

Capítulo 4

OBTENCIÓN DEL ÁNGULO DE ORIENTACIÓN DE LA CÁMARA

El cálculo del ángulo de orientación de la cámara es una tarea importante en el campo de la robótica móvil. Nos puede proporcionar información de orientación del robot cuando nos encontramos en una posición desconocida dentro de un entorno, conocido o no. En este capítulo vamos a presentar un nuevo método de obtención de dicho ángulo que se deriva del presentado en [Coughlan y Yuille, 1999]. Entre las ventajas de este método destacamos la utilización de una única imagen y su robustez ya que se basa en un esquema de inferencia bayesiana. Sin embargo, la eficiencia del método debe mejorarse para que éste sea aplicable a robótica móvil. Proponemos utilizar como base para la inferencia el agrupamiento de uniones descrito en capítulos anteriores ya que dicha estructura acelera la estimación. Para ello revisaremos los supuestos de aplicación del método bayesiano y lo situaremos en el contexto de las técnicas basadas en puntos de fuga.

El esquema bayesiano de obtención de la orientación relativa asume que el mundo se ajusta a la estructura del mundo de Manhattan. En el mundo de Manhattan tenemos un sistema de coordenadas en $\vec{i}, \vec{j}, \vec{k}$ (ver Figura 4.1). Si podemos determinar la posición relativa de la cámara en este sistema de coordenadas, podremos interpretar fácilmente la escena que estamos visualizando. Una característica propia de este tipo de mundo son los puntos de fuga (*vanishing points*).

Los puntos de fuga son aquellos puntos en el plano imagen donde convergen las proyecciones de las líneas paralelas del mundo. Son características invariantes a escala y rotación, por lo que pueden ser utilizadas para tareas de correspondencia entre imágenes [Moons *et al.*, 1996], calibración de la cámara o reconocimiento de objetos [Parodi y Piccioli, 1996, Shufelt, 1999a].

En un mundo estructurado tenemos tres puntos de fuga, uno por cada orientación posible de las aristas. Las aristas verticales tienen el punto de fuga en el infinito.

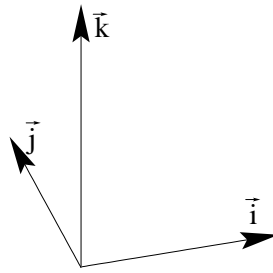


Figura 4.1: Sistema de coordenadas en el mundo de Manhattan.

En adelante vamos a suponer que nos encontramos en un entorno estructurado. Este tipo de suposición es perfectamente válida para la mayoría de entornos por los cuales un robot se va a mover: pasillos, habitaciones, naves industriales, etc. Sin embargo, también es perfectamente válido para entornos de exterior, siempre y cuando nos aparezcan estructuras construidas por el hombre tales como aceras, edificios, carreteras, etc.

Una última suposición es que la cámara sólo se puede orientar en el plano horizontal, es decir, gira alrededor del eje \vec{k} .

4.1 Técnicas de obtención de puntos de fuga

En esta sección vamos a presentar los primeros enfoques en la detección de puntos de fuga. Uno de los primeros trabajos en este sentido es [Barnard, 1983]. En él se propone una metodología que ha establecido el paradigma para la detección de puntos de fuga más ampliamente extendido. Se supone conocida la distancia focal de la cámara, por lo que conocemos el centro de coordenadas. Se define entonces una esfera centrada en dicho centro, con radio unitario. Esta esfera se utiliza como espacio de orientaciones. Cada segmento de línea en la imagen tiene dos puntos, que junto con el centro de la esfera forman un plano, llamado plano

de interpretación. Cada plano intersecta con la esfera para formar un círculo (ver Figura 4.2). La esfera se utiliza como acumulador: zonas de la esfera donde inter-

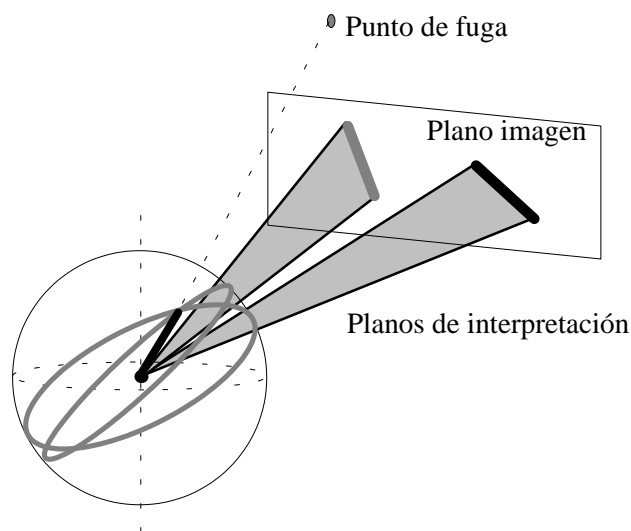


Figura 4.2: Esfera de acumulación para el cálculo del punto de fuga.

sectan círculos indican orientaciones comunes entre los segmentos que han generado los círculos. La idea de este método consiste en buscar zonas de la esfera más votadas, al estilo de la transformada de Hough. Este trabajo inicial fue extendido en [Magee y Aggarwal, 1984] y [Quan y Mohr, 1989], utilizando geometría esférica y transformada de Hough jerárquica, respectivamente. En general, la mayoría de los métodos propuestos asumen que los puntos iniciales y finales de los segmentos están bien calculados. Otros métodos propuestos basados en este enfoque son: [Brillault-O Mahoney, 1991], [McLean y Kotturi, 1995] y [Shufelt, 1999b]. Recientemente se ha propuesto un nuevo enfoque [Coughlan y Yuille, 1999] que no toma decisiones a priori sobre si un píxel pertenece o no a una arista, sino que se basa en un enfoque bayesiano. Además, explota las restricciones geométricas que se presentan en un mundo estructurado. Profundizamos este enfoque en este capítulo.

4.2 Geometría tridimensional

Vamos a definir y detallar algunos conceptos de geometría tridimensional que utilizaremos después en la definición del método. Definimos Ψ como el ángulo de orientación de la cámara dentro del mundo de Manhattan [Faugeras, 1993]: la cámara está orientada en la dirección $\cos\Psi\vec{i} - \sin\Psi\vec{j}$. Las coordenadas en el plano imagen $\vec{u} = (u, v)$ se relacionan con las coordenadas del mundo (x, y, z) mediante las ecuaciones:

$$u = \frac{f \cdot (-x \sin\Psi - y \cos\Psi)}{x \cos\Psi - y \sin\Psi}, \quad v = \frac{f \cdot z}{x \cos\Psi - y \sin\Psi} \quad (4.1)$$

donde f es la longitud focal de la cámara.

Los puntos de fuga de las líneas en las direcciones \vec{i} y \vec{j} se encuentran en los puntos $(-f \tan\Psi, 0)$ y $(f \cot\Psi, 0)$, respectivamente, en el plano imagen. Las líneas en la dirección \vec{k} son todas verticales debido a la suposición de encontrarnos en un mundo estructurado.

Por último, vamos a relacionar el gradiente de intensidad obtenido en un punto de la imagen con el ángulo de orientación. Un punto de la imagen $\vec{u} = (u, v)$ con gradiente de intensidad $(\cos\theta, \sin\theta)$ es consistente con una línea \vec{i} (es decir, apunta al punto de fuga), si $-v \tan\theta = u + f \tan\Psi$. Esta ecuación no cambia si añadimos $\pm\pi$ a θ , es decir, la polaridad de la arista no afecta a los cálculos. De forma similar tenemos $v \tan\theta = -u + f \cot\Psi$ para las líneas \vec{j} .

4.3 Modelo bayesiano inicial

Vamos a presentar el modelo bayesiano propuesto en [Coughlan y Yuille, 1999]. La diferencia de este modelo con respecto a otros enfoques propuestos con anterioridad es que no nos fuerza a decidir prematuramente qué píxeles están o no en una arista. Además, permitirá etiquetar posteriormente un píxel como perteneciente a una arista de los tres tipos que nos podemos encontrar en el mundo de Manhattan: $\vec{i}, \vec{j}, \vec{k}$.

4.3.1 Estadísticas de las aristas

En este trabajo no se utiliza un mapa de aristas binario, sino un modelo estadístico que aporta más información a la hora de etiquetar un píxel de la imagen como perteneciente a una arista. Para ello, se utiliza el trabajo desarrollado en [Konishi *et al.*, 1999]. El método y las estadísticas obtenidas se detallan en la Sección 2.5.

4.3.2 Evidencia de un píxel

Tenemos el valor del gradiente $\vec{E}_{\vec{u}}$ (magnitud y orientación) en un punto de la imagen \vec{u} . Este valor se explica mediante uno de los cinco modelos siguientes $m_{\vec{u}}$: $m_{\vec{u}} = 1, 2, 3$ para cada una de las posibles aristas generadas por líneas en las direcciones \vec{i}, \vec{j} y \vec{k} , respectivamente, en la escena; $m_{\vec{u}} = 4$ significa que el gradiente ha sido generado por una arista aleatoria (no debida a una arista $\vec{i}, \vec{j}, \vec{k}$); y, por último, $m_{\vec{u}} = 5$ significando que el píxel no está en una arista. Las probabilidades a priori $P(m_{\vec{u}})$ de cada modelo las hemos estimado experimentalmente.

Se asume que la probabilidad del gradiente de la imagen $\vec{E}_{\vec{u}}$ tiene dos factores correspondientes a la componente de magnitud ($E_{\vec{u}}$) y orientación del gradiente ($\sigma_{\vec{u}}$):

$$P(\vec{E}_{\vec{u}}|m_{\vec{u}}, \Psi, \vec{u}) = P(E_{\vec{u}}|m_{\vec{u}})P(\phi_{\vec{u}}|m_{\vec{u}}, \Psi, \vec{u}) \quad (4.2)$$

donde $P(E_{\vec{u}}|m_{\vec{u}})$ es igual a $P_{off}(E_{\vec{u}})$ si $m_{\vec{u}} = 5$ o $P_{on}(E_{\vec{u}})$ si $m_{\vec{u}} \neq 5$. Por otro lado, $P(\phi_{\vec{u}}|m_{\vec{u}}, \Psi, \vec{u})$ es igual a $P_{ang}(\phi_{\vec{u}} - \theta(\Psi, m_{\vec{u}}, \vec{u}))$ si $m_{\vec{u}} = 1, 2, 3$ o $U(\phi_{\vec{u}})$ si $m_{\vec{u}} = 4, 5$. La orientación $\theta(\Psi, m_{\vec{u}}, \vec{u})$ es la normal determinada por la ecuación $-v \tan \theta = u + f \tan \Psi$ para líneas \vec{i} , $v \tan \theta = -u + f \cot \Psi$ para líneas \vec{j} , y $\theta = 0$ para líneas \vec{k} . Ψ es el ángulo de orientación determinado de la cámara.

En la propuesta inicial, en vez de determinar el modelo concreto al que pertenece un píxel, se acumula la evidencia para los cinco posibles modelos:

$$P(\vec{E}_{\vec{u}}|\Psi, \vec{u}) = \sum_{m_{\vec{u}}=1}^5 P(\vec{E}_{\vec{u}}|m_{\vec{u}}, \Psi, \vec{u})P(m_{\vec{u}}) \quad (4.3)$$

De esta forma vamos a ser capaces de determinar la evidencia del ángulo de la cámara Ψ sin conocer a cuál de los cinco modelos pertenece el píxel.

4.3.3 Evidencia de todos los píxeles

Para determinar la evidencia sobre todos los píxeles de la imagen, denotados por $\{\vec{E}_{\vec{u}}\}$, se asume que los datos de la imagen son condicionalmente independientes en todos los píxeles, dada la dirección de orientación Ψ :

$$P(\{\vec{E}_{\vec{u}}\}|\Psi) = \prod_{\vec{u}} P(\vec{E}_{\vec{u}}|\Psi, \vec{u}) \quad (4.4)$$

Así, la distribución a posteriori sobre la dirección de orientación viene dada por el producto: $\prod_{\vec{u}} P(\vec{E}_{\vec{u}}|\Psi, \vec{u})P(\Psi)/Z$, donde Z es un factor de normalización y $P(\Psi)$ es la distribución a priori uniforme sobre el ángulo de orientación. Para

encontrar el MAP necesitamos maximizar el log del anterior término, ignorando Z que es independiente frente a Ψ :

$$\log[P(\{\vec{E}_{\vec{u}}\}|\Psi)P(\Psi)] = \log P(\Psi) + \sum_{\vec{u}} \log[\max_{m_{\vec{u}}} P(\vec{E}_{\vec{u}}|m_{\vec{u}}, \Psi, \vec{u})P(m_{\vec{u}})] \quad (4.5)$$

El algoritmo propuesto en un principio evalúa numéricamente el log a posteriori para cada dirección de orientación Ψ en el rango -45° a $+45^\circ$, tomada en incrementos de un grado (ver Figura 4.3). En la Figura 4.4 se muestra un resultado de evaluación de esta función. En ellas, podemos ver la existencia de un único máximo.

Algoritmo de obtención del ángulo de orientación de la cámara ALG_CA

Entrada: Imagen de entrada
Salida: Ángulo de orientación detectado.

Procesar la imagen para obtener el gradiente (magnitud y orientación).
 Calcular el P_{on} y el P_{off} de cada píxel de la imagen.

Para cada punto de la imagen hacer
 Para cada ángulo posible ($[-45, 45]$) **hacer**
 Calcular la evidencia del píxel con el ángulo actual mediante la Ecuación 4.6.
 Acumular la evidencia para el ángulo en concreto.

FinPara

FinPara
 Devolver el ángulo con valor máximo de evidencia.

Figura 4.3: Algoritmo de obtención del ángulo de orientación de la cámara.

4.3.4 Resultados

Se ha aplicado el método de [Coughlan y Yuille, 1999] a varias imágenes. Las Figuras 4.5, 4.6, 4.7 y 4.8 muestran la aplicación del método en escenas de exterior; las Figuras 4.9 y 4.10 hacen lo propio con escenas de interior. Para cada figura se ha mostrado dos imágenes: la de la izquierda muestra hacia donde apuntan los puntos de fuga de las líneas \vec{i} y \vec{j} (las líneas \vec{k} apuntan verticalmente al infinito); la de la derecha muestra el etiquetado final de los píxeles: rojo, modelo 1, verde, modelo 2, azul, modelo 3, negro, modelo 4, y blanco, modelo 5.

Las probabilidades a priori para cada modelo $P(m_{\vec{u}})$, se han determinado experimentalmente. Para estos experimentos hemos utilizado 0.2, 0.2, 0.2, 0.4, 0.9 para $m_{\vec{u}} = 1, 2, 3, 4, 5$, respectivamente. Sin embargo, cuando queremos detectar

los objetos no alineados con el mundo de Manhattan (ver Figuras 4.13 y 4.14), hemos aumentado la probabilidad del modelo 4 a 6.6. Para la función P_{ang} hemos sintonizado los siguientes valores: $\epsilon = 0.1$ y $\tau = 4^\circ$.

Las Figuras 4.5, 4.6, 4.7 y 4.8 muestran resultados del algoritmo con imágenes tomadas en el exterior. Las Figuras 4.9, 4.10, 4.11 y 4.12 corresponden a imágenes de interior. Podemos observar que no se ha tenido un cuidado especial en mantener la horizontalidad de la cámara para la obtención de estas imágenes. Sin embargo, el algoritmo responde bastante bien a pequeños cambios de orientación con respecto a la horizontal. El tiempo medio de ejecución fue de 90 segundos en un Pentium II a 266 Mhz trabajando con imágenes de 640x480.

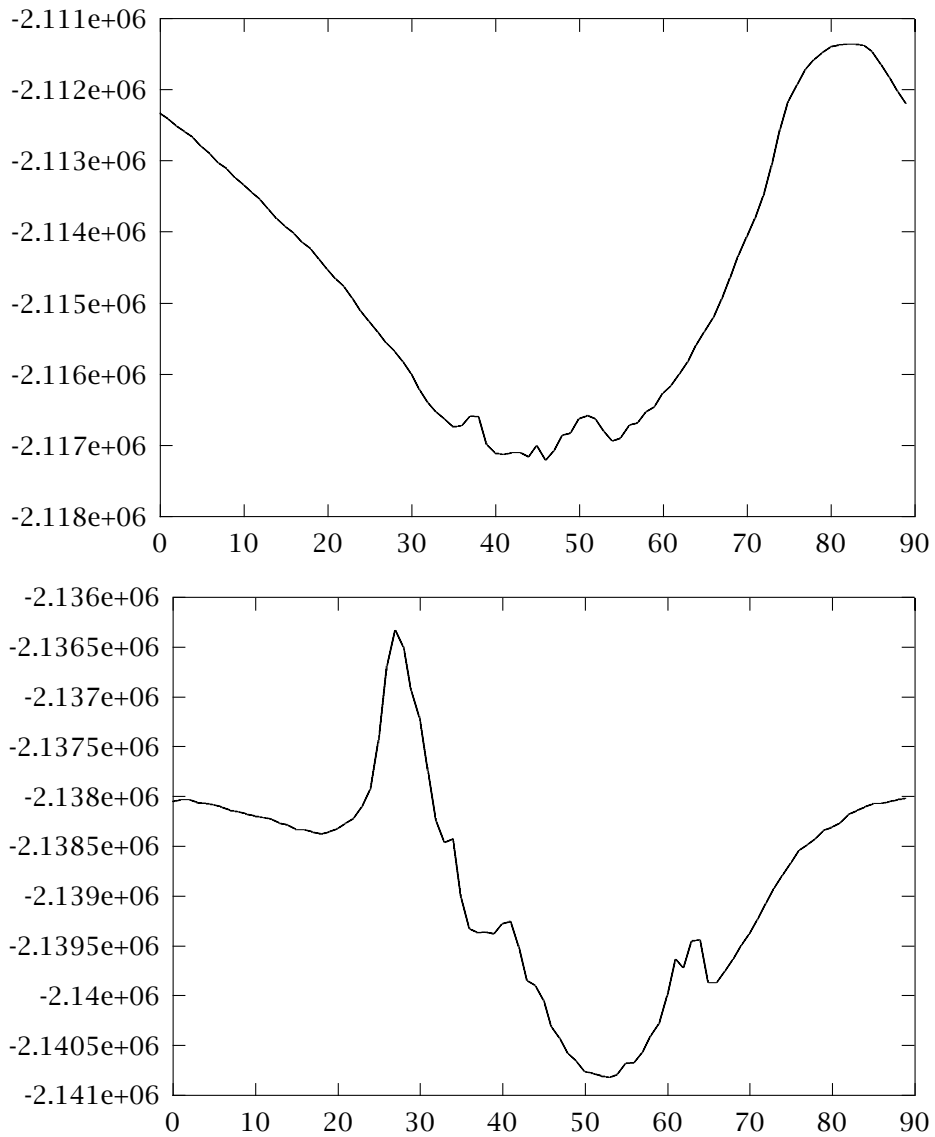


Figura 4.4: Gráficas del MAP frente al ángulo de orientación. Estas gráficas se corresponden con las Figuras 4.5 y 4.9.

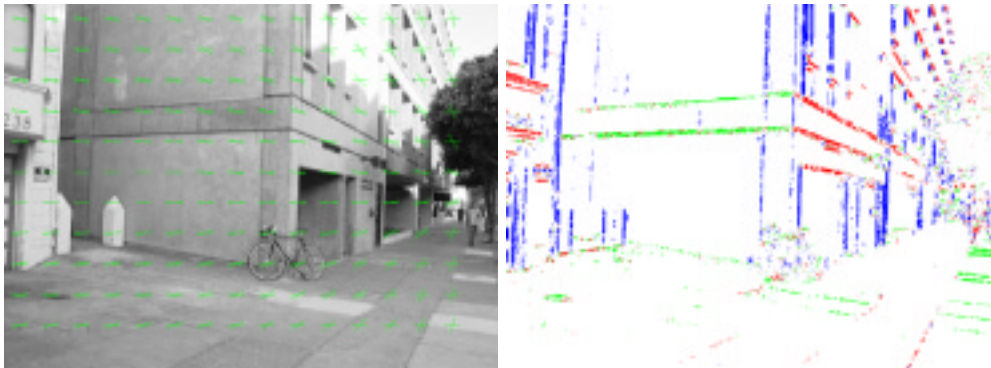


Figura 4.5: Resultado obtenido en imagen de exterior (1). El ángulo de orientación de la cámara obtenido fue de -18° .

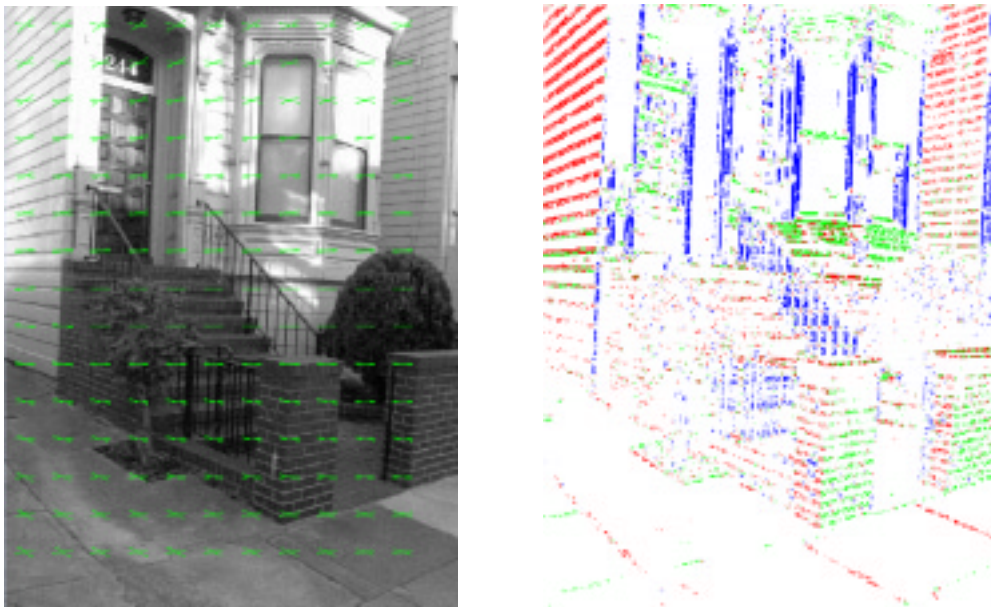


Figura 4.6: Resultado obtenido en imagen de exterior (2). El ángulo de orientación de la cámara obtenido fue de 32° .

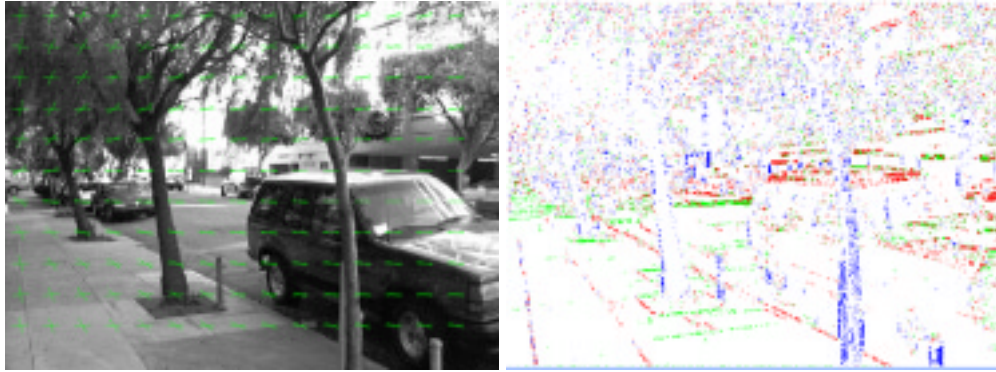


Figura 4.7: Resultado obtenido en imagen de exterior (3). El ángulo de orientación de la cámara obtenido fue de 20° .



Figura 4.8: Resultado obtenido en imagen de exterior (4). El ángulo de orientación de la cámara obtenido fue de 8° .

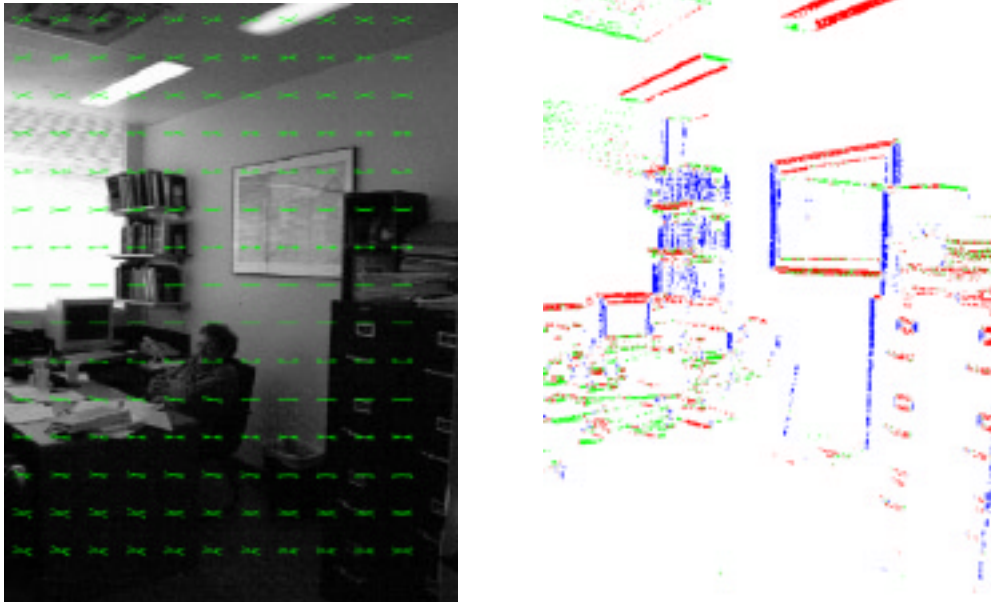


Figura 4.9: Resultado obtenido en imagen de interior (1). El ángulo de orientación de la cámara obtenido fue de 37° .

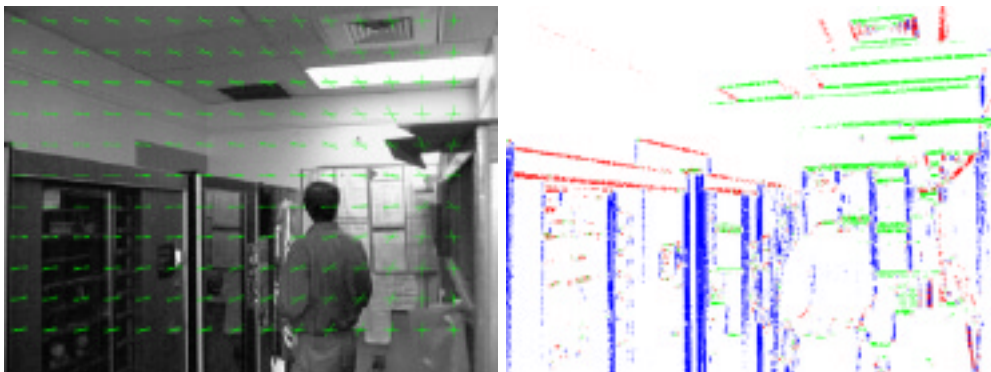


Figura 4.10: Resultado obtenido en imagen de interior (2). El ángulo de orientación de la cámara obtenido fue de -15° .

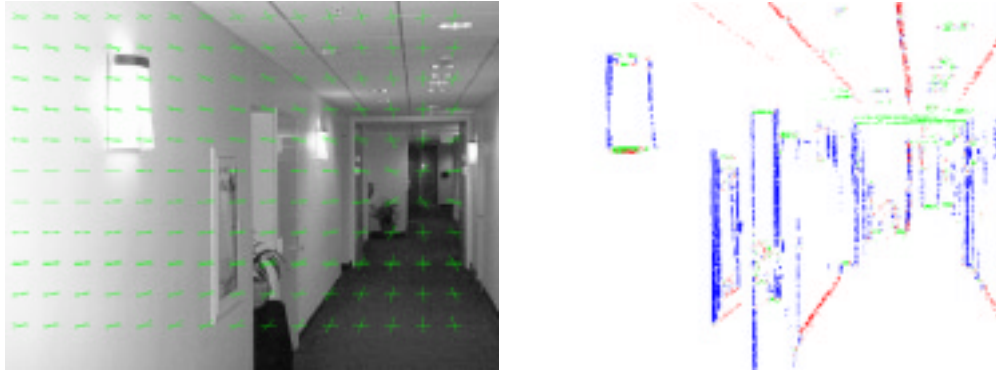


Figura 4.11: Resultado obtenido en imagen de interior (3). El ángulo de orientación de la cámara obtenido fue de -13° .

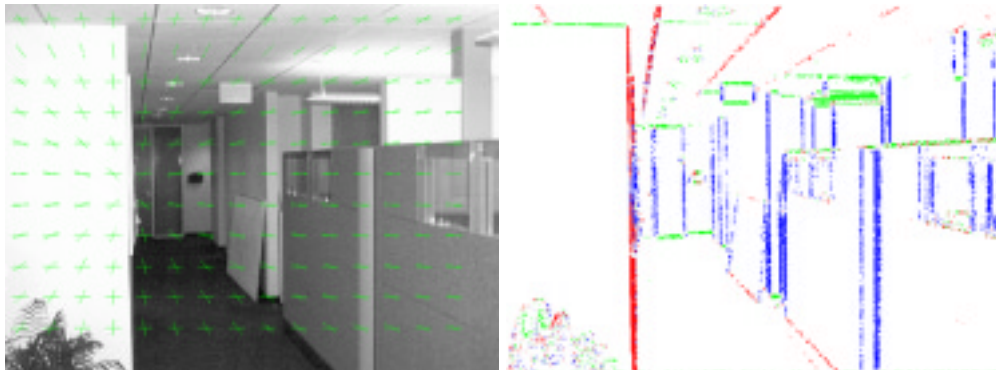


Figura 4.12: Resultado obtenido en imagen de interior (4). El ángulo de orientación de la cámara obtenido fue de 10° .

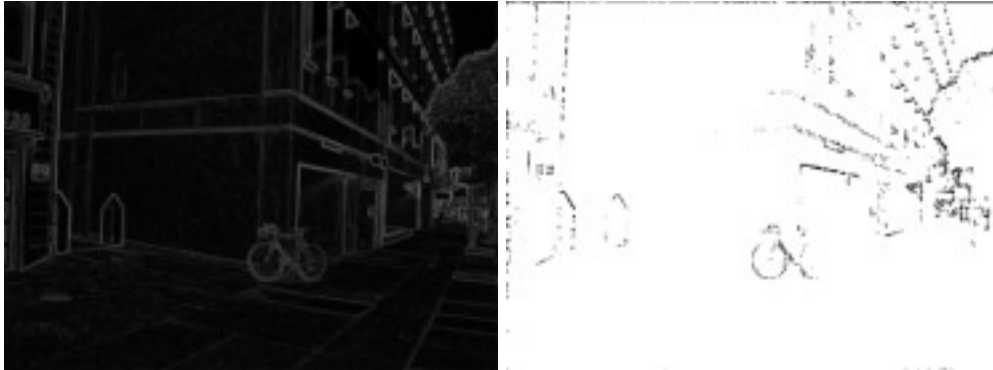


Figura 4.13: Puntos etiquetados con modelo 4 (derecha) y magnitud del gradiente (izquierda).

4.3.5 Etiquetado de los píxeles

Una aplicación inmediata de este método es la detección de objetos cuya forma no esté alineada con el mundo de Manhattan, es decir, la dirección de sus aristas no apuntan a los puntos de fuga. Estos puntos se corresponderán con el modelo 4. Para detectar estos puntos, una vez calculado el ángulo de orientación, podemos etiquetar cada píxel de la imagen estimando el modelo mediante el MAP de $P(m_{\vec{u}} | \vec{E}_{\vec{u}}, \Psi^*, \vec{u})$ para estimar el $m_{\vec{u}}$ en cada píxel \vec{u} . Para realizar este etiquetado se ha aumentado la probabilidad a priori del modelo 4 a 6.6. En las Figuras 4.13 y 4.14 se muestran resultados obtenidos en el etiquetado de estos píxeles.

4.3.6 Conclusiones del método

Este método nos permite calcular la orientación de la cámara a partir de la evidencia de los datos de la imagen. No es necesario tener un conocimiento específico del entorno en el que nos movemos, ni tomar decisiones precipitadas sobre la pertenencia de un píxel a un determinado tipo de línea. También nos permite detectar aquellos píxeles cuya forma no sea estructurada. De esta forma podremos aislar aquellos objetos, tales como personas, que se encuentren en nuestro entorno y que son obstáculos potenciales.

En cuanto a las limitaciones del método, como todo modelo bayesiano se ve afectado mucho por las funciones de distribución obtenidas. Para un entorno determinado, se puede hacer necesario tomar nuevas estadísticas de la respuesta de

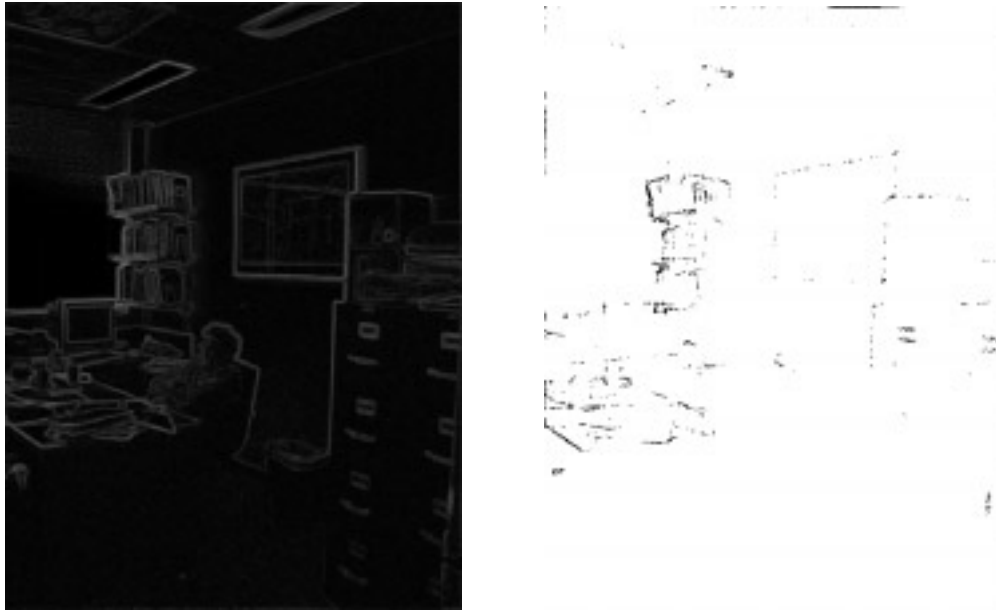


Figura 4.14: Puntos etiquetados con modelo 4 (derecha) y magnitud del gradiente (izquierda).

de un determinado filtro de aristas, así como ajustar las probabilidades a priori de cada modelo. Una vez ajustados estos datos, es posible que los datos de la imagen no nos proporcionen suficiente evidencia como para determinar con exactitud el ángulo de orientación de la cámara. Esto puede ser debido a que las aristas presentes en la imagen no nos proporcionen una correcta información. Por ejemplo, la imagen Figura 4.15 nos muestra un etiquetado erróneo debido a la no utilización del flash en la obtención de la imagen. Por ello, debemos ser especialmente cuidadosos en la obtención de las probabilidades que modelen nuestro entorno. Otra limitación es el tiempo de ejecución. El tiempo medio fue de 90 segundos. De esta forma, se hace muy difícil pensar en la implementación de este algoritmo en una plataforma móvil y trabajar en tiempo real.

En cuanto a posibles ampliaciones del método, una inmediata es incorporar las posibles orientaciones de la cámara en los dos ejes restantes. Esto no parece una tarea fácil a priori, pero permitiría una localización exacta de la orientación de la cámara. También, una posible ampliación es la utilización de mejores filtros de gradiente. En el trabajo antes mencionado [Konishi *et al.*, 1999] presentan una

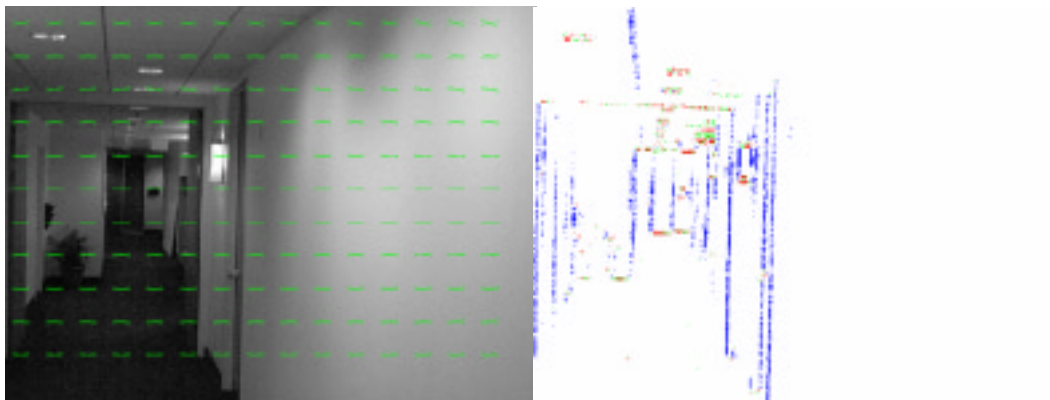


Figura 4.15: Resultado erróneo debido a la mala calidad de la imagen obtenida.

medida de la efectividad de los filtros de obtención de aristas. La utilización de filtros de color a diferentes escalas proporcionaría mejores resultados en cuanto a la localización y dirección exacta de las aristas. Sin embargo, el color es una característica muy dependiente del entorno en el que nos movemos. Los colores de un prado al aire libre no son los mismos que en el interior de un edificio de oficinas.

4.4 Obtención del ángulo de orientación de la cámara utilizando la información de arista

Una crítica inmediata al algoritmo de obtención del ángulo de orientación es el tiempo de ejecución elevado. Vamos a plantear un nuevo enfoque, pero haciendo uso de la formulación bayesiana planteada al principio del capítulo. Para ello utilizaremos el resultado de aplicar el método de agrupamiento de uniones desarrollado en el Capítulo 3.

4.4.1 Obtención de caminos rectos

La información obtenida como resultado de aplicar el algoritmo de agrupamiento, propuesto en el capítulo anterior, es un conjunto de vértices y caminos entre estos vértices. Para disminuir el tiempo de ejecución hemos convertido los caminos en segmentos rectos. Hemos utilizado el algoritmo propuesto en [Lowe, 1987] que se

detalla en el Apéndice C. Por lo tanto, después de aplicar dicho algoritmo tenemos un conjunto de segmentos rectos: C . Este conjunto será la entrada para el cálculo del ángulo de orientación.

4.4.2 Estadística de las aristas

La formulación bayesiana es la misma que la definida al principio de este capítulo. Lo que hemos variado son las funciones $P_{on}(E_{\vec{u}})$ y $P_{off}(E_{\vec{u}})$. Estas funciones tomarán los siguientes valores:

$$P_{on}(E_{\vec{u}}) = \begin{cases} 0.99 & \text{si } E_{\vec{u}} \in C \\ 0.01 & \text{en otro caso} \end{cases}$$

$$P_{off}(E_{\vec{u}}) = \begin{cases} 0.99 & \text{si } E_{\vec{u}} \notin C \\ 0.01 & \text{en otro caso} \end{cases}$$

También el ángulo de orientación en cada píxel ($\sigma_{\vec{u}}$) cambia, puesto que ahora no se calcula del gradiente, sino que cada píxel tiene asociado el ángulo de la normal del segmento correspondiente, caso de que el píxel pertenezca a una arista.

En la práctica, y en aras de mejorar la eficiencia, computamos sólo una vez por cada segmento el ángulo de la normal a dicho segmento y acumulamos el correspondiente valor multiplicando por su longitud, ya que dicha longitud da una idea de la confianza en la arista.

En la experimentación previa del algoritmo hemos detectado ciertos errores. En la Figura 4.16 se muestra un cálculo erróneo del ángulo de orientación, debido a que cerca del punto de fuga existen líneas verticales que pertenecen al modelo 1 y al modelo 3 a la vez. Por ello acumulan dos veces la misma evidencia debido a la suma de los valores que se produce en la Ecuación 4.3. Para solucionar este problema proponemos seleccionar el máximo valor de los cinco modelos, en vez de la suma. La Ecuación 4.3 quedaría:

$$P(\vec{E}_{\vec{u}}|\Psi, \vec{u}) = \max_{m_{\vec{u}}} \{P(\vec{E}_{\vec{u}}|m_{\vec{u}}, \Psi, \vec{u})P(m_{\vec{u}})\} \quad (4.6)$$

La mejora al aplicar esta nueva ecuación se muestra en la Figura 4.17, en la cual el ángulo de orientación es calculado correctamente. En la Figura 4.18 se muestra el log de la probabilidad a posteriori. Cuando utilizamos la suma podemos observar un pico en -15 grados que es el máximo de la función. Esto es debido a la acumulación de evidencia por parte de dos modelos con ese ángulo: modelo 1 (rectas \vec{i}) y modelo 3 (rectas \vec{k}). Cuando utilizamos el máximo el pico se atenúa y podemos seleccionar el valor correcto.

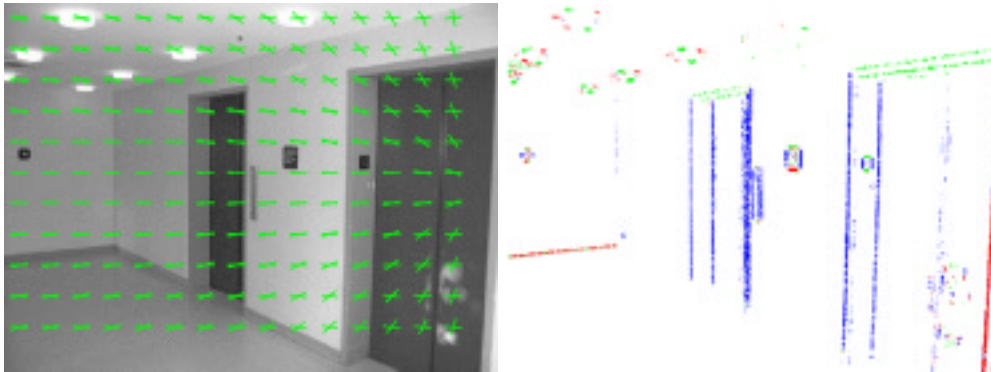


Figura 4.16: Resultado erróneo ($\Psi = -15^\circ$) utilizando la suma de la evidencia para cada modelo.

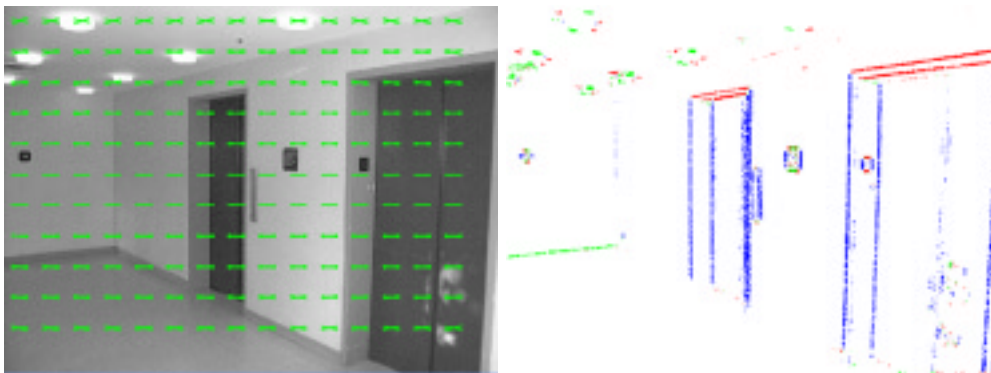


Figura 4.17: Resultado correcto ($\Psi = 40^\circ$) utilizando el máximo valor de evidencia.

4.4.3 Cálculo del ángulo de orientación

Una vez comentadas las modificaciones efectuadas al algoritmo original, nuestra propuesta completa se muestra en la Figura 4.19.

4.4.4 Resultados

Se ha aplicado este método al mismo conjunto de imágenes que en el método anterior. Una vez aplicado el método, el error medio cometido con respecto al método anterior fue de 0.5 grados. En cuanto al tiempo de computación, hemos

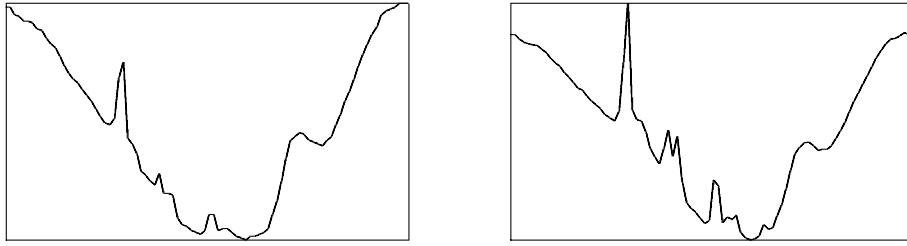


Figura 4.18: Función log posteriori. Izquierda: utilizando el máximo de la función. Derecha: utilizando la suma.

Algoritmo de obtención del ángulo de orientación de la cámara ALG_CAA

Entrada: Imagen de entrada

Salida: Ángulo de orientación detectado.

Aplicar el algoritmo de detección de uniones y el de agrupamiento de uniones.

Convertir los caminos en segmentos rectos.

Para cada segmento hacer

Para cada ángulo posible ($[-45, 45]$) hacer

Calcular la evidencia del segmento con el ángulo actual mediante la Ecuación 4.6 y utilizando el ángulo de la normal del segmento. Multiplicar la evidencia por la longitud del segmento.

Acumular la evidencia para el ángulo en concreto.

FinPara

FinPara

Devolver el ángulo con valor máximo de evidencia.

Figura 4.19: Algoritmo de obtención del ángulo de orientación de la cámara utilizando información de arista.

obtenido una media de 0.2 segundos, sin contar el tiempo de procesamiento de las aristas. Las Figuras 4.20 a 4.25 muestran ejemplos de aplicación del algoritmo propuesto. Las imágenes inferiores muestran el resultado del agrupamiento una vez aplicado el método para detectar segmentos rectos.



Figura 4.20: Resultado obtenido en imagen de exterior. El ángulo de orientación de la cámara obtenido fue de -15° .



Figura 4.21: Resultados obtenidos en imágenes de exterior. El ángulo de orientación de la cámara obtenido fue de 22° .

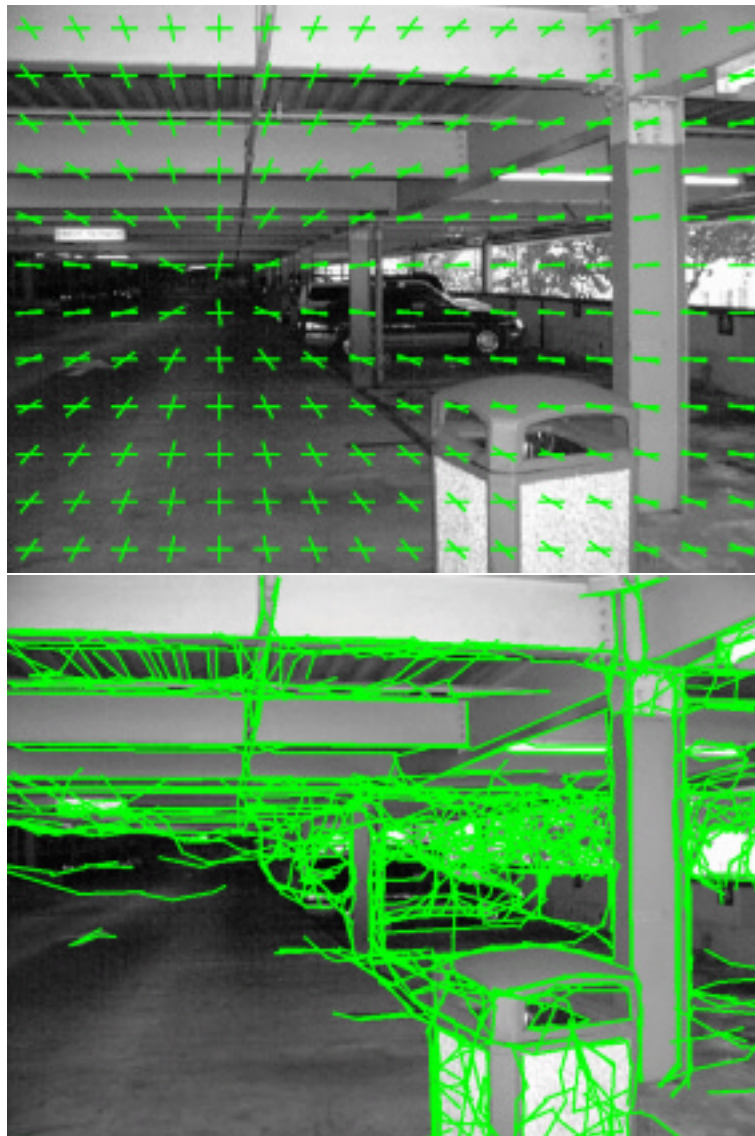


Figura 4.22: Resultados obtenidos en imágenes de exterior. El ángulo de orientación de la cámara obtenido fue de 9° .

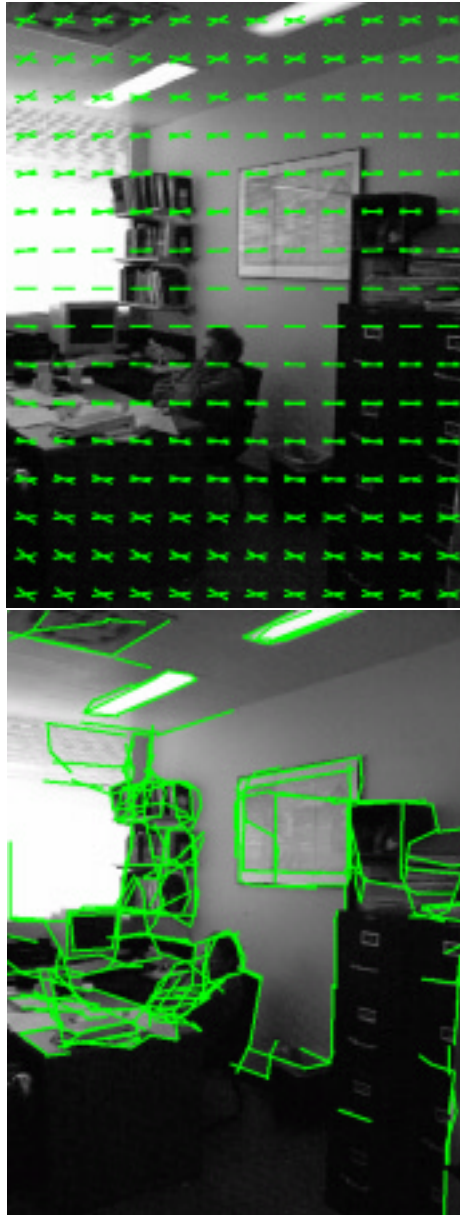


Figura 4.23: Resultados obtenidos en imágenes de interior. El ángulo de orientación de la cámara obtenido fue de 34° .

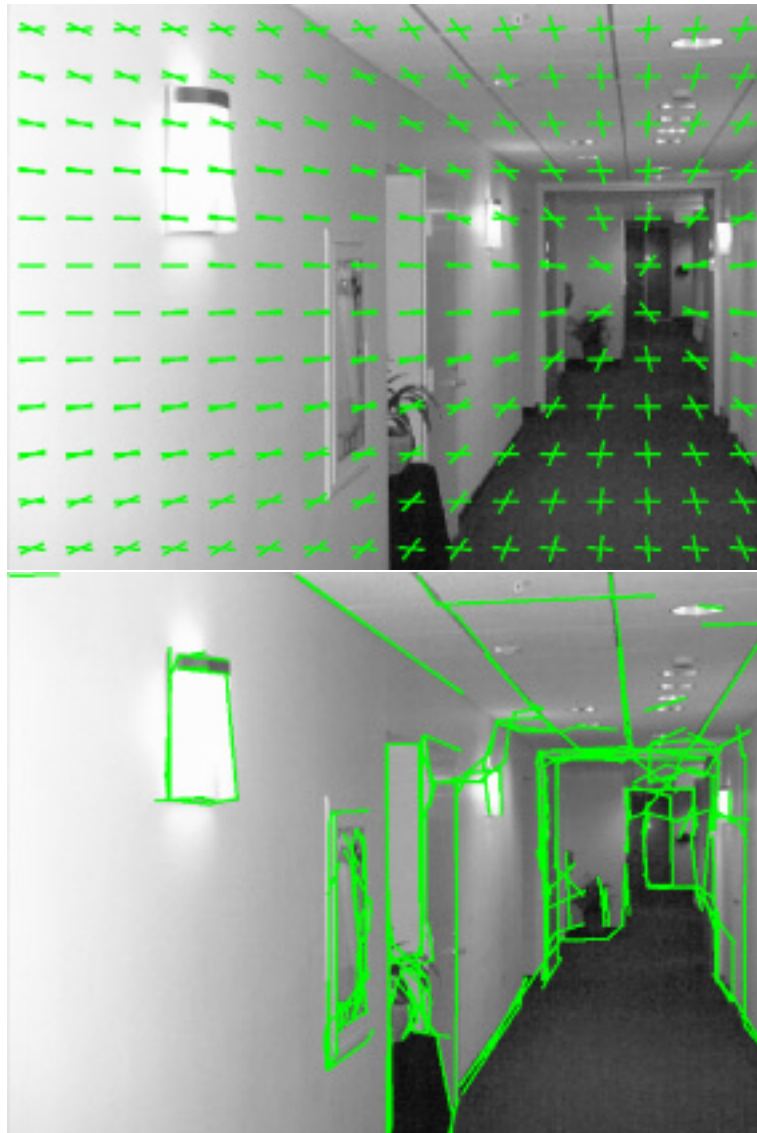


Figura 4.24: Resultados obtenidos en imágenes de interior. El ángulo de orientación de la cámara obtenido fue de -13° .

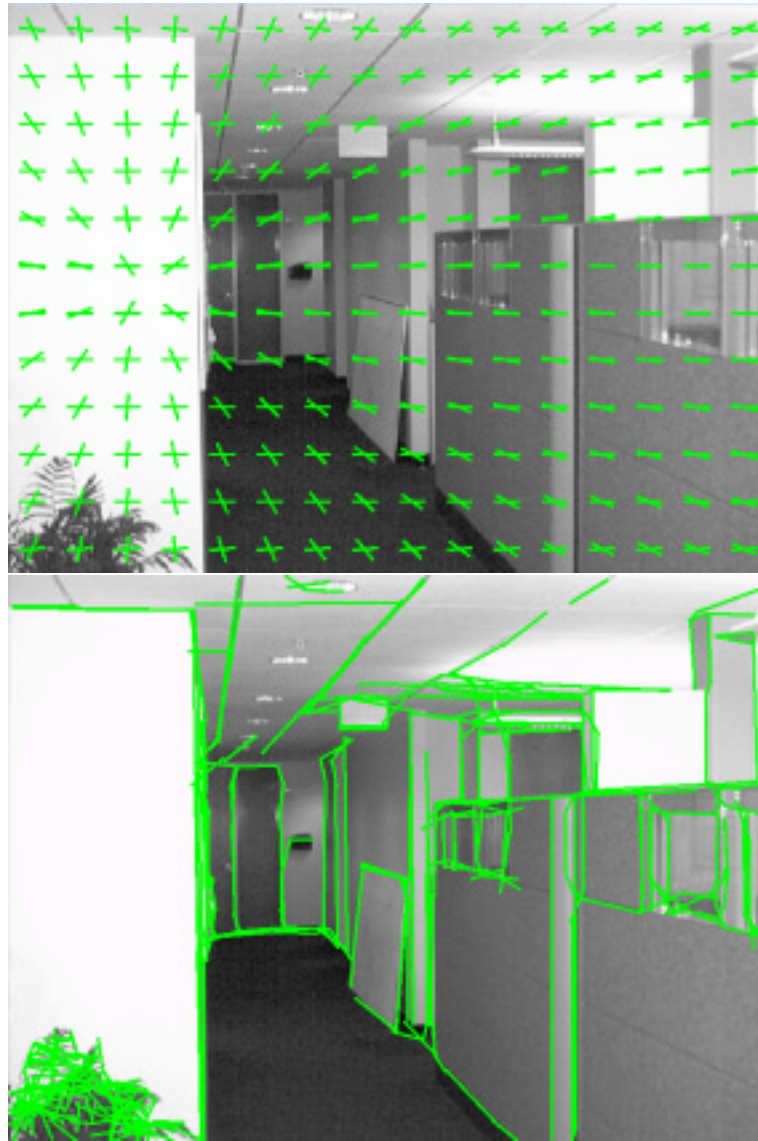


Figura 4.25: Resultado obtenido en imagen de interior. El ángulo de orientación de la cámara obtenido fue de 13° .

4.4.5 Etiquetado de los segmentos

Una posible aplicación del cálculo del ángulo de orientación es la eliminación de aquellas aristas que no se alineen con el mundo de Manhattan. En el trabajo de [Coughlan y Yuille, 1999] se obtiene el conjunto de puntos que no se etiquetan como pertenecientes a una de las tres orientaciones. En nuestro caso, hemos eliminado aquellas aristas que no cumplan las restricciones de Manhattan. En las Figuras 4.26 y 4.27 se muestra el resultado de eliminar estas aristas. Las imágenes de arriba muestran aquellas aristas no alineadas con el mundo de Manhattan. Las de debajo sí que lo están.

4.5 Discusión

El objetivo en este capítulo ha sido la obtención del ángulo de orientación de la cámara. Primero hemos presentado las características básicas de un novedoso método de detección de dicho ángulo que utiliza información bayesiana de pertenencia de un determinado píxel a una arista. Hemos realizado experimentos para comprobar su funcionamiento, proponiendo mejoras para solventar algunos problemas. Se ha comprobado que el tiempo de ejecución es excesivo para el dominio de trabajo de esta aplicación, la robótica móvil. Por ello hemos adaptado este método para, siguiendo también una formulación bayesiana, tomar como fuente el resultado del agrupamiento descrito en el capítulo anterior. El nuevo método presenta una mejora sustancial en cuanto a tiempo de computación y el error medio cometido es prácticamente despreciable. Estos experimentos muestran que el agrupamiento de uniones puede ser una estructura útil para la simplificación de ciertas tareas en robótica móvil.

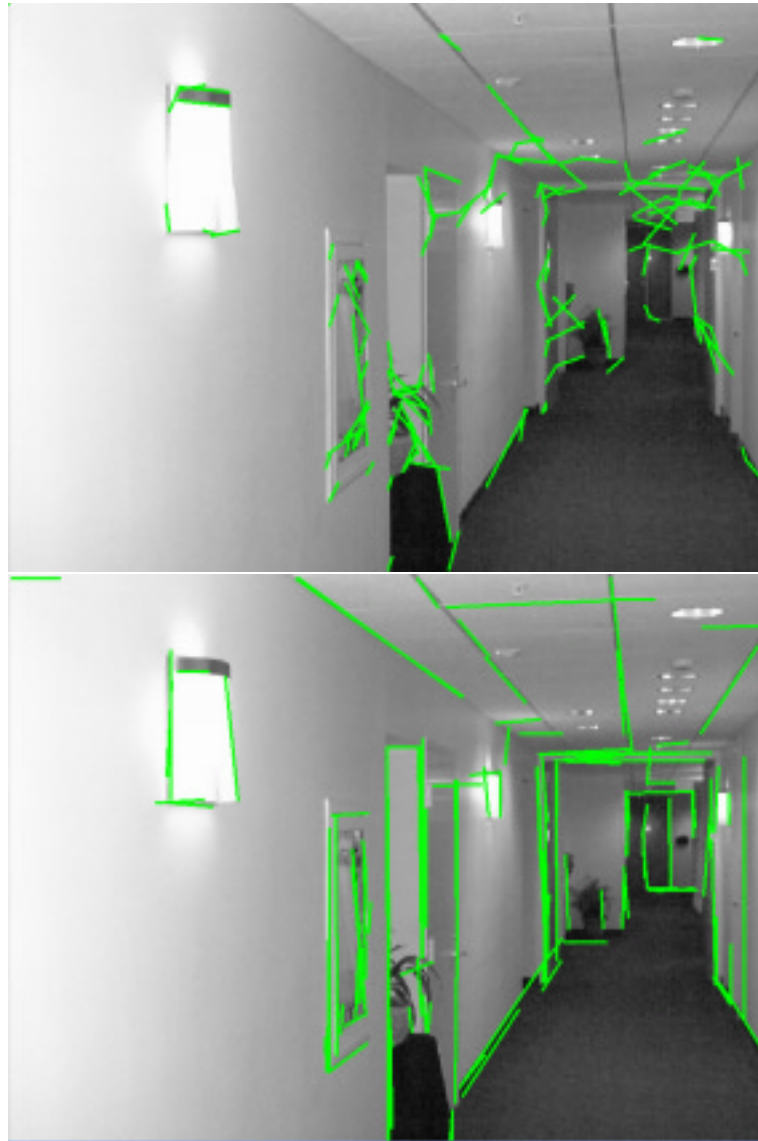


Figura 4.26: Separación de aristas dependiendo de si están alineadas con el mundo de Manhattan o no.

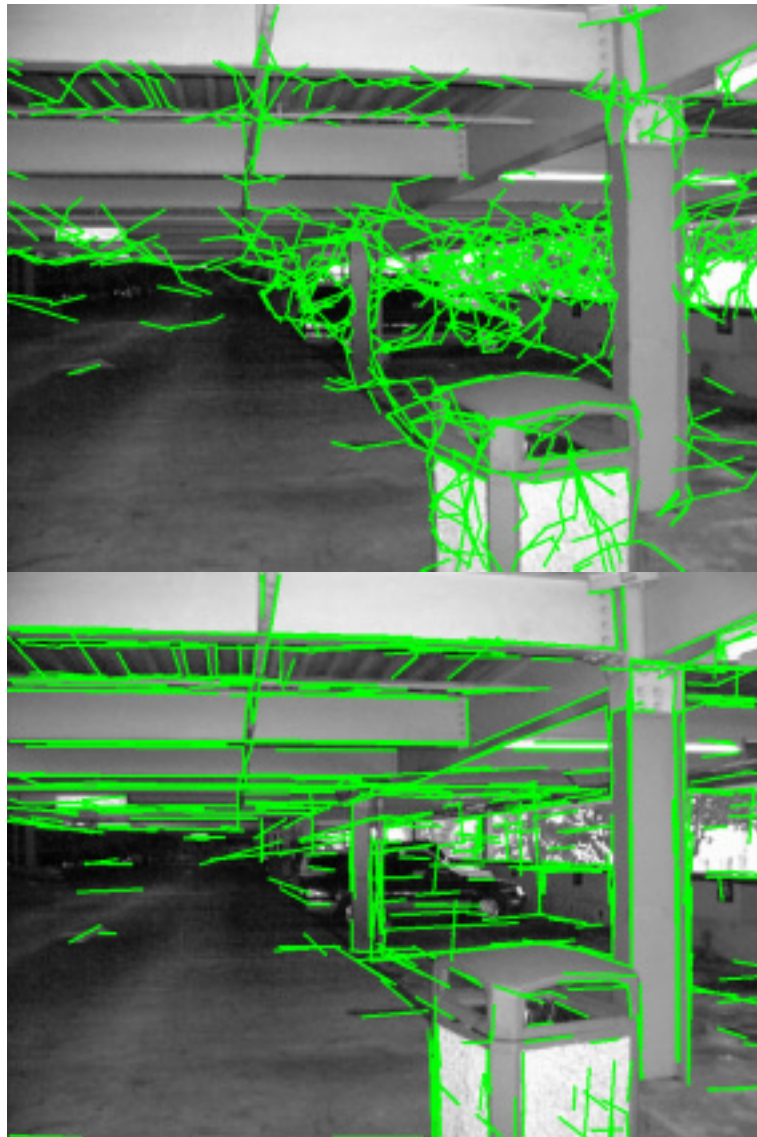


Figura 4.27: Separación de aristas dependiendo de si están alineadas con el mundo de Manhattan o no.

