

Capítulo 2

DETECCIÓN Y CLASIFICACIÓN DE UNIONES

El primer paso en nuestra propuesta es la detección y clasificación de uniones. Este proceso lo podemos dividir en dos fases bien diferenciadas: localización del centro de la unión (detección) y localización de los límites de las secciones angulares que forman la unión (clasificación). La calidad de la fase de clasificación depende principalmente del error cometido en la fase de detección. Debemos, pues, utilizar un detector de puntos esquinas lo más preciso posible. En la primera parte de este capítulo realizaremos una revisión de los detectores de puntos característicos previamente definidos para a continuación realizar una comparativa en profundidad entre dos de estos métodos que proporcionan robustez y rapidez: el operador de SUSAN y el de Nitzberg.

En la Sección 2.3 especificamos el problema de la clasificación de uniones en términos de la minimización de una función de energía. Para ello nos basamos en el método Kona [Parida *et al.*, 1998] y proponemos dos nuevos métodos de clasificación. A continuación realizamos una comparación entre nuestros propuestos y Kona. Finalizamos este capítulo con una discusión y algunas posibles extensiones de este trabajo.

2.1 Métodos de detección de puntos característicos

Una imagen contiene una gran cantidad de datos la mayoría de los cuales proporciona muy poca información para interpretar la escena. Un sistema que incorpore visión artificial debe, en un primer paso, extraer de la forma más eficaz y

robusta posible determinadas características que nos proporcionen la máxima información posible. Estas características deben cumplir, entre otras, las siguientes condiciones:

- Su extracción a partir de la imagen no debe suponer un coste excesivo al sistema en el cual está integrado. El tiempo total de extracción debe ser lo más pequeño posible.
- Su localización debe ser muy precisa. El error cometido en la estimación de las características también debe ser lo más pequeño posible.
- Deben ser robustas y estables. Deberían permanecer a lo largo de una secuencia.
- Contendrán la máxima información posible de la escena, es decir, debemos ser capaces de extraer información de tipo geométrico a partir de ellas.

Las aristas, esquinas y uniones satisfacen estas condiciones. Aparecen de forma natural en la mayoría de escenarios tanto naturales como artificiales (construidos por el hombre). Sin embargo, la obtención de las aristas suele ser fuente de muchos errores. Pongamos por ejemplo el algoritmo de detección de aristas más comúnmente utilizado, el algoritmo de Canny [Canny, 1986]. Este algoritmo combina un operador diferencial con un filtro gaussiano. Variando el parámetro de varianza del filtro que utiliza este algoritmo, podemos tener el siguiente comportamiento (ver Figura 2.1): Cuando disminuimos la varianza aparecen un gran número de aristas, la mayoría de las cuales no aportan mucha información. La localización de las aristas en este caso es muy buena. Al aumentar la varianza nos quedamos con aquellas aristas que se encuentran en una zona con gran contraste. Sin embargo, la localización en este caso no es exacta y el tiempo de procesamiento aumenta. Este comportamiento sucede también en la mayoría de los filtros de detección de aristas. También el uso de derivadas de la imagen para el cálculo de aristas supone un error adicional por el ruido que se produce en el proceso de captación de la imagen, tal como se deduce del trabajo de [Micheli *et al.*, 1989].

Podemos definir una unión como la intersección de dos o más aristas en un determinado punto de la imagen. Debemos distinguir una unión de un punto esquina o *punto característico* (utilizaremos ambos términos indistintamente). Un punto esquina es un punto de la imagen donde el contorno de una arista tiene curvatura alta y/o el centro de una unión.

Desde el punto de vista computacional, se han propuesto una serie de enfoques para la detección de este tipo de características bidimensionales. Podemos clasificar estos enfoques en dos grupos principales:

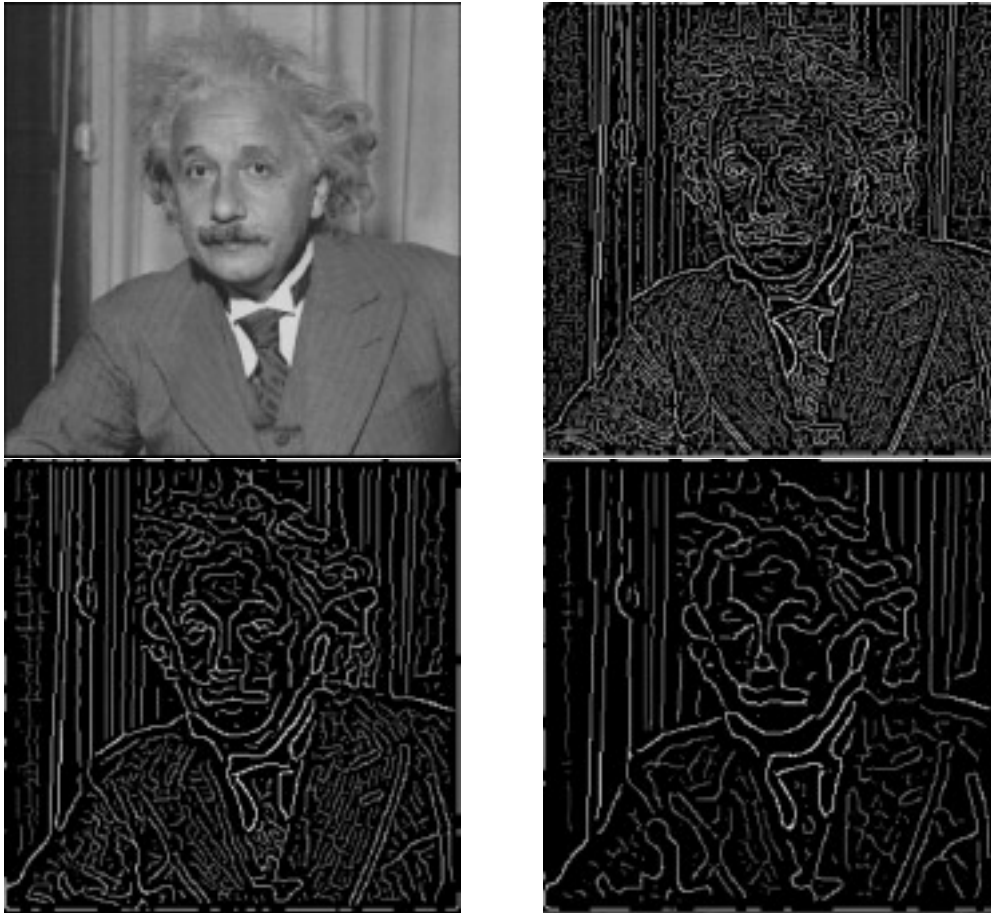


Figura 2.1: Aplicación del filtro de Canny modificando la varianza. De izquierda a derecha y de arriba a abajo se han utilizado varianzas de 1, 2 y 3.

1. Métodos que obtienen las aristas de la imagen mediante algún método de detección de aristas como el de Canny, para, a continuación, detectar puntos de cruce entre aristas o con un cambio sustancial en la dirección de la arista. Estos puntos se clasifican como puntos esquina o unión.
2. El resto de métodos trabajan directamente sobre imágenes de gris, es decir,

no se infieren a partir de la extracción de aristas.

Ciertos métodos clasifican las esquinas detectadas, encontrando los límites angulares donde se encuentran situadas las separaciones de las distintas regiones que confluyen en ellas. En las siguientes subsecciones se comentan algunos de estos métodos empleados tanto para la detección de esquinas como para su clasificación. Podemos consultar [Giraudon y Deriche, 1991] y [Smith y Brady, 1997] para una revisión de los métodos de detección de puntos esquina.

2.1.1 Puntos característicos a partir de aristas

Todos los métodos de obtención de esquinas y/o uniones dentro de este grupo tienen en común una primera fase de extracción de aristas. Algunos métodos de extracción de aristas pueden ser [Torre y Poggio, 1986], [Canny, 1986] o [Smith y Brady, 1997]. Una vez obtenida las aristas estas se representan mediante un código de cadena. Un código de cadena permite expresar, por ejemplo, el cambio de dirección de un píxel al siguiente o la discretización de la tangente para todos los puntos de la arista. Ahora el objetivo consiste en encontrar puntos de máxima curvatura dentro del código de cadena. Este puede ser llevado a cabo de dos posibles formas:

1. Calcular, en cada punto del código de cadena, el valor de la tangente utilizando las coordenadas de los píxeles [Asada y Brady, 1986], [Han *et al.*, 1989]. En [Medioni y Yasumoto, 1987] utilizan un B-spline cúbico para detectar los puntos esquina, ajustando este spline a los puntos de la imagen. También podemos detectar los puntos de máxima curvatura calculando las diferencias entre los códigos de cadena dentro de una ventana. En [Bell y Pau, 1990] se calcula el siguiente valor para todos los puntos del código cadena:

$$\frac{1}{SCAN/2} \sum_{i=1}^{SCAN/2} [codigo[i] - codigo[i + SCAN/2]]$$

siendo SCAN el tamaño de la ventana. Si este valor supera un cierto umbral se considera que existe un punto de máxima curvatura dentro de la ventana

2. Estimar la tangente en un punto del código de cadena mediante las derivadas parciales de la imagen en ese punto [Deriche y Faugeras, 1990].

Existen propuestas que detectan los puntos esquinas haciendo uso del espacio escala [Rattarangsi y Chin, 1992], [Asada y Brady, 1986] y [Lindeberg y Li, 1997]. En general, la eficiencia de estos métodos dependen directamente de la calidad del

método empleado para la obtención de aristas: si este último no localiza correctamente los puntos de arista difícilmente podremos detectar puntos de esquina de forma exacta. Además se añade un tiempo extra de procesamiento que en ciertos sistemas puede ser prohibitivo.

2.1.2 Puntos característicos a partir de la propia imagen

Existen otros métodos que obtienen los puntos de máxima curvatura de una imagen utilizando directamente los valores de la imagen, sin realizar el paso previo de obtención de aristas. Estos métodos definen una medida que suelen denominar de “esquinidad” la cual se calcula para todos los puntos de la imagen. Cuando esta medida supera un cierto umbral se considera que el punto es una esquina. La mayoría de estos métodos utilizan operadores diferenciales.

Uno de los primeros métodos para la obtención de puntos característicos se describe en [Moravec, 1977]. Dada una ventana rectangular, este operador calcula el valor de cambio de la intensidad de la imagen cuando desplazamos ligeramente esta ventana. Una esquina viene caracterizada por elevados cambios de intensidad. Es altamente sensible a ruido.

En [Beaudet, 1978] se propone un operador que calcula el determinante de una matriz y queda definido como:

$$DET = I_{xx}I_{yy} - I_{xy}^2$$

donde I_{xy} es la derivada parcial de la imagen con respecto al eje x y después con respecto al eje y , y I_{xx} y I_{yy} es la segunda derivada con respecto a x e y , respectivamente. Este operador presenta ciertas características de curvatura interesantes. Las esquinas se obtienen aplicando este operador y realizando una umbralización. En un posterior trabajo [Deriche y Giraudon, 1993] se amplía este método incorporando conocimiento de arista y multiresolución y se realiza un estudio analítico del modelo de esquina. Posteriormente, en [Harris y Stephens, 1988] se continúa investigando sobre las propiedades del determinante antes mencionado. En este último trabajo se presenta un nuevo método de detección de puntos característicos que presenta robustez frente al ruido. La definición de este método se realiza de manera similar al operador de Moravec, pero calculando las derivadas de primer orden de la imagen. La localización de los puntos no es del todo exacta. Comentar que también este método sirve como base para la definición del operador de Nitzberg, comentado más adelante.

En [Kitchen y Rosenfeld, 1982] se presentan tres métodos que obtienen esquinas y que, de manera esquemática, se detallan a continuación:

1. Uso del producto del gradiente de intensidad y del gradiente de la dirección en un punto de la imagen como una medida de esquinidad.

2. Uso de la diferencia entre las direcciones de gradiente de puntos vecinos, los cuales son perpendiculares a la dirección gradiente del punto, como una medida de esquinidad.
3. En una vecindad 3×3 localizar 2 puntos A y B con valor de gris similar al punto C bajo consideración. La diferencia en la dirección entre los vectores CA y CB es una medida de esquinidad.

Una vez aplicada esta medida realizan una supresión de falsos candidatos para quedarse con un único punto en la vecindad. Un enfoque similar se sigue en [Zuniga y Haralick, 1983] donde se detectan esquinas en aquellos puntos donde se producen cambios significativos de gradiente a lo largo de las aristas.

En los trabajos de [Koenderink y Richards, 1988], [Mehrota y Nichani, 1990] y [Dreschler y Nagel, 1982] se definen operadores de cálculo de curvatura sobre una determinada vecindad haciendo uso de derivadas de primer, segundo y tercer orden de alguna función Gaussiana. En [Rangarajan *et al.*, 1989] utilizan 12 máscaras distintas para detectar distintas clases de esquinas. Hacen uso de un detector de aristas para eliminar falsos candidatos. En [Vasselle *et al.*, 1994] se introduce el concepto de espacio escala y determina la localización de esquinas estudiando las características de éstas en dicho espacio. En [Alvarez y Morales, 1997] también se basan en el concepto de espacio escala para determinar la localización de una esquina.

Por último, el enfoque de SUSAN [Smith y Brady, 1997] proporciona un método rápido, robusto frente al ruido (no utiliza derivadas de la imagen) y la localización de los puntos característicos presenta buenos resultados. Este algoritmo se describe en la Sección 2.2.2.

2.2 Comparativa de detectores de puntos característicos

Revisadas las propuestas previas, optamos por descomponer en dos subproblemas. La calidad de un detector de uniones viene determinado por la localización exacta del centro de la unión. Por ello, vamos a analizar dos métodos que presentan buenas características en cuanto a rapidez y error cometido. Describiremos ambos métodos, realizando a continuación una comparación entre ellos.

2.2.1 El operador de Nitzberg

En [Nitzberg *et al.*, 1993] se propone un operador que permite calcular la magnitud del gradiente y su orientación, así como la evidencia de presencia de puntos esquina. Para ello se utiliza la información de gradiente en una determinada vecindad y esto se combina con un operador de gradiente para reducir ruido. Lo primero que debemos hacer es calcular las derivadas parciales en x e y en toda la imagen. Estas se pueden estimar de la siguiente manera:

$$\begin{aligned} I_x(x, y) &= (I(x+1, y) - I(x-1, y))/2 \\ I_y(x, y) &= (I(x, y+1) - I(x, y-1))/2 \\ \nabla I(x, y) &= \begin{pmatrix} I_x(x, y) \\ I_y(x, y) \end{pmatrix} \end{aligned}$$

A continuación se define la siguiente matriz 2×2 :

$$Q(\mathbf{x}) = \int d\mathbf{x}' \rho(\mathbf{x} - \mathbf{x}') \nabla I(\mathbf{x}') \nabla I(\mathbf{x}')^T = \int d\mathbf{x}' \rho(\mathbf{x} - \mathbf{x}') \begin{pmatrix} I_x^2(\mathbf{x}') & I_x I_y(\mathbf{x}') \\ I_x I_y(\mathbf{x}') & I_y^2(\mathbf{x}') \end{pmatrix} \quad (2.1)$$

donde $\nabla I(\mathbf{x})$ es un vector columna y $\rho(\mathbf{x})$ es una función de ponderación decreciente con valor máximo en 0.

La matriz $Q(\mathbf{x})$ tiene un autovalor mayor $\lambda_1(\mathbf{x})$, asociado al autovector $e_1(\mathbf{x})$ y un autovalor menor $\lambda_2(\mathbf{x})$. La magnitud del gradiente viene dada por $I_e(\mathbf{x}) = \lambda_1(\mathbf{x})$ y su orientación por la orientación del autovector asociado $\theta_I(\mathbf{x}) = e_1(\mathbf{x})$. Para el cálculo de los puntos esquina se utiliza el valor del segundo autovalor o autovalor menor, $I_c(\mathbf{x}) = \lambda_2(\mathbf{x})$.

La función de ponderación $\rho(\mathbf{x})$ nos permite suavizar en mayor o menor medida. En los experimentos, y por simplicidad, hemos elegido la siguiente función:

$$\rho(\mathbf{x}) = \begin{cases} \frac{1}{w^2} & \text{si } -\frac{w}{2} \leq x \leq \frac{w}{2} \text{ y } -\frac{w}{2} \leq y \leq \frac{w}{2} \\ 0 & \text{en otro caso.} \end{cases}$$



Figura 2.2: Resultado de calcular los autovalores de una imagen. Arriba derecha: autovalor mayor (aristas), debajo izquierda: autovalor menor (esquinas), debajo derecha: puntos esquina detectados aplicando una umbralización y una búsqueda de máximo locales.

fiándose $w = 4$. Para obtener los puntos esquina realizaremos una umbralización utilizando un valor prefijado, seguido de una supresión de mínimos locales. En la Figura 2.2 se muestra el resultado de calcular los autovalores de una imagen.

2.2.2 Descripción de SUSAN

En [Smith y Brady, 1997] se presenta un nuevo enfoque para el procesamiento de imágenes a nivel bajo. Se busca introducir un aspecto clave en el desarrollo de un sistema de visión artificial: rapidez. Este nuevo enfoque es guiado por la premisa de obtener un sistema que extraiga características de la imagen (aristas, esquinas, uniones) en el menor tiempo posible. Por supuesto, el error cometido en la localización de estas características debe ser mínimo. Para ello evitan, en la medida de lo posible, el uso de derivadas para la obtención de las características bidi-

mensionales. En [Micheli *et al.*, 1989] se demuestra que la diferenciación es un problema definido de forma ambigua (*ill-posed*). Esto es debido a que la diferenciación amplifica las componentes de alta frecuencia del ruido lo que implicará una pérdida de estabilidad o continuidad. El problema puede ser transformado en uno *well-posed* aplicando filtros paso bajo para eliminar las componentes de alta frecuencia, con la consecuente pérdida en la definición de los bordes.

Principio de SUSAN

El elemento principal de SUSAN es un filtro no lineal para la detección de las características. Este filtro calcula el área (área SUSAN) de la imagen (dentro del dominio definido por el filtro) que tiene el mismo nivel de gris que el centro. Si, por ejemplo, utilizamos un dominio circular como filtro y trabajamos sobre una imagen discretizada, el filtro acumula el número de puntos dentro del círculo con un nivel de gris similar al del centro. En la Figura 2.3 se puede observar cuál es el resultado de aplicar el filtro a toda la imagen. Los valores mostrados dentro del círculo indican el valor que proporciona el filtro en el cual se basa la detección de los puntos esquina. En la figura de la derecha se representa el resultado de calcular el área SUSAN para la parte superior izquierda del rectángulo (la figura está invertida). También podemos observar que las aristas en la imagen y las esquinas presentan mínimos locales que serán los que se deberán extraer según deseemos una característica u otra.

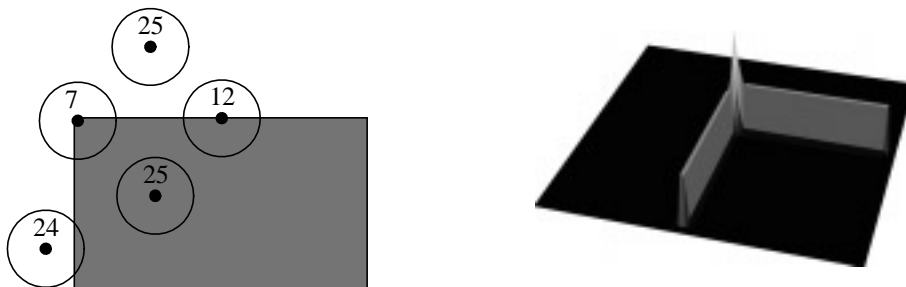


Figura 2.3: Área SUSAN.

El primer paso de cálculo del área SUSAN es idéntico tanto para la detección de aristas como para los puntos esquinas. Este paso consiste en calcular el área SUSAN, la cual se determina en cada punto $\mathbf{x}_0 = (x_0, y_0)$ de la imagen de la siguiente

forma:

$$n(\mathbf{x}) = \sum_{\mathbf{x} \in \mathcal{D}} c(\mathbf{x}, \mathbf{x}_0) \quad \text{siendo} \quad c(\mathbf{x}, \mathbf{x}_0) = \begin{cases} 1 & \text{si } |I(x, y) - I(x_0, y_0)| \leq t \\ 0 & \text{en otro caso} \end{cases}$$

donde \mathcal{D} es el dominio del círculo, $I(x, y)$ es el brillo de la imagen en el punto $\mathbf{x} = (x, y)$ y t es el umbral que indica el contraste de brillo permitido. Si t es igual a cero estamos teniendo en cuenta únicamente aquellos puntos con el mismo nivel de gris que el centro. Al aumentar t aumentamos el contraste tolerado, lo cual permite absorber puntos de ruido. Esto influye notablemente en el número de puntos esquina detectados, pues al disminuir el valor de t aumentamos dicho número, al ser más sensible a ruido.

El detector de aristas de SUSAN

Dada el área SUSAN las aristas se detectan comparando su valor con un umbral geométrico g que en este caso se asigna a $3n_{max}/4$, donde n_{max} es el valor máximo que n puede tomar (el área correspondiente al dominio del filtro). El valor de la respuesta inicial es:

$$R(\mathbf{x}_0) = \begin{cases} g - n(\mathbf{x}_0) & \text{si } n(\mathbf{x}_0) < g \\ 0 & \text{en otro caso} \end{cases}$$

Esta formulación nos proporciona una respuesta que es mayor cuanto menor sea el área SUSAN de la imagen. El valor de la normal de la arista suele ser necesaria en varias aplicaciones. Para calcular esta normal primero se calcula el centro de gravedad. El centro de gravedad es el centro del área SUSAN y se calcula de la siguiente manera:

$$\bar{\mathbf{x}}(\mathbf{x}_0) = \frac{\sum_{\mathbf{x} \in \mathcal{D}} \mathbf{x} c(\mathbf{x}, \mathbf{x}_0)}{\sum_{\mathbf{x} \in \mathcal{D}} c(\mathbf{x}, \mathbf{x}_0)}$$

La normal será el vector cuyo origen se sitúe en el centro del filtro y cuyo extremo se encuentre en el centro de gravedad. También suele ser necesario un adelgazamiento de las aristas y esto se realiza en una dirección perpendicular a la arista (en la misma dirección de la normal de la arista). La Figura 2.4 muestra el centro de gravedad (marcado en rojo) en distintos puntos de la imagen.

En la Figura 2.5 y siguientes se muestra un ejemplo de detección de aristas con SUSAN. El tiempo de procesamiento no varía al variar los parámetros y es de 0.06s por imagen, nueve veces menor que el procesamiento con el filtro de Canny. En las figuras se puede observar el efecto de variar el parámetro t (contraste). Al aumentar el factor t disminuimos el número de aristas detectadas, permaneciendo sólo aquellas con mayor cambio de intensidad.

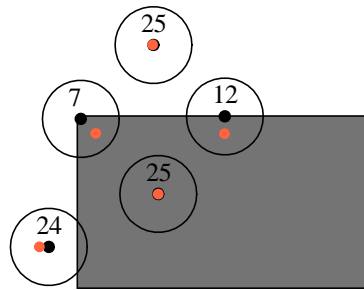


Figura 2.4: Centro de gravedad.

Figura 2.5: $t=10$.Figura 2.6: $t=20$.Figura 2.7: $t=30$.

El detector de esquinas de SUSAN

De manera idéntica al detector de aristas, el detector de esquinas calcula el área SUSAN de la imagen y también la compara con un umbral geométrico. Sin embargo, tal como se puede observar en la Figura 2.3, una esquina tendrá un valor SUSAN menor que las aristas por lo que este umbral se fija en $n_{max}/2$. Para eliminar falsos positivos se realizan los siguientes pasos:

1. Se calcula el centro de gravedad del área para el candidato. La distancia del centro de gravedad al centro del área determina si nos encontramos con un falso positivo: si esta distancia es corta, será un falso positivo.
2. Para que sea una esquina verdadera todos los puntos que descansan en una línea recta que une el centro del área con el centro de gravedad, deben ser parte del área SUSAN.
3. Aplican una supresión de mínimos locales, eliminando todos los mínimos

alrededor del candidato menor, dentro de una vecindad 5×5 .

En las Figuras 2.8 y 2.9 se muestran algunos resultados de esquinas detectadas con SUSAN.

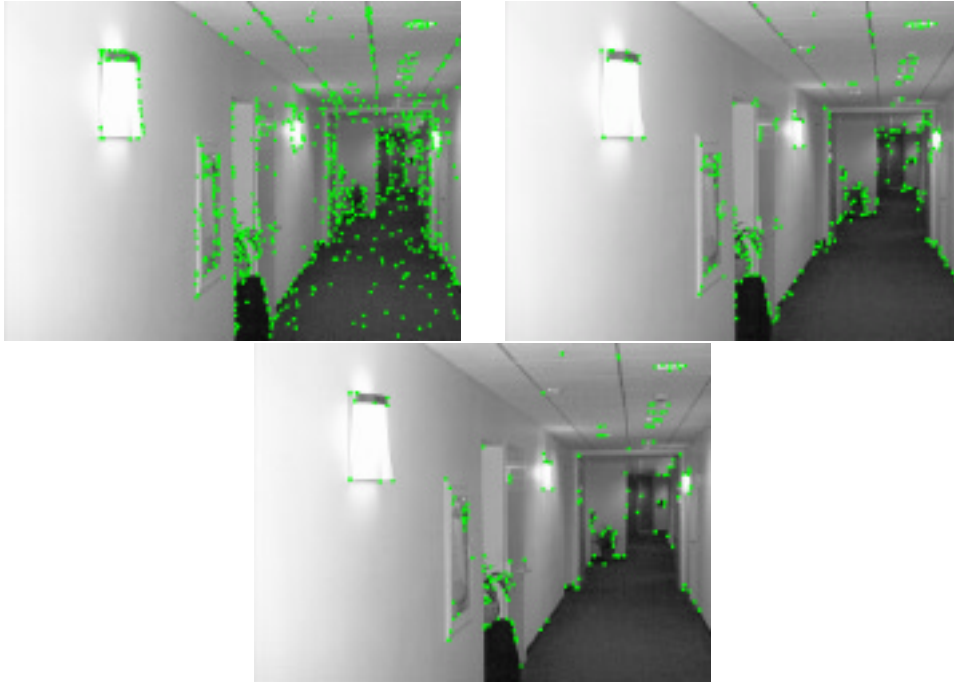


Figura 2.8: Puntos esquinas detectados por SUSAN. Arriba izquierda: $t=10$, arriba derecha: $t=20$, abajo: $t=30$.

2.2.3 Comparación de SUSAN y Nitzberg

Nuestro objetivo no es realizar una comparativa en profundidad, sino identificar efectos importantes que nos afecten a nuestro futuro trabajo. Se ha realizado un estudio comparativo del funcionamiento y la robustez de ambos operadores. Las Figuras 2.10 y 2.11 muestran resultados de aplicar el operador de SUSAN y de Nitzberg, respectivamente, variando el nivel de ruido en la imagen. En ambos experimentos se han seleccionado los mejores valores de los parámetros para ambos métodos.



Figura 2.9: Resultados al ir aumentando el parámetro de geometría (izquierda-derecha-abajo).

La primera gráfica que se muestra es la de la Figura 2.12 y muestra el error medio cometido por ambos operadores frente a la presencia de ruido. El error se mide en número de píxeles y nos da la distancia del centro real al centro detectado por el operador. Para las estadísticas sólo se han tenido en cuenta aquellos puntos esquina detectados que son reales y que no están producidas por ruido.

A continuación se han obtenido estadísticas del número de falsos positivos y falsos negativos en función del ruido. Los falsos positivos son aquellos puntos característicos detectados por el operador, pero que en realidad no aparecen en la imagen. Los falsos negativos son aquellos puntos que aparecen en la imagen y sin embargo no son detectados. El número medio de falsos negativos es despreciable en ambos operadores: del orden de 2%. Un dato observado en este estudio es que al aumentar el ruido en la imagen, se detectan mejor los puntos característicos. De hecho, cuando se supera un umbral del 20% de ruido gaussiano ambos operadores son capaces de detectar el 100% de los puntos característicos en la imagen. En la

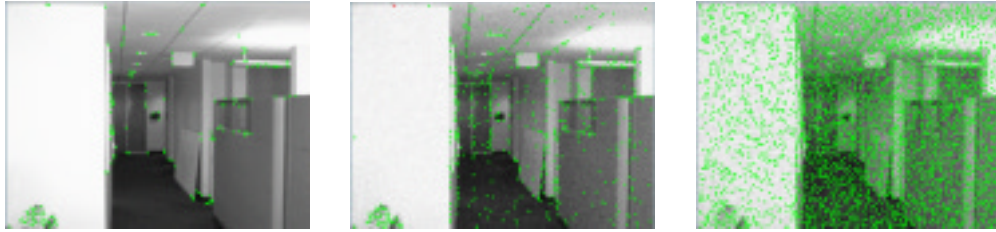


Figura 2.10: Ejemplo de ejecución del operador de SUSAN variando el ruido en la imagen. Izquierda: sin ruido, central: ruido gaussiano de varianza 20, derecha: ídem varianza 50.



Figura 2.11: Ejemplo de ejecución del operador de Nitzberg variando el ruido en la imagen. Izquierda: sin ruido, central: ruido gaussiano de varianza 20, derecha: ídem varianza 50.

Figura 2.13 se muestra el porcentaje del número de puntos candidatos detectados que son falsos positivos.

Por último, hemos tomado tiempos de la ejecución de ambos métodos aumentando el tamaño de la imagen. Nuestras imágenes originales tienen tamaño 640x480. Hemos ido reduciendo el tamaño de la imagen original por un factor de 5, 4, 3 y 2. En la Figura 2.14 se muestra el tiempo de CPU de ambos métodos. Aquí el tamaño 5 indica el tamaño original de las imágenes.

El operador de SUSAN es más rápido que el de Nitzberg cuando la imagen supera el tamaño de 320x240. Si nos fijamos en la robustez, ambos presentan las mismas características de robustez. Sin embargo, será necesario el uso de un filtro que elimine ruido, antes de aplicar cualquiera de estos filtros. El error medio cometido (distancia del centro real al detectado) ha sido de 1 píxel, valor que perfectamente se puede asumir en la mayoría de las aplicaciones de visión.

Sin embargo, el operador de Nitzberg no necesita ajustar tantos parámetros como el de SUSAN. La selección del parámetro de contraste en el operador de SUSAN es bastante crítica. Por todo esto, nos decantamos por el operador de Nitzberg como base para nuestro detector de uniones. Por otro lado, cuando la

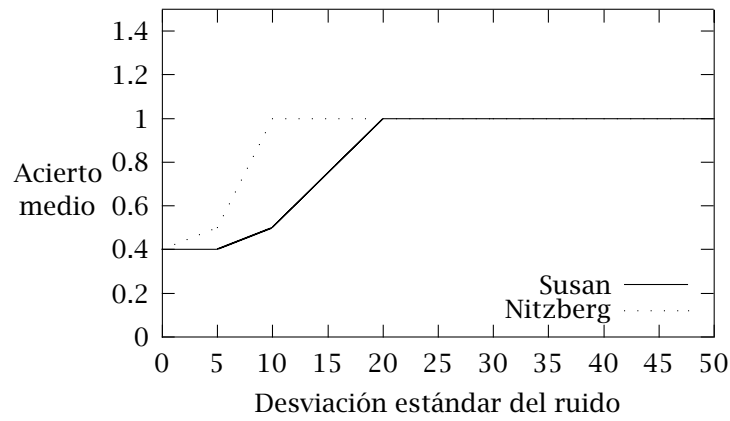


Figura 2.12: Comparación del error medio cometido por los operadores de SUSAN y Nitzberg.

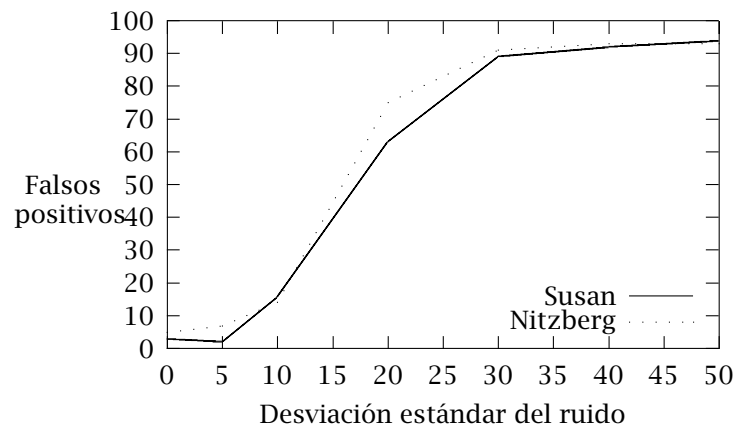


Figura 2.13: Comparación del número de falsos positivos detectados por los operadores de SUSAN y Nitzberg.

aplicación requiera el uso de imágenes de gran tamaño o definición, y también se busque rapidez, recurriremos al método SUSAN.

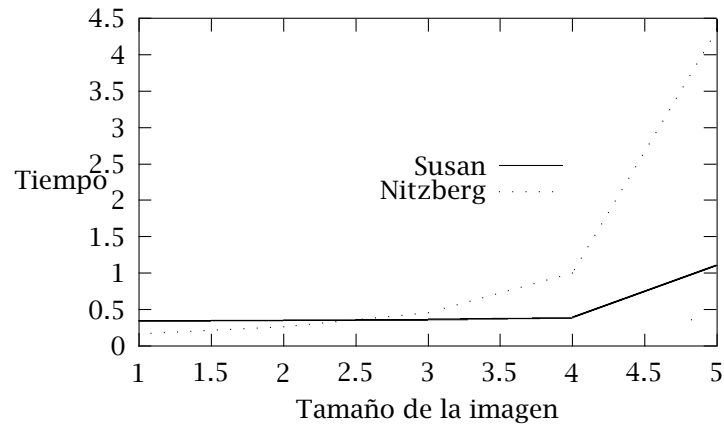


Figura 2.14: Comparación de tiempos de ejecución frente al tamaño de la imagen de los operadores de SUSAN y Nitzberg.

2.3 Definición de uniones mediante Kona

Para clasificar uniones hemos seleccionado el método Kona. Este método define una unión mediante un conjunto de parámetros. Se trata de encontrar los valores de los parámetros que mejor definan los datos de la imagen. La filosofía de trabajo es la misma que la de las plantillas deformables. También la clasificación de uniones se separa en dos fases: detección del centro y clasificación de los límites angulares, un enfoque que se ajusta a nuestra propuesta. Siguiendo la formulación paramétrica del método Kona [Parida *et al.*, 1998] una unión viene definida por (ver Figura 2.15):

- Centro de la unión: x_c, y_c .
- Radio exterior R_1 y radio interior R_0 de la unión. El radio interior es utilizado para evitar el ruido en la proximidad del centro de la unión.
- Número de secciones angulares de la unión: M .
- Los ángulos en los que se encuentran situados los límites de las secciones angulares: $\theta_1, \theta_2, \dots, \theta_M$.
- Los valores en nivel de gris de las secciones angulares: T_1, T_2, \dots, T_M .

Para encontrar los valores de los parámetros que definen una unión se utiliza una plantilla cuyo comportamiento se ajusta al principio de descripción mínima

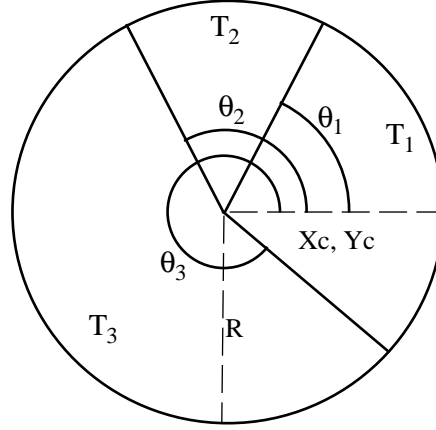


Figura 2.15: Parámetros de la unión.

longitud (MDL) intentándose encontrar el mínimo número de secciones angulares que mejor describen la unión. Para ello se define la siguiente función de energía:

$$E = \mathcal{D} + \lambda \mathcal{G}$$

donde \mathcal{D} es una medida de la distancia de los datos a la plantilla usando la norma L^2 :

$$\mathcal{D} = \int_{R_0}^{R_1} \int_0^{2\pi} |\mathcal{I}(r, \theta) - T(\theta)|^2 dr d\theta$$

\mathcal{G} es una medida de la distancia del gradiente:

$$\mathcal{G} = \int_{R_0}^{R_1} \int_0^{2\pi} |\nabla \mathcal{I}(r, \theta) - \nabla T(\theta)|^2 dr d\theta$$

Separando las componentes angular y radial de \mathcal{G} tenemos:

$$\mathcal{G} = \mathcal{A} e_\theta + \mathcal{R} e_r$$

$$\mathcal{A} = \int_{R_0}^{R_1} \int_0^{2\pi} \frac{1}{r^2} \left(\frac{\partial \mathcal{I}}{\partial \theta} - \frac{\partial T}{\partial \theta} \right)^2 dr d\theta$$

$$\mathcal{R} = \int_{R_0}^{R_1} \int_0^{2\pi} \left(\frac{\partial \mathcal{I}}{\partial r} \right)^2 dr d\theta$$

La función de energía global queda:

$$E = \lambda \mathcal{R} + (\mathcal{D} + \lambda \mathcal{A})$$

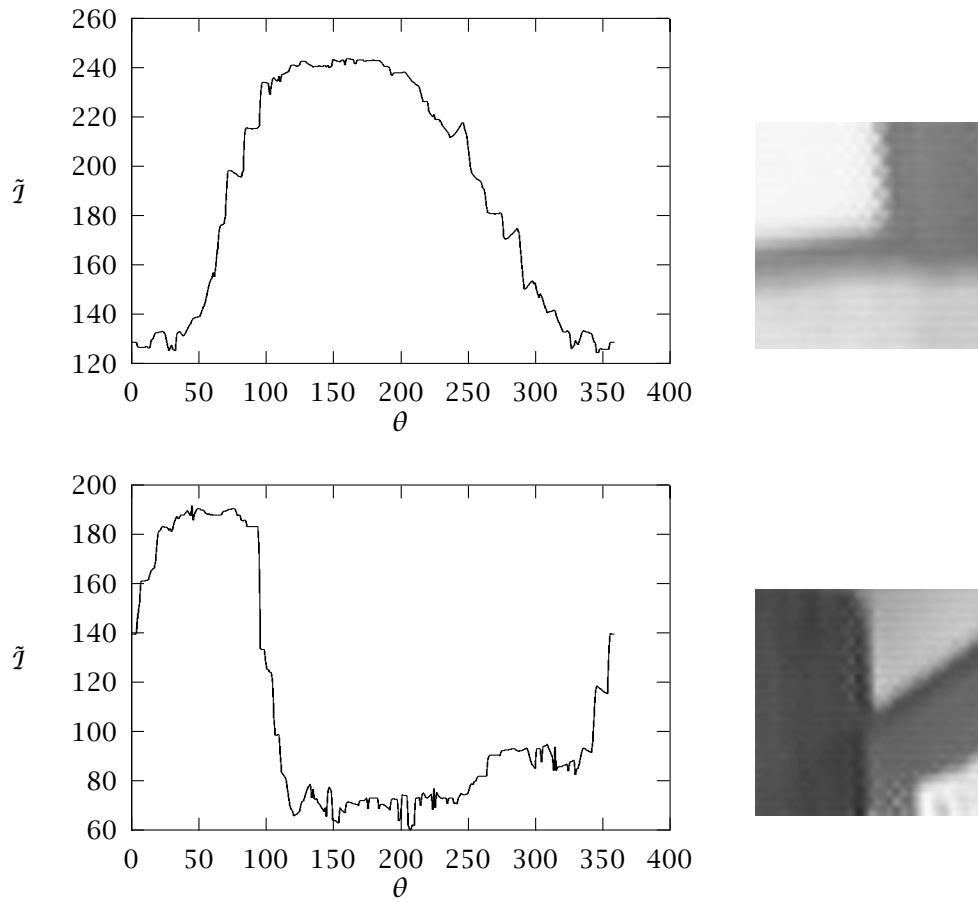


Figura 2.16: Valor acumulado de los radios en imágenes reales.

La función \mathcal{R} se utiliza para localización del centro de la unión y para obtener los radios R_1 y R_0 . Para calcular el resto de parámetros de la unión (número de secciones angulares, límites de dichas secciones y valores de gris) utilizan los otros dos términos de la función E :

$$E = \lambda \mathcal{R} + (\mathcal{D} + \lambda \mathcal{A}) = \lambda \mathcal{R} + \tilde{E}$$

$$\tilde{E} = \mathcal{D} + \lambda \mathcal{A} = \mathcal{F} + \mathcal{V}$$

donde

$$\mathcal{F} = \int_{R_0}^{R_1} \int_0^{2\pi} \left[\mathcal{I}^2(r, \theta) + \lambda \left(\frac{\partial \mathcal{I}(r, \theta)}{\partial \theta} \right)^2 \right] dr d\theta$$

$$\mathcal{V} = \int_{R_0}^{R_1} \int_0^{2\pi} \left[T^2(\theta) - 2T(\theta)\mathcal{I}(\theta) + \lambda \left(\frac{\partial \mathcal{T}^2}{\partial \theta} - 2 \frac{\partial \mathcal{I}}{\partial \theta} \frac{\partial \mathcal{T}}{\partial \theta} \right) \right] dr d\theta$$

\mathcal{F} no depende de los valores a estimar por lo que se descarta. Aproximan \mathcal{V} de forma numérica. Para ello utilizan el valor de intensidad del ángulo θ , $\tilde{\mathcal{I}}_\theta$ que definimos a continuación.

Intensidad angular

Un primer paso es realizar un cambio de dominio. Pasamos de un dominio circular alrededor de un punto candidato (centro de la unión), a otro lineal en función del ángulo. Este cambio se realiza calculando el valor medio de los puntos de la imagen a lo largo del radio situado en un determinado ángulo θ , $\tilde{\mathcal{I}}_\theta$. Este valor se obtiene, de forma discreta, tal como ilustra la Figura 2.17. Cada intervalo (su longitud) se multiplica por el valor del punto de la imagen correspondiente:

$$\tilde{\mathcal{I}}_\theta = \frac{1}{(R_1 - R_0)} \sum_{i=1}^N l_i \cdot \mathcal{I}_{pixel_i}$$

donde \mathcal{I}_{pixel_i} es la intensidad del punto de la imagen del correspondiente segmento l_i . Las Figuras 2.18 y 2.16 muestra el vector de acumulación angular $\tilde{\mathcal{I}}_\theta$ obtenido para las imágenes que aparecen en las mismas figuras.

Radios

Para obtener el radio R_1 (radio externo de la plantilla) podemos utilizar la siguiente función definida en [Parida *et al.*, 1997]:

$$\mathcal{R} = \int_{R_0}^{R_1} \int_0^{2\pi} \left(\frac{\partial \mathcal{I}}{\partial r} \right) dr d\theta$$

Se calcula el valor de la función aplicada al punto candidato, incrementando el valor de R_1 . Un valor preestablecido limita el valor de \mathcal{R} y así se obtiene el valor R_1 . R_0 suele ser una fracción de R_1 . R_0 es importante debido a que cerca de un punto esquina se produce un cambio elevado en la dirección del gradiente y debido a la discretización se produce un elevado nivel de ruido. Normalmente se fija el valor de R_1 pues su cálculo resulta extremadamente costoso. Sin embargo,

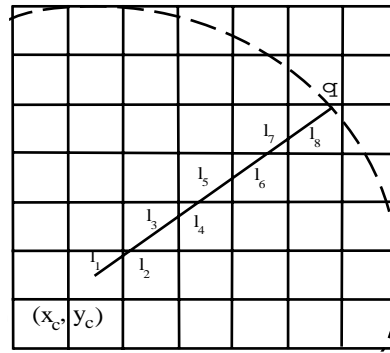


Figura 2.17: Ejemplo de cálculo de la discretización del vector angular.

estos valores pueden ser críticos, tal como se muestra en la Figura 2.19. En el caso de la imagen de la derecha, los valores utilizados fueron de $R_0 = 2, R_1 = 5$. En el caso de la izquierda, $R_0 = 0, R_1 = 10$. Como se puede observar, la imagen de la izquierda presenta peores resultados debido a que el dominio de la unión engloba otras regiones. Por ello aparecen límites angulares en posiciones donde teóricamente no deberían de aparecer. En nuestros experimentos hemos utilizado los valores de $R_0 = 2, R_1 = 5$.

Para localizar el centro de la unión se calcula el valor de \mathcal{R} en toda la imagen y aquel punto con un mínimo local de la función (que supere un cierto umbral prefijado) es un candidato a unión.

Cálculo de las secciones angulares

Una vez identificados el centro y los radios, los únicos parámetros que quedan por encontrar son el número de secciones angulares y los ángulos donde se encuentran los límites de las secciones angulares. Los pasos a seguir para clasificar las uniones dentro de una imagen son:

1. Calcular los valores de intensidad de las secciones angulares. Se realiza mediante una media de los valores dentro de la sección angular:

$$T_l = \frac{\sum_{j=k_l}^{K_{l+1}} \tilde{\mathcal{I}}(\theta_j)}{k_{l+1} - k_l}$$

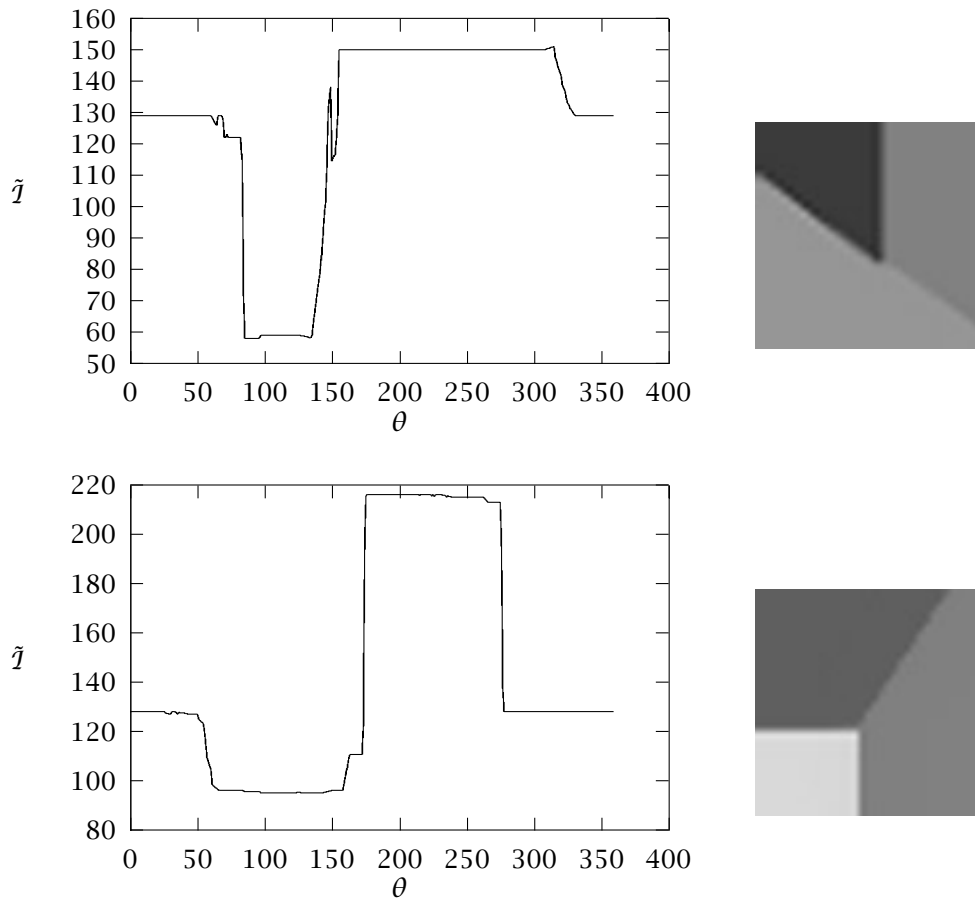


Figura 2.18: Valor acumulado de los radios en imágenes sintéticas.

2. Para calcular las posiciones de los límites de las secciones angulares se exploran todas las posibles situaciones de dichos límites. Para buscar las situaciones correctas se utiliza programación dinámica.
3. Se calcula el valor \bar{E} de la siguiente manera:

$$\bar{E}^p = \sum_{l=1}^p \sum_{j=k_l}^{k_{l+1}} (\tilde{I}(\theta_j) - T_l)^2$$

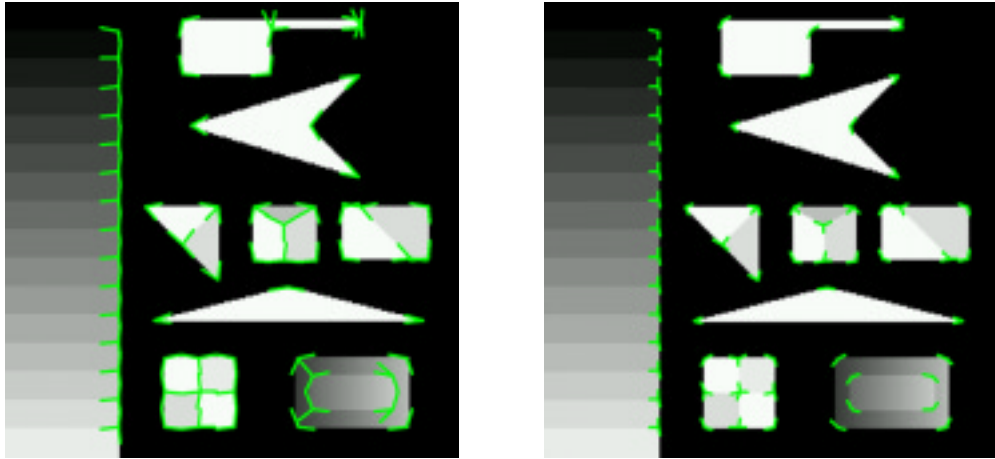


Figura 2.19: Efecto no deseado al utilizar un radio incorrecto.

que es una medida del error cometido con respecto a la imagen. Se procede a calcular desde $p = 1$ hasta que el cociente $r^n = \frac{\bar{E}^{n+1}}{\bar{E}^n}$ sea inferior a un cierto umbral prefijado. Aquel valor de n que hace que r^n descienda por debajo del umbral, nos determina el número de secciones angulares.

Las Figuras 2.20 y 2.21 nos muestran resultados de aplicar el método Kona. El tiempo medio de ejecución, utilizando el operador de SUSAN como inicialización, fue de 22 segundos para clasificar 200 uniones. En la siguiente sección se realiza un análisis más profundo de los resultados obtenidos con este operador.

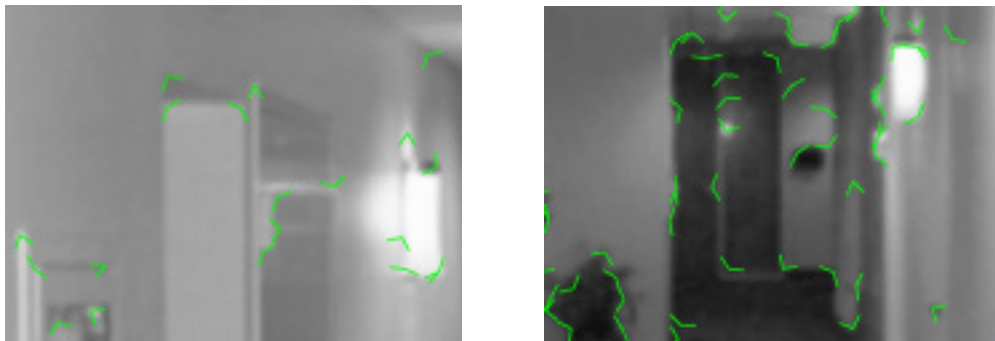


Figura 2.20: Resultado de aplicar el método Kona.

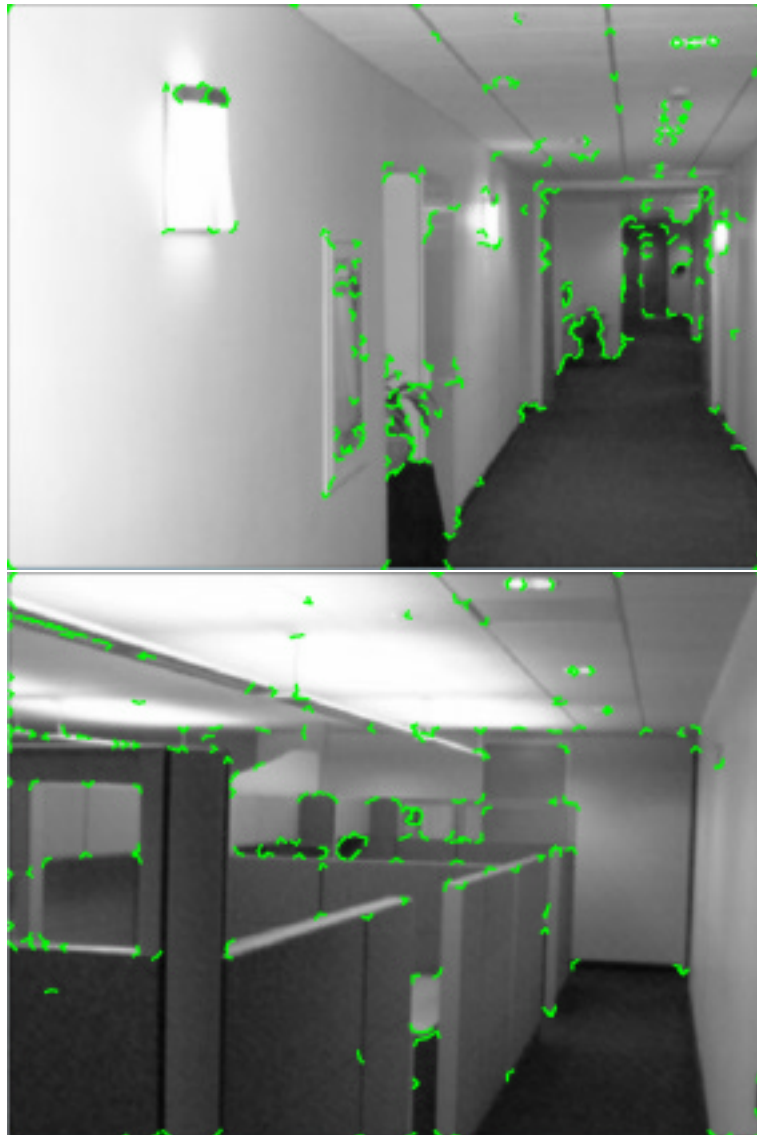


Figura 2.21: Resultado de aplicar el método Kona.

Discusión

Hemos realizado una experimentación previa con el operador Kona. Presenta una buena clasificación de las uniones con un error próximo a 15°. Sin embargo, la principal desventaja de este método es el alto coste temporal. Sustituyendo su detector del centro por el de SUSAN, mucho más rápido, el tiempo de ejecución por imagen fue de 22 segundos. Un tiempo inaceptable para el ámbito de aplicación de esta tesis, la robótica móvil. Por ello se hace necesario un nuevo método de clasificación de los límites angulares que reduzca lo más posible el tiempo de ejecución. En las siguientes secciones vamos a presentar dos nuevos métodos de clasificación de uniones. El primero consiste en aplicar el algoritmo *Region Competition* aplicado a una dimensión, de manera que la búsqueda de los límites de las secciones angulares se realice de forma voraz. En el segundo método explotaremos el modelado estocástico de aristas. Acabaremos este capítulo presentando resultados y comparativas entre ambos métodos.

2.4 Clasificación de uniones mediante segmentación

Este método sigue la misma filosofía que el algoritmo de *Region Competition* [Zhu y Yuille, 1996a], pero aplicado a segmentación en una sola dimensión. El método realiza una búsqueda de las secciones angulares que maximicen la homogeneidad de la intensidad angular. Una sección angular $S_i = \{\theta | \theta \in [\theta_{S_i}, \theta_{S_{i+1}}]\}$ se considera homogénea si sus valores de intensidad son consistentes con los generados por una familia de distribuciones de probabilidad preespecificadas $P(\hat{I}_\theta | \alpha)$ donde α son los parámetros que codifican la distribución.

Vamos a definir una función de energía global aplicando un criterio MDL¹:

$$\mathcal{E}(\Theta, \Gamma, \{\alpha_i\}) = \sum_{i=1}^M \left\{ -\log P(\{\hat{I}_\theta : \theta \in R_i\} | \alpha_i) \right\} \quad (2.2)$$

donde M es el número de secciones angulares, $P(\{\hat{I}_\theta : \theta \in S_i\} | \alpha_i)$ es la suma del coste de codificar cada valor \hat{I}_θ dentro de la sección angular S_i de acuerdo a la distribución $P(\hat{I}_\theta | \alpha_i)$ y $\Gamma = (\theta_{S_1}, \theta_{S_2}, \dots, \theta_{S_M})$ son los límites de las secciones angulares. Asumiendo modelos de probabilidad independiente, tenemos que:

$$\log P(\{\hat{I}_\theta : \theta \in S_i\} | \alpha_i) = \int_{S_i} \log P(\hat{I}_\theta | \alpha_i) d\theta \quad (2.3)$$

¹Minimum Description Length

y la función de energía global resultante es²:

$$\mathcal{E}(\Theta, \Gamma, \{\alpha_i\}) = \sum_{i=1}^M \left\{ - \int_{\theta_{S_i}}^{\theta_{S_{i+1}}} \log P(\hat{\mathcal{I}}_{\theta} | \alpha_i) d\theta \right\} \quad (2.4)$$

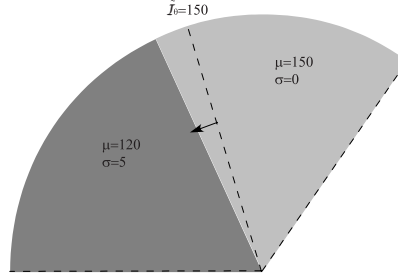


Figura 2.22: Dinámica del límite de una sección angular. En este caso, $P(\mathcal{I}_{\theta_s} | \mu = 150, \sigma = 0) > P(\mathcal{I}_{\theta_s} | \mu = 120, \sigma = 5)$, por lo que la posición del límite se desplaza hacia la izquierda.

Para encontrar el mínimo de esta función de energía se pueden utilizar algoritmos tales como el *simulated annealing* [Geman y Geman, 1984], pero inducen un alto coste computacional y nosotros buscamos un esquema que permita trabajar en tiempo real. Presentamos, pues, un algoritmo voraz dividido en dos fases:

1. Partimos de una segmentación en secciones previa. Para realizar esta segmentación calculamos las aristas utilizando el método de SUSAN adaptado a una dimensión, realizando una supresión de falsos mínimos. A continuación, tenemos que encontrar los α_i , $i = 1, 2, \dots, M$ que minimizan el coste de descripción de cada región.

La estimación de los α_i 's se puede realizar analíticamente y suponiendo que $P(\hat{\mathcal{I}}_{\theta} | \alpha_i)$ es una distribución Gaussiana, entonces $\alpha = (\mu, \sigma)$ es la media y la varianza.

2. En el segundo paso, calculados los $\{\alpha_i\}$, realizamos descenso por gradiente con respecto a los puntos $\theta \in \Gamma$. Para cualquier punto θ_{S_i} que pertenece a los límites de una sección angular tenemos:

$$\frac{d\theta_{S_i}}{dt} = - \frac{\partial \mathcal{E}(\Theta, \Gamma, \{\alpha_i\})}{\partial \theta_{S_i}} = \log \left\{ \frac{P(\hat{\mathcal{I}}_{\theta_{S_i}} | \alpha_{i-1})}{P(\hat{\mathcal{I}}_{\theta_{S_i}} | \alpha_i)} \right\} \quad (2.5)$$

²En lo sucesivo, si $i = M$ entonces $S_{i+1} = S_1$.

Esta función obtenida es equivalente a la regla de decisión de clasificación bayesiana de dos categorías [Duda y Hart, 1973] (ver Apéndice A). El movimiento de θ_{S_i} se determina por el ratio anterior. Si $P(\hat{\mathcal{I}}_{\theta_{S_i}} | \alpha_i) > P(\hat{\mathcal{I}}_{\theta_{S_i}} | \alpha_{i-1})$, es decir, la intensidad en el punto se ajusta mejor a la distribución de la sección angular S_i que a la de S_{i-1} , entonces el límite de la sección angular se desplazará disminuyendo su valor radial. Las Figuras 2.22 y 2.23 muestran de forma gráfica qué ocurre en este caso.

En este punto se nos presentan algunos problemas que debemos solventar:

1. El movimiento del contorno depende de la probabilidad de $\hat{\mathcal{I}}_{\theta}$. Pequeños cambios en la intensidad pueden provocar que los límites de las secciones angulares no se clasifiquen correctamente. Por ello utilizaremos una discretización de $\pi/180$ para que dicha discretización no afecte demasiado.
2. Debemos permitir la unión de regiones cuando éstas tengan parámetros similares. Incorporamos ahora una nueva fase que permita reducir el número de regiones si con ello se disminuye la energía. Para ello, dadas dos regiones adyacentes, fusionaremos las dos regiones en una si la suma de la energía de ambas regiones es mayor o igual que la energía de la región resultado de la fusión. En la práctica si el cociente entre la suma de las energías y la energía de la región fusionada está por debajo de un cierto umbral, realizamos la unión.

En nuestro problema, vamos a trabajar con distribuciones Gaussianas. Esto se debe a que suponemos que dentro de una sección angular los valores de intensidad se pueden modelar mediante una distribución de este tipo. Esto se puede ampliar a otros tipos de distribuciones, que permitan trabajar, por ejemplo, con texturas. Por lo tanto, nuestra distribución va a estar modelada por una media y una varianza, $\alpha = (\mu, \sigma)$ y

$$P(\hat{\mathcal{I}}_{\theta} | (\mu, \sigma)) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{(\hat{\mathcal{I}}_{\theta} - \mu)^2}{2\sigma^2} \right\} \quad (2.6)$$

Sustituyendo 2.6 en 2.5 obtenemos la siguiente expresión, que puede ser utilizada en vez del cálculo de logaritmos y exponenciales, sumamente costosos desde el punto de vista computacional:

$$\frac{d\theta_{S_i}}{dt} = \log\left(\frac{\sigma_i}{\sigma_{i-1}}\right) + \frac{(\hat{\mathcal{I}}_{\theta_{S_i}} - \mu_i)^2}{2\sigma_i^2} - \frac{(\hat{\mathcal{I}}_{\theta_{S_i}} - \mu_{i-1})^2}{2\sigma_{i-1}^2} \quad (2.7)$$

Vamos a introducir una serie de restricciones que deben cumplir las uniones para que sean consideradas como tal. Estas restricciones deberán ser tenidas en

cuenta a la hora de clasificar una unión y las tendremos en cuenta en los dos métodos propuestos. Las restricciones son tres:

No existencia de unión Si una unión se clasifica con menos de dos secciones angulares significa que no es tal unión. Podemos rechazar aquellas uniones que se clasifiquen con menos de dos límites angulares.

Esquina recta Cuando tengamos únicamente dos secciones angulares y la diferencia entre los ángulos de los límites de las secciones angulares sea igual a ± 180 , se trata de una recta, no de una unión. En determinadas aplicaciones nos puede interesar eliminar estas falsas uniones.

Secciones muy pequeñas Cuando una sección angular no supera un cierto umbral de anchura, podemos unirla con otra mayor. Nuestro umbral es de $\pi/9$.

Otro problema que debemos resolver es el siguiente: cuando acumulamos el valor de intensidad angular, se calcula una media de los píxeles a lo largo del radio. Esta media puede inducir a error debido a la posible presencia de otra región dentro del dominio de la unión. La Figura 2.25 muestra este problema. Para solventarlo aplicamos la mediana, tal como se muestra en la figura. Por lo tanto, en vez de utilizar \bar{I}_θ utilizaremos \hat{I}_θ , la mediana a lo largo del radio.

En la Figura 2.24 se presenta el algoritmo de segmentación lineal mediante competición de regiones y las Figuras 2.26 y 2.27 muestran el resultado de aplicar este método de clasificación de uniones.

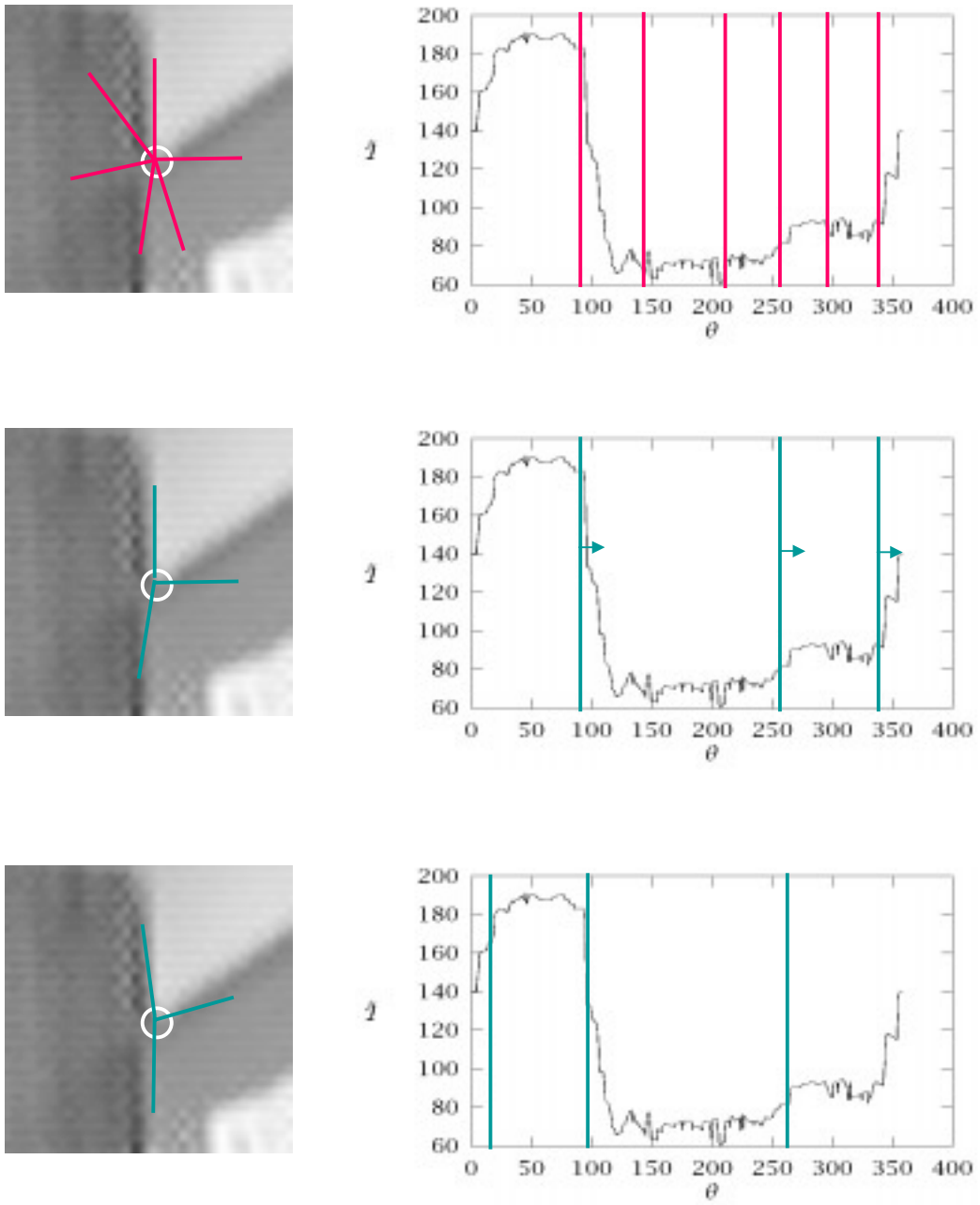


Figura 2.23: Dinámica del límite de una sección angular.

Algoritmo SEGLINEAL

Entrada: \hat{I} : imagen de entrada, R_1, R_0 : radios máximo y mínimo de la unión.

Salida: Uniones detectadas. Cada unión tendrá asociada los límites de las secciones angulares.

- (a) Se aplica el detector de esquinas seleccionado. El resultado serán los posibles candidatos a unión.
- (b) **Para cada** candidato a unión **hacer:**
 - i. Realizar la transformación del dominio circular a una dimensión.
 - ii. Inicializar la segmentación: calcular las aristas en una dimensión mediante SUSAN.
 - iii. Calcular los parámetros $\{\alpha_i\}$. En nuestro caso tenemos que encontrar la media y la varianza dentro de la sección angular.
 - iv. Aplicar la dinámica de Γ , minimizando la función de energía.
 - v. Ejecutar 2(b)iii, 2(b)iv iterativamente hasta que el movimiento de Γ converja. Entonces ir a 2(b)vi.
 - vi. Unir dos regiones adyacentes de tal manera que la unión provoque descenso de energía. Si se produce unión ir a 2(b)iii. Si el número de secciones es menor de dos, eliminar el candidato actual y pasar al siguiente.
 - vii. Si la unión no cumple las restricciones de no existencia de unión o esquina recta eliminar este candidato de la lista de uniones y pasar al siguiente candidato.
- (c) **FinPara**

Figura 2.24: Algoritmo de segmentación lineal mediante competición de regiones.

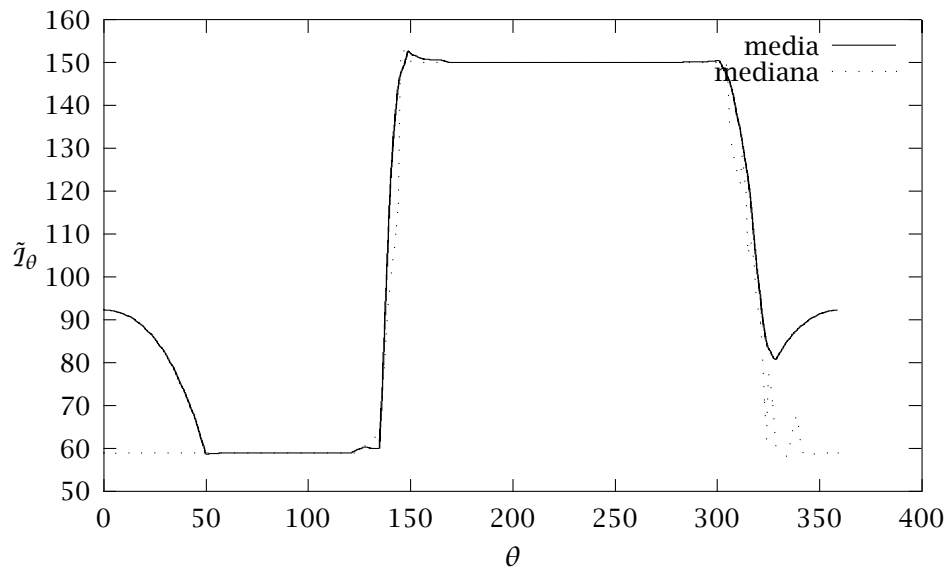


Figura 2.25: Error producido en la discretización.

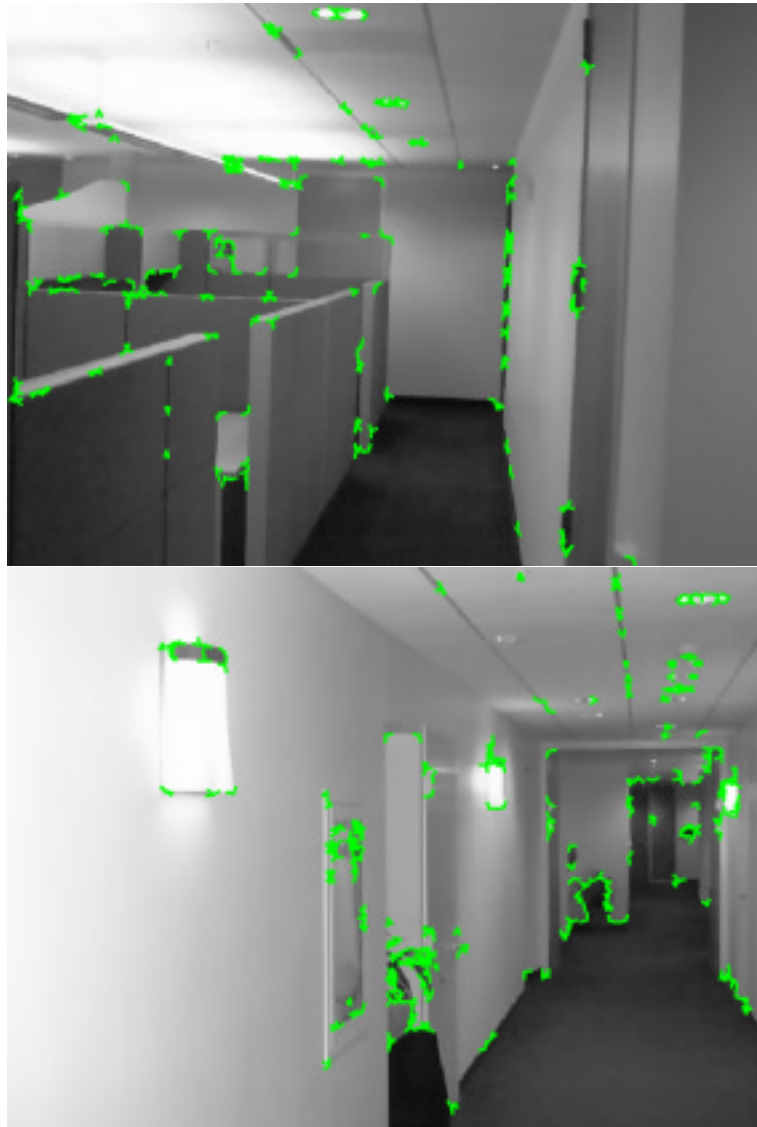


Figura 2.26: Resultado de aplicar el clasificador de uniones.



Figura 2.27: Resultado de aplicar el clasificador de uniones.

2.5 Clasificación de uniones mediante modelado de aristas

El método de segmentación lineal se basa en un criterio estadístico para medir la homogeneidad de regiones. Una alternativa, también bayesiana, a la homogeneidad de regiones consiste en medir la respuesta de los filtros de detección de aristas. Para ello nos basamos en recientes estudios [Konishi *et al.*, 1999] orientados a evaluar la calidad de dichos filtros.

Modelado de aristas

Supongamos un filtro local, como por ejemplo un filtro de gradiente. Si clasificamos previamente los píxeles de una imagen es posible extraer una caracterización estadística del filtro determinando la respuesta de éste cuando se encuentra en una arista real y cuando no lo está. Esta respuesta del filtro se suele cuantificar para que tome un número discreto de valores. En la Figura 2.29 se muestra las estadísticas obtenidas en el trabajo antes referenciado cuando el píxel se encontraba en una arista o fuera de ella, con una discretización de 20 valores. También, la Figura 2.30 muestra la función logaritmo del ratio entre estas dos funciones. Estas distribuciones nos proporcionan la probabilidad de que dada una respuesta, ésta corresponda a una arista o no. Denotaremos estas probabilidades como $P_{on}(E_{\vec{u}})$ y $P_{off}(E_{\vec{u}})$, respectivamente.

El filtro que se va a utilizar es muy simple: se trata de utilizar un filtro de gradiente seguido de lo derivada de una gaussiana con desviación estándar 1: $\vec{\nabla}G_{\sigma=1} * I$. Las imágenes resultantes una vez aplicado el filtro y calculados la magnitud y la orientación se muestra en la Figura 2.28. Este método se utilizará en éste y los siguientes capítulos como base para la formulación bayesiana propuesta.

Vamos a introducir el término de orientación de las aristas en las probabilidades. Tenemos $P_{on}(\vec{E}_{\vec{u}}|\beta)$ y $P_{off}(\vec{E}_{\vec{u}})$, donde $\vec{E}_{\vec{u}} = (E_{\vec{u}}, \phi_{\vec{u}})$, β es la orientación real de la arista y $\phi_{\vec{u}}$ es la dirección del gradiente medida en el punto \vec{u} . A continuación, realizan una suposición de factorización que $P_{on}(\vec{E}_{\vec{u}}|\beta) = P_{on}(E_{\vec{u}})P_{ang}(\phi_{\vec{u}} - \beta)$ y $P_{off}(\vec{E}_{\vec{u}}) = P_{off}(E_{\vec{u}})U(\phi_{\vec{u}})$. $P_{ang}(\cdot)$ codifica una medida de la distancia de la orientación de la arista $\sigma_{\vec{u}}$ a una orientación dada β . P_{ang} está basada en datos experimentales y tiene picos en 0 y π . La Figura 2.30 muestra el tipo de función utilizada. $U(\cdot) = 1/2\pi$ es una distribución uniforme.

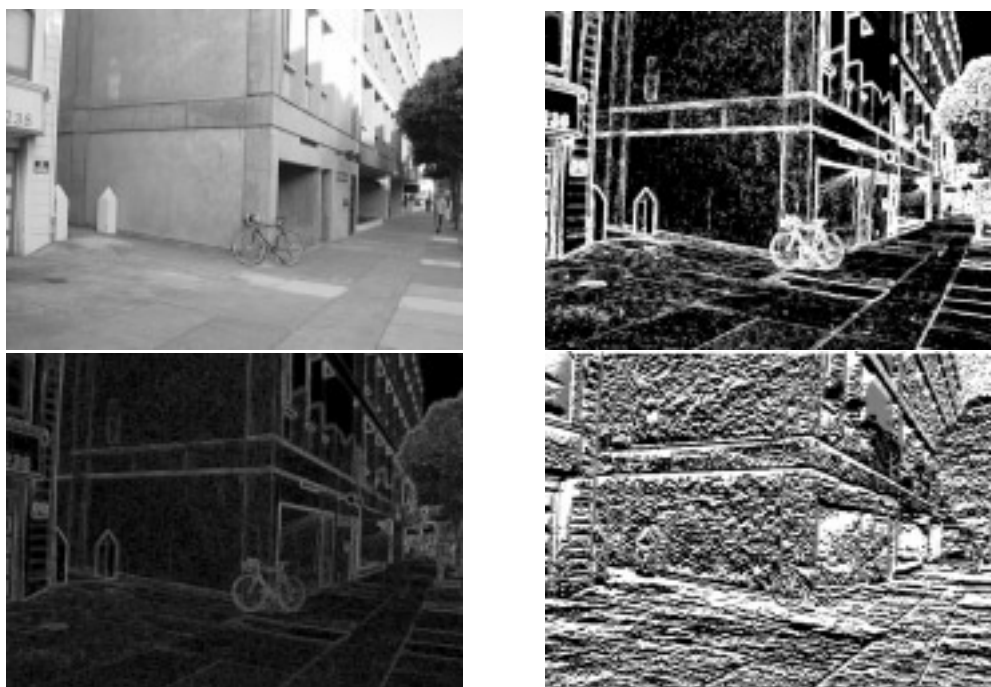


Figura 2.28: Arriba: Imagen original y el resultado de calcular el $\log(P_{on}/P_{off})$ en cada píxel. Abajo: Cálculo de los valores de orientación (izquierda) y magnitud (derecha).

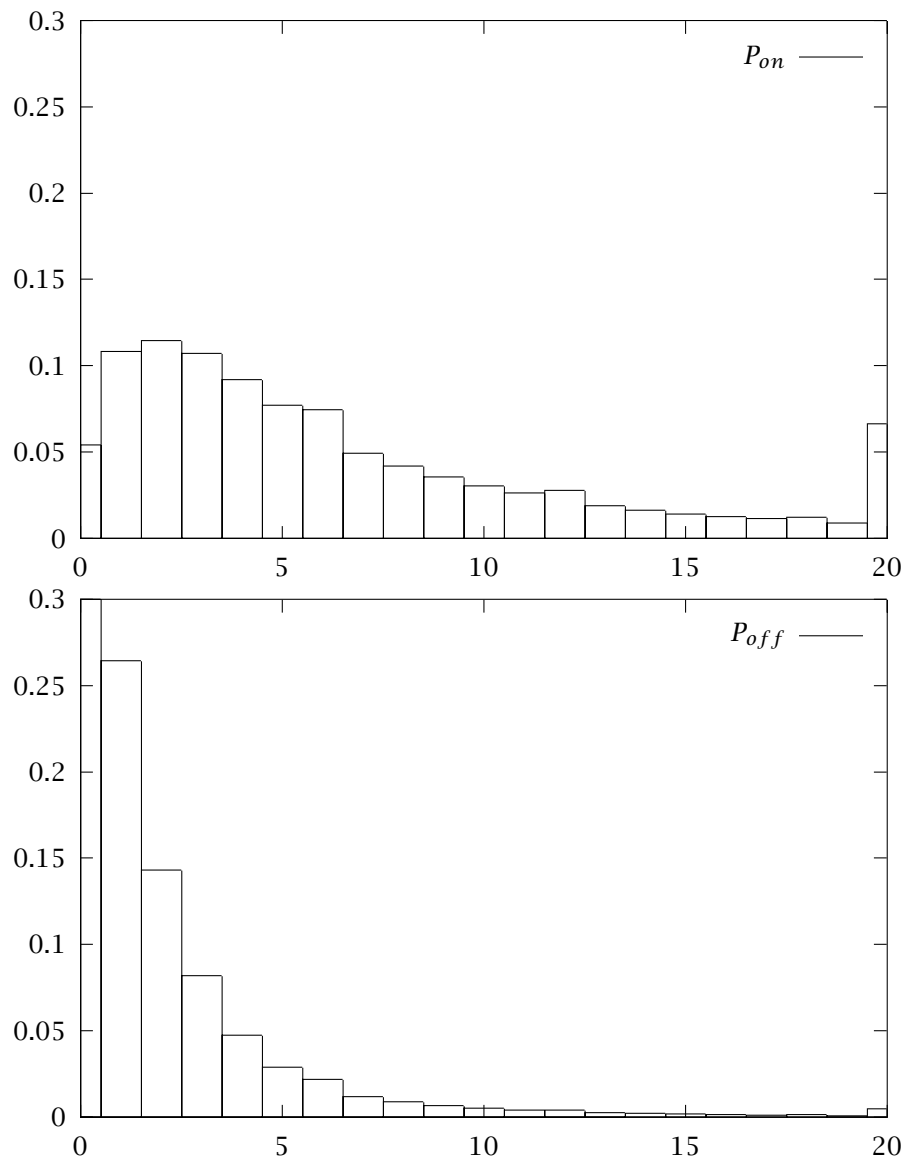


Figura 2.29: Estadísticas de las aristas: P_{on} arriba, P_{off} abajo.

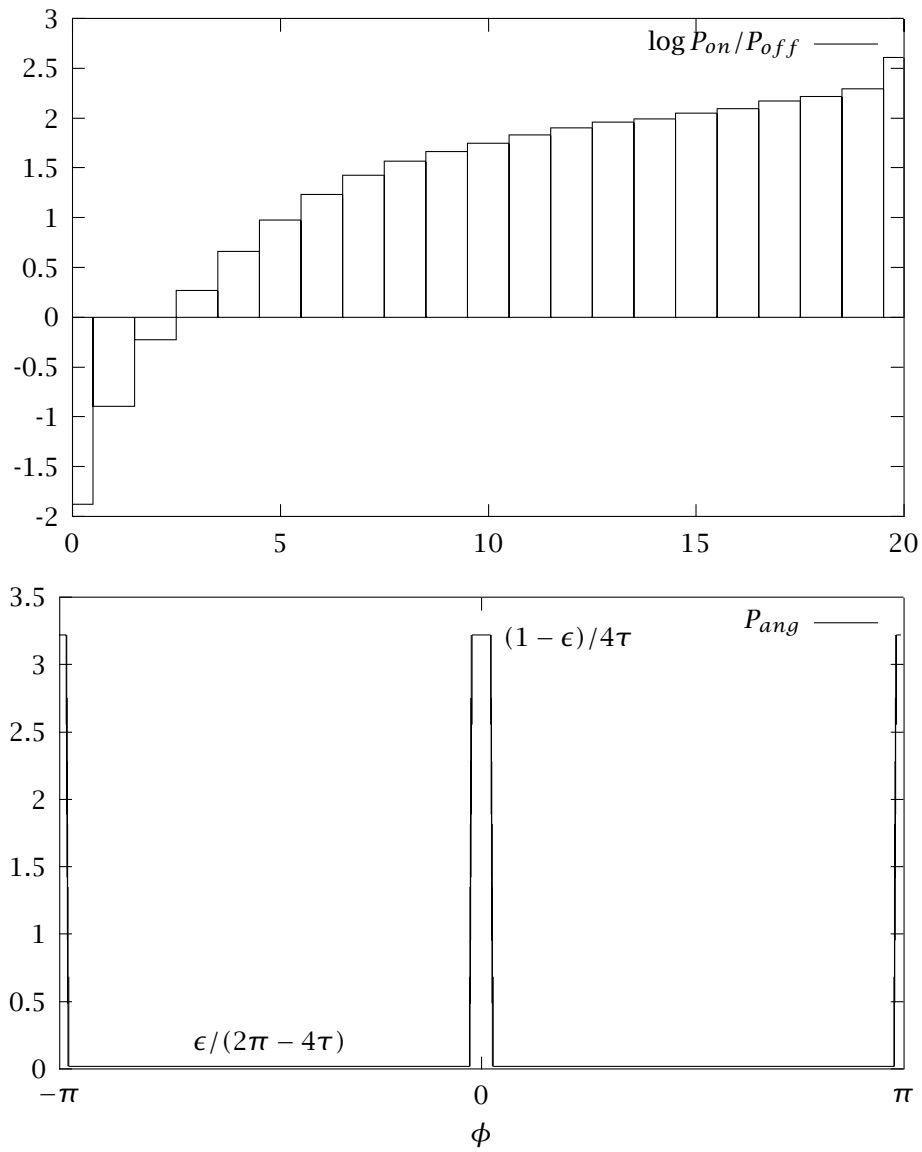


Figura 2.30: Gráfica del log de P_{on}/P_{off} (arriba) y de la función utilizada para P_{ang} (abajo).

Clasificación de uniones

Dado este modelo variamos la forma de obtener los vectores de acumulación, es decir, de realizar la transformación de dos dimensiones a una. Para ello haremos uso de la siguiente ecuación:

$$\hat{\mathbf{I}}_{\theta} = \frac{1}{(R_1 - R_0)} \sum_{i=1}^N l_i \cdot \log \frac{P_{on}(\vec{E}_{\vec{u}}|\beta)}{P_{off}(\vec{E}_{\vec{u}})} \quad (2.8)$$

donde $\vec{E}_{\vec{u}} = (E_{\vec{u}}, \sigma_{\vec{u}})$ siendo $E_{\vec{u}}$ la magnitud del gradiente y $\sigma_{\vec{u}}$ la orientación del mismo en el píxel \vec{u} , y siendo β el ángulo de la normal a θ .

Una vez hemos realizado la transformación a una dimensión, procedemos a marcar un umbral de corte, previamente definido, para detectar los picos de la función. Estos picos nos determinan los límites de las secciones angulares. Por último se realiza una búsqueda de máximo local para localizar las posiciones de dichos límites. El perfil obtenido después de la transformación se muestra en la Figura 2.31.

La Figura 2.32 muestra el algoritmo de detección de los límites angulares mediante la información de modelado de aristas. Las Figuras 2.33 y 2.34 muestran el resultado de aplicar este método de clasificación de uniones.

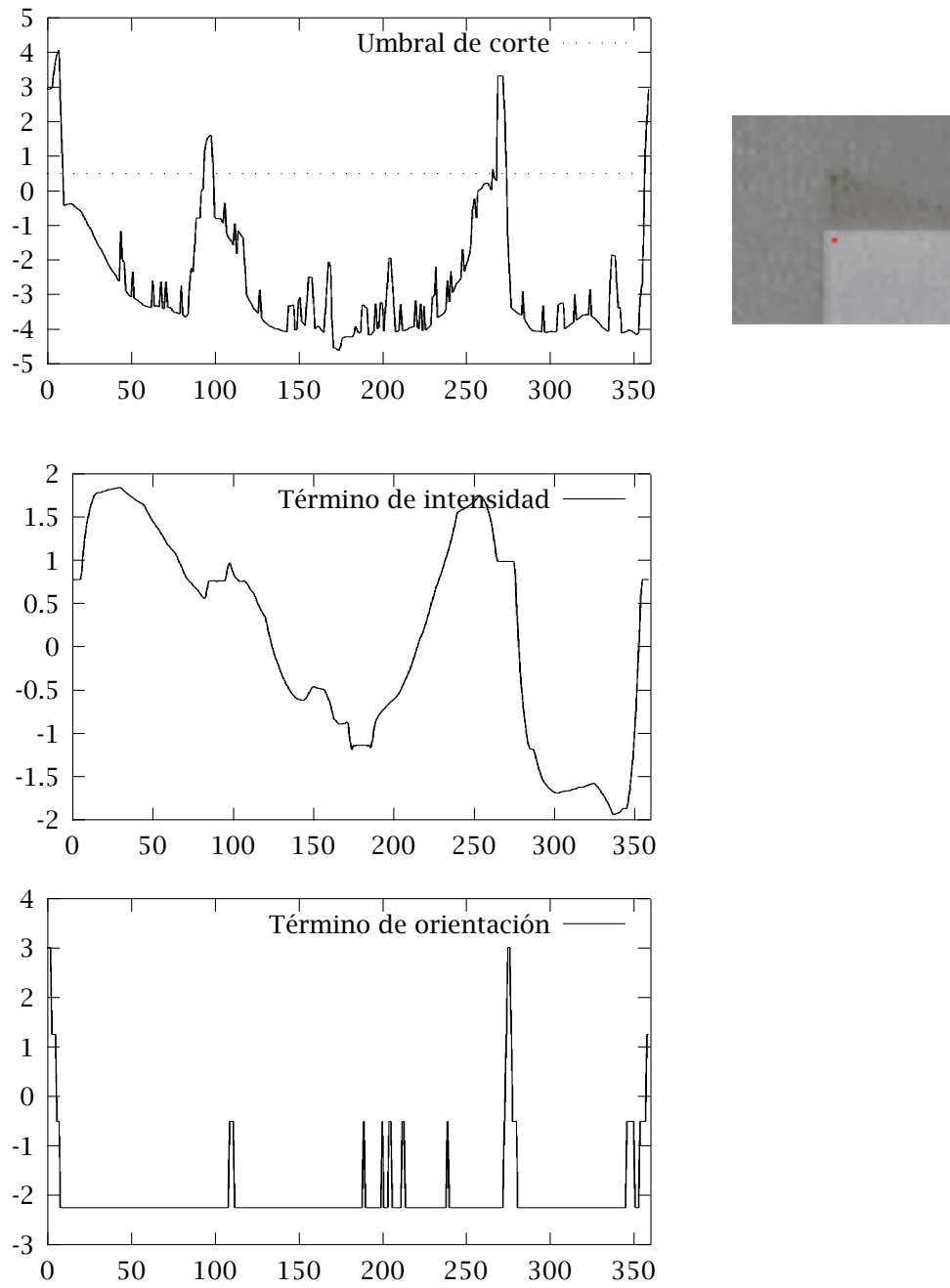


Figura 2.31: Transformación de la imagen con el método de P_{on}/P_{off} .

Algoritmo MODELARISTAS

Entrada: \hat{I} : imagen de entrada, R_1, R_0 : radios máximo y mínimo de la unión.

Salida: Uniones detectadas. Cada unión tendrá asociada los límites de las secciones angulares.

1. Se aplica el detector de uniones seleccionado. El resultado serán los posibles candidatos a ser unión.
2. Se aplica el filtro gaussiano a la imagen de entrada. El resultado serán dos imágenes: la magnitud del gradiente y su orientación.
3. **Para cada** candidato a unión **hacer:**
 - (a) Realizar la transformación del dominio circular a una dimensión. La transformación se realiza aplicando la Ecuación 2.8.
 - (b) Aplicar un umbral de corte. Todos aquellos valores inferiores al umbral pasan a tener valor cero.
 - (c) Buscar valores máximos locales. Estos máximos serán las posiciones de los límites angulares.
 - (d) Si la unión o algunos de sus límites no cumple las restricciones de no existencia de unión o esquina recta eliminar la unión o el límite correspondiente.
4. **FinPara**

Figura 2.32: Algoritmo de detección de uniones mediante modelado de aristas.

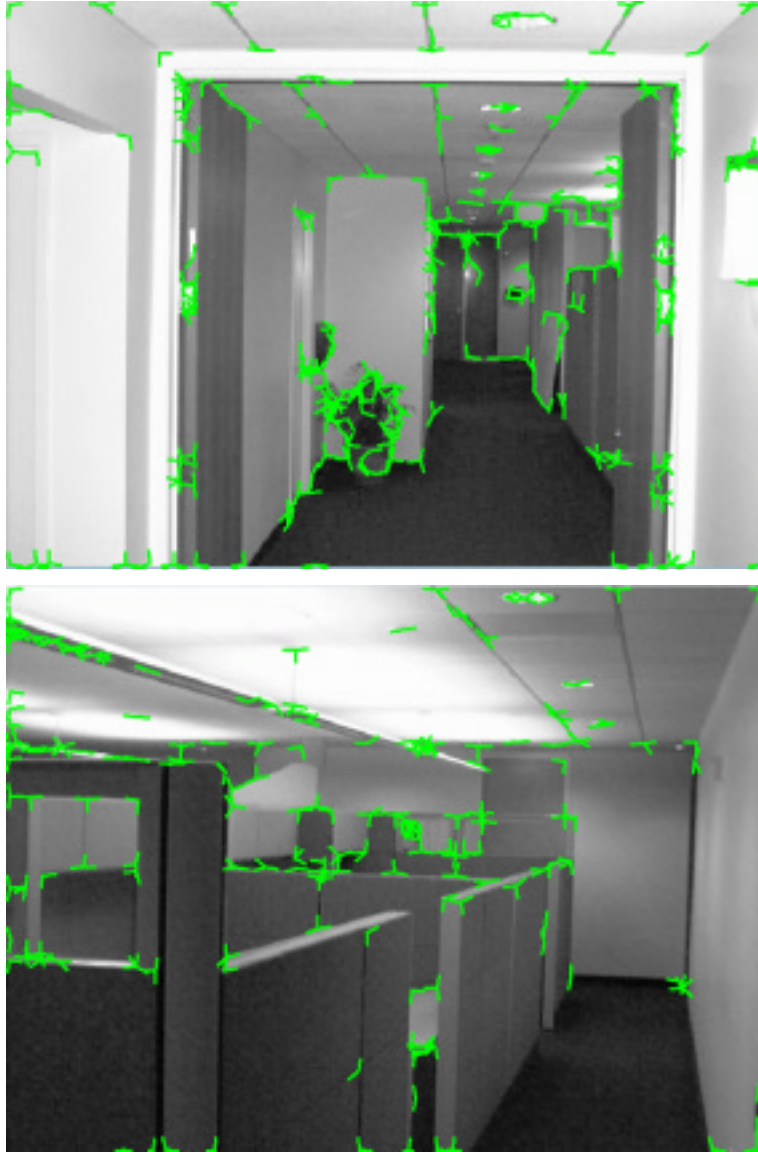


Figura 2.33: Resultado de aplicar el método de modelado de aristas.



Figura 2.34: Resultado de aplicar el método de modelado de aristas.

2.6 Discusión

Hemos realizado comparativas entre los tres métodos presentados: el de Kona, el algoritmo de segmentación lineal y el de modelado de aristas. Los tres métodos se probaron en una secuencia de interior, en un escenario por donde se desplazaría un robot. La secuencia constaba de 18 instantáneas. También se probó en imágenes de exterior. Todas las imágenes tenían tamaño 640x480. Los algoritmos se implementaron en C++, utilizando la librería de procesamiento de imágenes Vista. Los experimentos se realizaron en un Pentium II 266 MHz bajo Linux.

En cuanto al error angular cometido en la obtención de los límites angulares, se obtiene un error medio de 15° para Kona, frente a los 10° obtenidos por los dos métodos propuestos. El algoritmo de modelado de aristas presenta un menor tiempo de procesamiento. La media del tiempo de computación fue de 4 segundos. A continuación, tenemos el algoritmo de segmentación lineal con 7 segundos. Por último, el método de Kona presentó un tiempo de procesamiento medio de 22 segundos.

El método de modelado de aristas es más sensible a la longitud del radio debido a su formulación basada en características locales. Necesita un radio mayor para acumular evidencia, lo que provoca errores como los descritos en la Sección 2.3. Sin embargo, este método presenta un número menor de límites no detectados frente a los otros dos. También presenta un mayor número de falsos positivos, límites que en realidad no lo son.

El algoritmo de segmentación lineal responde más eficientemente a la detección de los límites. Sin embargo, es más sensible a ruido.

Los tres métodos son poco robustos, y en la misma medida, al error por la mala localización del centro. Esta situación es bastante lógica al formularse los tres en el mismo dominio y con la misma transformación.

Resumiendo, en este capítulo hemos presentado dos nuevos métodos para la clasificación de uniones. Estos métodos hacen uso de operadores de detección de puntos característicos, es decir, de puntos donde la curvatura de los contornos es alta. También son puntos característicos aquellos puntos donde convergen varias aristas. Hemos comparado dos de los métodos de detección de dichos puntos, el operador de SUSAN y el de Nitzberg. Hemos determinado en qué situaciones utilizaremos uno y otro.

Definimos qué vamos a considerar una unión, presentando sus parámetros, que serán los que tendremos que encontrar. Para ello definimos los dos nuevos métodos mediante un esquema de minimización de función de energía. Ambas formulaciones tienen su fundamento en un esquema bayesiano.

Hemos realizado experimentos para comprobar la eficacia de los métodos propuestos, presentando resultados obtenidos al aplicarlos sobre imágenes tanto de

interior como de exterior. Los dos métodos presentan buenos resultados en imágenes de interior. Sin embargo, cuando procesamos imágenes de exterior, y debido a la aparición de objetos no estructurados (árboles, carteles, texturas, etc.) el número de uniones detectadas es muy elevado, produciéndose esto por la cantidad de puntos candidatos detectados por los operadores. También hemos observado que los métodos propuestos son muy dependientes de la localización del centro de la unión. Malas localizaciones de dicho centro provocan errores en la clasificación de los límites angulares.

Hemos analizado la eficiencia de estos métodos propuestos frente a Kona, obteniéndose un error y un tiempo de procesamiento menor cuando aplicamos nuestros operadores. El algoritmo de segmentación lineal detecta un menor número de falsos positivos. Sin embargo, el algoritmo de modelado de aristas tiene un tiempo de computación menor y deja un menor número de límites sin detectar.

En algunas situaciones se detectan límites inexistentes. Es el caso de cuando tenemos una región no perteneciente a la unión que se encuentra en el dominio de ésta. Este caso no se puede detectar mediante los métodos propuestos, y una clara ampliación es la prolongación de la unión para buscar otra unión en la dirección que marca el límite de la sección angular. Esta propuesta es la que desarrollaremos en el siguiente capítulo.

