
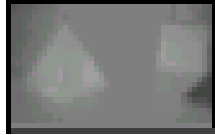


Figura 3.13 Obtención de las ventanas



*Figura 3.14 Reducción de escala al aplicar el método de las ventanas*

Una vez que se tiene la imagen a una escala menor, el siguiente paso es aplicar un umbral para obtener una imagen binaria, se eligió el umbral de 133 debido a que con este valor se obtuvieron buenos resultados en la segmentación de los objetos rojos en la imagen, los límites aceptables de umbralización tal que se obtuvo una imagen binaria con suficiente información útil es: [125 - 140].



*Figura 3.15 Imagen binaria al aplicar un umbral de 133*

### **3.2.2 Modificación del filtro I2\_GR**

Con el filtro I2\_GR como se mencionó anteriormente se obtuvieron buenos resultados al segmentar de las imágenes los obstáculos de color rojo, sin embargo se observó pérdida de información en colores oscuros, muchas veces las aristas de los objetos se presentan en colores oscuros, algunas veces es necesario identificar tipos de obstáculos o diferenciar entre obstáculos y paredes, Ulrich en [19] introduce una restricción en el problema que él soluciona, en donde las paredes del ambiente son pintadas con líneas verticales negras, esto con la finalidad de ayudar al sistema de control del robot a orientarse, este experimento no sería posible repetirlo con el robot kephera usando el filtro I2\_GR, se decidió modificar el filtro I2\_GR, para que permitiera segmentar tonalidades de rojos sin pérdida de información dada por colores oscuros, y tener la capacidad de en el futuro de repetir experimentos con el mencionado arriba, la idea fue la siguiente:

Partiendo de la ecuación (3.3), al implementar esta ecuación fue necesario aplicarle un corrimiento de 127 positivo. Esto es porque los valores arrojados por la ecuación están dentro del intervalo [-127,127], debido a que R, G Y B son valores entre [0,255], dando como resultado la ecuación (3.5).

$$(3.5) F = \frac{R-G}{2} + 127$$

Como se puede ver (3.5), no involucra al color azul, la modificación consiste en hacer que este canal aporte peso negativo para la realización de la segmentación en rojo, se decidió después de realizar varias pruebas que el canal azul reducirá en un porcentaje de hasta 50% del valor del nuevo píxel de la imagen segmentada resultante calculada con el filtro I2\_GR.

Entonces se requiere una función tal que cuando el valor del canal azul sea máximo, la reducción también sea máxima cuyo valor sería de a lo más 127, y cuando el valor del canal azul sea pequeño la reducción sea mínima, para mapear esto se utilizó una función exponencial neperiana como se muestra en la figura 3.16

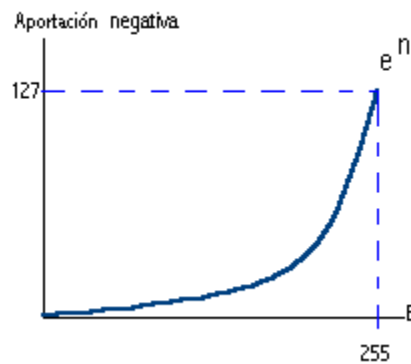


Figura 3.16 Función de mapeo

Entonces el filtro queda como sigue:

$$(3.6) \quad F = \frac{R-G}{4} - e^n + 127$$

El valor de  $n$  deberá estar en función del valor de la componente azul, es decir cuando la componente azul sea 255,  $e^n$  deberá ser igual a 127, por tanto  $n$  deberá ser a lo mas  $\ln(127) = 4.844$ , para calcular el valor de  $n$  en función de  $B$ (componente azul), se utilizó la ecuación de la recta mostrada en la figura 3.17.

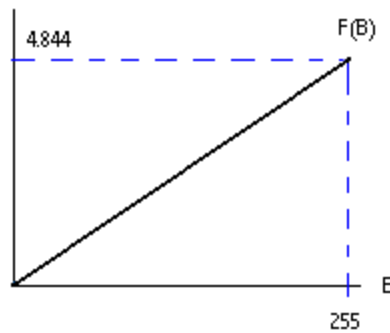


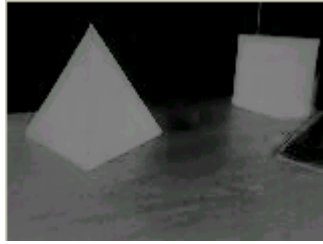
Figura 3.17 Función de mapeo 2

$$(3.7) \quad F(B) = 0.0189 * B$$

La ecuación del filtro final queda como sigue, el valor absoluto nos descarta la posibilidad de tener valores negativos en nuestra imagen resultante:

$$(3.8) \quad F = \left| \frac{R-G}{4} - e^{B*0.0189} + 127 \right|$$

Utilizando la ecuación (3.8), en la imagen 3.11 obtenemos:



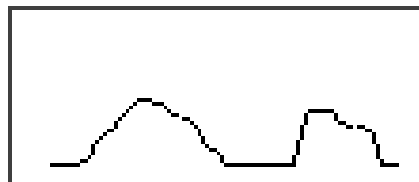
*Figura 3.18 Imagen resultante al aplicar el filtro propuesto*

Es evidente que los resultados obtenidos con el filtro modificado I2\_GR propuesto en este capítulo son mucho mejores en el sentido de que los obstáculos se aprecian mejor definidos, lo que permite segmentar de manera más eficiente los objetos rojos en la imagen.

### **3.2.3 Acoplación del sistema de visión al vehículo explorador de Braitenberg**

Debido a que los sensores infrarojos con los que cuenta el robot khepera, tienen un alcance muy limitado, se le adaptó el sistema de visión arriba mencionado, dicho sistema entrega como resultado una imagen binaria como la mostrada en la figura 3.15

Una vez que se tiene la imagen binaria, se calcula el histograma en el eje “x” obteniéndose un vector de longitud 40, en donde cada uno de sus valores representa la frecuencia de aparición de píxeles blancos sobre la columna en cuestión, al graficar este vector se obtiene la figura 3.18.



*Figura 3.19 Histograma en x*

Al observar la figura 3. 18 podemos identificar 2 picos, cada uno de los cuales representa a los objetos rojos identificados sobre el horizonte del sistema de visión del robot.

Al tener más información sobre el ambiente gracias al sistema de visión del robot, las ecuaciones que describen el comportamiento del vehículo explorador de Braitenberg adaptadas al robot khepera, se modificaron de la siguiente manera:

$$(3.9) \text{ motor\_derecho} = \frac{vc + cs0 + cs1 + cs2}{2} + \frac{hizq}{-35}$$

$$(3.10) \text{ motor\_izquierdo} = \frac{vc + cs3 + cs4 + cs5}{2} + \frac{hder}{-35}$$

donde:

vc = velocidad de crucero

cs0 = contribución de sensor cero

cs1 = contribución de sensor uno

cs2 = contribución de sensor dos

cs3 = contribución de sensor tres

cs4 = contribución de sensor cuatro

cs5 = contribución de sensor cinco

$$hizq = \sum_{i=0}^{i=19} \text{vector\_histograma}[i]$$

$$hder = \sum_{i=20}^{i=39} \text{vector\_histograma}[i]$$

Considerando las ecuaciones (3.1) y (3.2) podemos conocer que la velocidad máxima con la que avanza el robot, es igual a la velocidad de crucero y la velocidad máxima con la que el robot puede girar para evadir un obstáculo es de  $-35$  pulsos/10ms en cada uno de los motores, entonces las ecuaciones (3.9) y (3.10) describen un comportamiento tal que la velocidad de giro para evadir un obstáculo, ahora está en función en un 50% de las aportaciones de los sensores de proximidad y el otro 50% en la detección de objetos de color rojo por el sistema de visión del robot. La aportación del sistema de visión a la

velocidad de los motores, está dada por la sumatoria de las frecuencias de la mitad izquierda y derecha del histograma, cada una de estas sumas a lo más puede ser de 600, esto es porque la altura de la imagen reducida es de 30 y la longitud de la mitad del histograma en “x” es igual a 20, de esto se desprende que a lo más la detección de un obstáculo aporta  $600/(-35) \approx -17$ , por lo tanto, la velocidad máxima del robot para hacer una maniobra de evasión de obstáculos será de  $-34$  pulsos/10ms.

Al implementar este nuevo comportamiento el robot tuvo la capacidad de evitar obstáculos rojos mucho antes de que los sensores de proximidad los percibieran. Para el caso de los obstáculos azules si bien es cierto el sistema de visión no los detecta, pero al aproximarse el robot lo suficiente a ellos, los sensores de proximidad los percibirán y pondrán en marcha una maniobra para evitarlos.

### **3.3 Conclusiones del capítulo**

En este capítulo se propusieron e implementaron mejoras al vehículo definido por Braitenberg en [4] llamado vehículo con comportamiento explorador, dichas mejoras consistieron en adaptarle un mecanismo que permite al robot recuperarse de situaciones en las que queda varado dentro del ambiente desconocido, esto sucede al nivelarse la percepción de los sensores con los que cuenta el robot, y por medio de los cuales presenta repulsión hacia los obstáculos en el camino y atracción hacia espacios abiertos, otra aportación importante a los vehículos de Braitenberg consistió en que el comportamiento del agente autónomo ahora se ve influenciado por un sistema de visión robótico y no solo de sensores de proximidad, lo que permite tomar mejores decisiones sobre el rumbo que debe tomar en la tarea de exploración, la influencia del sistema de visión en la toma de decisiones puede ser ajustado por medio de un factor de sensibilidad, por lo que es posible controlar la distancia de acercamiento del robot a los obstáculos, dentro del sistema de visión del robot se propuso un filtro que permitió segmentar imágenes en color con heterogeneidad en los tonos de un mismo color, sin perder información que pudiera servir para aplicar algoritmos de navegación por marcas, como parte de experimentos futuros con el robot Khepera.