

UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

DEPARTAMENTO DE ELECTRÓNICA



TESIS DOCTORAL

**“Sistema de Localización y Mapeado Simultáneo
Basado en Visión Estereoscópica Para la Asistencia a
la Navegación de Vehículos en Grandes Entornos”**

David Schleicher Gómez

2009

UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

DEPARTAMENTO DE ELECTRÓNICA



**“Sistema de Localización y Mapeado Simultáneo Basado en
Visión Estereoscópica Para la Asistencia a la Navegación de
Vehículos en Grandes Entornos”**

Autor

David Schleicher Gómez

Directores

Luis Miguel Bergasa Pascual
Manuel Ocaña Miguel

2009

TESIS DOCTORAL

A mi familia, cuya desinteresada
ayuda ha sido clave en la consecución
del reto de terminar esta Tesis

Agradecimientos

Llegados a este punto, es importante reflexionar mirando atrás en el tiempo y recordar las diversas circunstancias personales y profesionales que ha habido que superar en estos últimos cuatro años de entrega e ilusión para llevar a buen puerto esta Tesis. Sin embargo, no es menos importante darse cuenta de la gran suerte de estar rodeado de tantas personas que sin su ayuda, directa o indirecta, este hecho no hubiera sido posible y a las que tanto tengo que agradecer.

En primer lugar, agradecer a mi tutor, Luis Miguel Bergasa, la dedicación plena demostrada en este objetivo, siendo un gran apoyo técnico y sobre todo, sabiendo “entrenar al corredor” de una forma tan magistral. Siempre recuerdo ese primer contacto, preguntando por los contenidos de los cursos de doctorado y cómo desde el primer momento demostró una sorprendente confianza en mi. Muchas gracias, Luismi, por permitirme cumplir mi sueño de ser un poco más “docto”.

A Manuel Ocaña, mi co-tutor, compañero y sobre todo amigo, por su apoyo técnico y práctico en esta etapa, además de animarme en su día a iniciar este camino, en el que finalmente hemos coincidido. También quiero agradecer a Pedro Revenga el apoyo técnico que en diversas ocasiones me ha prestado sobre todo en problemas relacionados con el software. A Rafael Barea y Elena López por echarme una mano con algún congreso al que no pude asistir y en general a todo el Departamento de Electrónica por el gran trabajo de investigación y docencia que se lleva a cabo.

No quiero olvidarme de agradecer a Andrew Davison, investigador del Departamento de Computación del Imperial College de Londres, por su contribución desinteresada en los inicios de esta Tesis, así como su inestimable asesoramiento en esta etapa final. Asimismo, quiero agradecer a Udo Frese, profesor de la Universidad de Bremen e investigador del Transregional Collaborative Research Center, por su aportación en esta Tesis.

A mis padres Mari Carmen y Kurt, por apoyarme siempre, tanto en buenos momentos de esta etapa como en los malos, además de ayudarme en todas las tareas posibles para que pudiera concentrarme en llevar a cabo esta Tesis. Nunca les estaré suficientemente agradecido. A mis amigos, por estar ahí, aún cuando yo no pudiera estar siempre que quisiera con ellos. Por último, a mi novia Ana por su apoyo, a veces difícil, en estos cuatro años.

Resumen

En esta Tesis se presenta un nuevo sistema jerárquico de localización y mapeado simultáneos (SLAM: *Simultaneous Localization and Mapping*) en tiempo real, aplicado a la localización robusta de un vehículo en grandes entornos exteriores de tipo fundamentalmente urbano, siendo capaz de mejorar los sistemas de asistencia a la navegación actuales, basados únicamente en GPS. Además, esta propuesta podría utilizarse en sistemas de navegación autónoma con trayectorias recurrentes, como por ejemplo, líneas regulares de autobús, recorridos internos en parques temáticos, etc. El sistema se basa exclusivamente en la información proporcionada tanto por una cámara estéreo de gran angular y bajo coste, como por un sensor GPS también de bajo coste. El uso de una cámara estéreo demuestra numerosas ventajas frente a la implementación monocular, como por ejemplo la posibilidad de estimar la escala del mapa en verdadera magnitud.

El método propuesto se basa en dividir el mapa global en diversos sub-mapas locales, identificados por las denominadas huellas, que definen las poses del vehículo en ciertos puntos y que se relacionan con los sistemas de referencia de cada uno de los sub-mapas. En este nivel de sub-mapa, denominado *SLAM de bajo nivel*, se implementa un método de tipo métrico consistente en el mapeado secuencial de marcas visuales naturales en 3D, así como en la obtención de la pose del vehículo, a través de un filtro de Kalman Extendido (EKF) que modela el comportamiento dinámico del sistema. De esta forma se logran sub-mapas localmente consistentes. Para la elección del tipo idóneo de marcas en cuanto a propiedades de seguimiento, identificación y tiempos de cómputo, se ha realizado un estudio comparativo entre el método SIFT (*Scale-Invariant Feature Transform*) y de correlación directa, obteniéndose mejores resultados para este último. Además, en este nivel se integran las medidas obtenidas del GPS, mejorando la estimación de la posición del vehículo y, por tanto, la propia estimación del mapa local.

Sobre el nivel bajo de SLAM, descrito anteriormente, se añade un nivel jerárquicamente superior de tipo topológico denominado *nivel alto de SLAM*, el cual está basado en uso de las denominadas *huellas* y la implementación del método de optimización y corrección del mapa topológico, denominado MLR (*MultiLevel Relaxation*). Este algoritmo tiene como propósito reducir el error global del mapa, manteniendo las restricciones de tiempo real y proporcionando una estimación consistente del mismo, incluso en situaciones de pérdida de cobertura GPS. Para la detección de situaciones de cierre de lazos, fundamentalmente en ausencia de señal GPS, se ha implementado un tipo especial de huellas denominado *huella SIFT*. Éstas son capaces de identificar lugares previamente visitados, en base a su apariencia visual, mediante el empleo de características de tipo SIFT.

Por último, se presentan resultados de validación del método SLAM propuesto a partir de numerosos ensayos realizados fundamentalmente en grandes entornos exteriores de tipo urbano con un vehículo real. Las conclusiones obtenidas de los mismos muestran unos resultados de localización suficientemente precisos para los propósitos de esta Tesis, manteniendo los tiempos de cómputo dentro de las restricciones de tiempo real establecidas y empleando un hardware de

coste reducido. Como aplicación práctica final se propone el empleo del método para la mejora de sistemas de asistencia a la navegación, habiéndose realizado diversos ensayos en entornos reales. Se demuestra su uso en situaciones decisivas para la planificación de rutas, con mala o nula recepción de señal de GPS, mejorando los sistemas de navegación actuales.

Abstract

In this thesis, a new real-time hierarchical Simultaneous Localization and Mapping (SLAM) system is presented. It is applied to the robust localization of a vehicle in large-scale outdoor environments, essentially urban like, improving the current vehicle navigation systems based only on GPS. Then, it can be used on autonomous vehicle guidance with recurrent trajectories, like bus journeys, theme parks internal journeys, etc. The system is exclusively based on the information provided by both a low-cost wide-angle stereo camera and a low-cost GPS. The use of a stereo camera shows several advantages respect to the monocular implementation as, for example, the ability of map real scale estimation.

The approach divides the whole map into local sub-maps identified by the so-called *fingerprints*, which define vehicle poses at certain places. They are associated to each sub-map reference frame. In this sub-map level, called *low level SLAM*, a metric approach is carried out. There, a 3D sequential mapping of visual natural landmarks is carried out and the vehicle pose is obtained, using an Extended Kalman Filter (EKF) to model the dynamic behavior. Therefore, each sub-map is locally consistent. To choose the most convenient landmark features implementation, regarding tracking properties, distinctiveness and processing times, the SIFT (*Scale-Invariant Feature Transform*) and direct correlation approaches are analyzed. The last one has shown better results on the study. GPS measurements are integrated within this low level as well, improving vehicle positioning and, therefore, local map estimation.

A higher topological level, called *high level SLAM* is added over the low level SLAM. It is based on the so-called *fingerprints* and the implementation of an optimization and correction method for the topological map, called *MultiLevel Relaxation* (MLR). Therefore, this algorithm is in charged of reducing the global error within the map, keeping real-time constraints. This level also provides consistent estimation, even with long-term GPS unavailability. In order to detect loop closing situations, mainly when GPS signal is not available, a special kind of fingerprints, called *SIFT fingerprints* are implemented, which provide previously visited places identification, based on their visual appearance. SIFT features are used for this purpose.

Finally, experimental results, obtained from a high amount of tests, mainly in large scale outdoor urban environments using a real vehicle, are presented. As a conclusion from these tests, a highly accurate results, well suited for the main purposes of this Thesis, are showed, keeping processing times below the real time constrain, while using a low-cost hardware. As a practical final application, the use of the proposed method for navigation assistance systems improvement, is proposed. Several tests on real environments have been carried out. The use of the system is shown in key situations for path planning tasks, when poor GPS signal is received or it is completely unavailable, improving the current navigation systems.

Índice general

1. Introducción	1
1.1. Introducción	3
1.2. Problemática asociada a la localización y mapeado simultaneos de un vehículo en grandes entornos	5
1.2.1. Derivados del sensor utilizado	5
1.2.2. Derivados del entorno	5
1.2.3. Derivados de los requerimientos del sistema	6
1.3. Planteamiento y objetivos iniciales	7
1.4. Estructura de la Tesis	8
2. Estado del arte y objetivos de la Tesis	9
2.1. Estado del arte	11
2.1.1. Sistemas SLAM para pequeños entornos	11
2.1.2. Sistemas SLAM para grandes entornos	22
2.1.3. Sistemas basados en sensor GPS	39
2.2. Discusión y enmarque de la Tesis	43
2.3. Objetivos específicos de la Tesis	45
3. Visión general del sistema	47
3.1. Introducción	49
3.2. Elección del método	49
3.3. Arquitectura general	49
3.4. Conclusiones y aportaciones	51
4. SLAM de bajo nivel	53
4.1. Introducción	55
4.2. Vector de estado	55
4.3. Introducción al Filtro de Kalman Extendido (EKF)	58
4.4. Modelo de Predicción o Movimiento.	59
4.5. Modelado de las cámaras	61
4.5.1. Resolución de las cámaras	62
4.5.2. Separación entre cámaras	64
4.5.3. Apertura de las lentes	65
4.6. Calibración de las cámaras	66
4.6.1. Modelo de distorsión	69
4.7. Corrección de la distorsión	71
4.8. Modelo de medida	75
4.8.1. Selección de marcas	75
4.8.2. Cálculo del área de búsqueda	77
4.8.3. Determinación de la medida	80

4.8.4.	Transformación del patch	87
4.8.5.	Actualización	90
4.9.	Inicialización de nuevas marcas	92
4.9.1.	Método de detección de características	94
4.9.2.	Obtención del vector de estado de la marca	97
4.9.3.	Obtención de la covarianza del vector de estado de la marca	102
4.9.4.	Adaptación del vector de estado completo y su covarianza	107
4.9.5.	Eliminación de marcas	107
4.10.	Conclusiones y aportaciones	108
5.	SLAM de alto nivel	111
5.1.	Introducción	113
5.2.	Implementación	114
5.3.	Sub-mapas locales	116
5.4.	Huellas SIFT	120
5.5.	Detección de cierre de lazo	120
5.6.	Corrección del mapa	123
5.6.1.	Método lineal	123
5.6.2.	Método MLR	126
5.6.3.	Comparativa	129
5.7.	Fusión con GPS	134
5.7.1.	Estimación de la incertidumbre del GPS	135
5.7.2.	Fusión en el SLAM de bajo nivel	137
5.7.3.	Fusión en el SLAM de alto nivel	140
5.8.	Conclusiones y aportaciones	141
6.	Implementación práctica y resultados experimentales	145
6.1.	Introducción	147
6.2.	Descripción del HW utilizado	147
6.3.	Resultados del sistema SLAM	150
6.3.1.	Resultados preliminares en entornos interiores	150
6.3.2.	Resultados en grandes entornos exteriores	155
6.4.	Aplicación a la mejora de un navegador GPS	180
6.4.1.	Resultados de las pruebas	181
6.5.	Conclusiones y aportaciones	182
7.	Conclusiones y trabajos futuros	187
7.1.	Conclusiones y trabajos futuros	189
7.2.	Conclusiones generales y aportaciones	189
7.3.	Trabajos futuros	190
8.	Bibliografía	193

Índice de figuras

1.1. <i>Imagen superior</i> : Vehículos autónomos participantes en la carrera DARPA Urban Challenge. <i>Imagen inferior</i> : Hardware de control y sensores de distancia láser empleados en uno de los vehículos participantes.	4
1.2. Situaciones de entorno variable.	6
2.1. Robot equipado con cámara estéreo de visión activa.	12
2.2. Cámara utilizada para los experimentos.	13
2.3. Resultados de SLAM monocular.	13
2.4. Ejemplo de imagen obtenida con la cámara con dos marcas tipo línea y una tipo punto.	14
2.5. Estimación del recorrido del robot junto con las marcas tipo línea (en azul) y las marcas tipo punto (en rojo).	15
2.6. Resultados de SLAM monocular.	15
2.7. En la imagen superior se muestra la estimación de un recorrido realizado por el robot, con una longitud de 30 m y constituido por un mapa de 37 segmentos. En la imagen inferior se muestra un sencillo experimento para comparar la precisión en el retorno del robot al punto de partida, sin usar la información de bordes (izquierda) y usando esta información (derecha)	16
2.8. En la figura de la izquierda se muestra un robot empleado en las pruebas en un entorno con rocas. En el centro se muestra el recorrido realizado por el robot y las lecturas del sensor de distancia. En la figura de la derecha se muestra la estimación usando FastSLAM (línea punteada) y las localizaciones de las rocas, determinadas manualmente.	17
2.9. Precisión del algoritmo FastSLAM en función del número de marcas N (a) y del número de partículas M (b).	17
2.10. Robot en el entorno de pruebas (a). Estimación del recorrido usando SVFastSLAM y sólo odometría, comparado con la referencia obtenida con un DGPS (b).	18
2.11. Estimación del recorrido usando el método presentado (línea rayada) comparada con el recorrido real (línea continua).	19
2.12. Ejemplo de reconocimiento de objetos tridimensionales con oclusiones.	20
2.13. SIFT aplicado a la navegación de un robot.	20
2.14. Agrupaciones de marcas obtenidas y mantenidas en una secuencia en exteriores.	21
2.15. Resultado de las medidas de sonar para marcas tipo línea (a) y tipo punto (b). Representación del robot en dos sub-mapas diferentes (c) y (d), compuestos por marcas tipo línea Sx y tipo punto Px	23
2.16. Red Bayesiana representando dos sub-mapas independientes y sus dependencias probabilísticas. Las variables u_i representan las acciones introducidas al sistema, x_i las diferentes poses del robot, f_i las marcas de cada sub-mapa y z_i las observaciones realizadas	23

2.17. Representación del mapa global antes y después de una situación de cierre de lazo. Cada submapa se muestra de un color diferente. Las marcas obtenidas se muestran como puntos.	24
2.18. <i>Imágenes superiores</i> : Imágenes capturadas por la cámara junto a las proyecciones de las marcas y sus correspondientes covarianzas. <i>Imagen inferior</i> : Reconstrucción de la trayectoria del móvil junto con las marcas obtenidas en el entorno de prueba.	25
2.19. Representación del mapa y el recorrido realizado por el robot.	26
2.20. Robot utilizado para probar el método HLM, equipado con un <i>array</i> de sensores sonar en la parte superior	26
2.21. Vehículo de pruebas y mapa estimado por el sistema. Las poses del vehículo se representan en amarillo. Las medidas del sonar aparecen como puntos rojos, siendo la estimación final de las marcas las señaladas como círculos rodeados por su incertidumbre asociada.	27
2.22. En la imagen superior se muestra el vehículo utilizado para las pruebas. En la imagen inferior se muestra la reconstrucción del recorrido realizado por el vehículo, así como las medidas realizadas con el sensor láser	28
2.23. Representación del mapa global dividido en diversos sub-mapas. Se muestran también las covarianzas de incertidumbre de los sub-mapas (nodos) relativas al sub-mapa número 1, indicado en la parte superior izquierda.	28
2.24. Representación geométrica del algoritmo.	30
2.25. Representación del mapa y del proceso de medida.	31
2.26. Sistema estéreo empleado en las pruebas del sistema, constituido por dos cámaras web y un sensor inercial de bajo coste.	32
2.27. Vista superior del mapa en un entorno interior. La imagen superior muestra la estimación del mapa basado en las medidas visuales. La imagen inferior muestra esta estimación después de la optimización basada en mínimos cuadrados. Las etiquetas A - F representan marcas de control.	32
2.28. A la izquierda se muestra el robot móvil utilizado para los ensayos. A la derecha, en color rojo, se muestra el recorrido estimado por el sistema y, en color azul, la referencia obtenida con un DGPS (cuando está disponible)	33
2.29. Robot utilizado y mapa del entorno de pruebas.	34
2.30. Resultados de las pruebas de <i>matching</i> basado en apariencia. El vehículo circula en el sentido de las agujas del reloj formando un lazo con una longitud de 1.6 Km, empezando por la parte superior izquierda y terminando en la parte inferior derecha. El cierre de lazo comprende dos de los lados del cuadrado. Las posiciones, obtenidas con un GPS, en las que el vehículo obtiene una imagen se marcan con un punto amarillo. La obtención de estas posiciones no forma parte del sistema y sólo se muestran como ayuda en la representación gráfica. Los pares de imágenes cuya probabilidad de provenir del mismo lugar es mayor de 0.999, se marcan en rojo y se unen con una línea verde.	35
2.31. Representación de la geometría cuadrifocal junto con resultados de estimación obtenidos.	36
2.32. Hardware utilizado y resultados de estimación.	37
2.33. Resultado de la estimación de trayectoria y tiempos de cómputo.	38
2.34. Descripción del sistema y estimación del recorrido.	38
2.35. Proceso de <i>bundle adjustment</i> local cuando se añade un nuevo <i>keyframe</i> C_i . Sólo se optimizan las últimas poses y marcas asociadas. Sin embargo, se tienen en cuenta las <i>reproyecciones</i> 2D de las últimas N imágenes.	40
2.36. Resultados de estimación del sistema.	40
2.37. Robot móvil utilizado para las pruebas	41

2.38. Vehículo empleado y estimación del recorrido.	42
2.39. robot empleado y estimación del recorrido.	43
2.40. Sistema Inertial+ basado en GPS, sensores inerciales y odometría.	44
3.1. Arquitectura básica del sistema. En la parte inferior se muestran los diferentes sub-mapas, cada uno con su estimación independiente de un tramo del recorrido total, así como las marcas visuales. Todo ello constituye el SLAM de bajo nivel. En la parte superior se muestra el mapa topológico, en el que cada nodo está asociado a un sub-mapa. Estos nodos se muestran numerados del 1 al 8, así como las relaciones entre ellos. El nodo 3 tiene asociada una huella SIFT, como se indica en la imagen. Todo ello constituye el SLAM de alto nivel.	50
3.2. Tareas fundamentales del sistema según su ubicación en el nivel alto de SLAM o en el nivel bajo de SLAM.	51
3.3. Elementos principales del sistema presentado. Se destacan el vehículo, la cámara estéreo, el receptor GPS de bajo coste y la unidad de procesamiento.	52
4.1. Proceso de captura de marcas	56
4.2. Representación del vector de rotación	57
4.3. Modelo cinemático de un vehículo con cuatro ruedas, dos de ellas direccionales	60
4.4. Representación de la geometría epipolar 3D y nomenclatura utilizada	63
4.5. Tiempos de cómputo medios y sus varianzas para diferentes resoluciones de imagen. La línea horizontal marca el límite de tiempo real	63
4.6. Efecto de la incertidumbre en la determinación de la posición real de la marca P a consecuencia del error de $+1$ píxel en las coordenadas de proyección. El efecto del error de 1 píxel en una marca mas lejana P' provoca mayor error de profundidad que en una marca más cercana P''	64
4.7. Error en la determinación de la profundidad en función de la separación entre cámaras T_x y la distancia de la marca	65
4.8. Zonas de visibilidad perspectiva y monocular en función de la separación entre cámaras. El origen de coordenadas, en este estudio, se encuentra situado en el punto medio entre cámaras.	66
4.9. Distancia mínima de captura de marcas en función del ángulo de apertura. La marca $P2$ no es capturada, debido a que su proyección cae fuera del campo visual de la cámara izquierda.	67
4.10. Número de marcas capturadas por el sistema a lo largo del recorrido. Para la prueba se utilizó un recorrido de prueba consistente en una rotación de 360° de la cámara sobre su propio eje.	67
4.11. Proceso de selección de puntos de calibrado a partir de un patrón damero (cámara izquierda)	68
4.12. Imágenes de calibración de ambas cámaras	68
4.13. Visualización de los resultados de calibración y las diferentes muestras del patrón empleadas en el proceso	69
4.14. Efecto de la distorsión radial y tangencial sobre la proyección de un punto en el plano de imagen.	70
4.15. Errores de reproyección (en coordenadas de píxel) para el caso de usar cuatro coeficientes y para el caso de usar un único coeficiente. Cada color indica una imagen del patrón diferente	71
4.16. Diagrama general para los procesos de inicialización de marcas y medida de las mismas. En cursiva se muestra el apartado de la Tesis asociado a la tarea.	73

4.17. Tiempos de cómputo promedio para las diferentes alternativas. Se muestra el tiempo de proceso total en cada frame (azul), tiempo dedicado a lectura de imagen y/o corrección de la misma (verde), tiempo empleado en medidas (amarillo) y tiempo dedicado a la inicialización de nuevas marcas (rojo)	74
4.18. Representación de la zona de selección de marcas	75
4.19. Vector de medida original h_{iorig} y actual h_i . Representación de los criterios de visibilidad	76
4.20. Áreas de búsqueda en las proyecciones de la marca, en función de su incertidumbre	80
4.21. Detección de extremos en el espacio de escala	82
4.22. Keypoints candidatos dentro del área de búsqueda	84
4.23. Marcas SIFT en un frame	84
4.24. Proceso de correlación píxel a píxel	85
4.25. Número de correlaciones correctas por marca para los casos de correlación directa y SIFT (media y desviación típica). Este último caso se estudia para diferentes tamaños de área de búsqueda	86
4.26. Tiempos de cómputo totales por frame para los casos de correlación directa y SIFT (media y desviación típica). Este último caso se estudia para diferentes tamaños de área de búsqueda	86
4.27. Estimación del camino de prueba recorrido por el vehículo usando SIFT y correlación directa. La referencia (ground truth) aparece como línea continua	87
4.28. Error acumulativo medio para el eje X y Z en función del número de frame n . Los resultados se muestran tanto para la implementación SIFT como para el método propuesto	88
4.29. Proceso de transformación de la apariencia del patch	89
4.30. Proceso de búsqueda progresiva del píxel ocupado más cercano	89
4.31. Tiempos de vida de las marcas sin aplicar el método de transformación de patch. Se muestra el número de correlaciones exitosas para cada marca.	90
4.32. Tiempos de vida de las marcas aplicando el método de transformación de patch. Se muestra el número de correlaciones exitosas para cada marca.	91
4.33. Tiempos de cómputo totales empleando correlación directa o la técnica del patch transformado	91
4.34. Ventana de búsqueda inicial	93
4.35. Región de búsqueda aleatoria	94
4.36. Estados posibles de las marcas	94
4.37. Procedimiento de detección de características	95
4.38. Procedimiento de detección de características	96
4.39. Procedimiento de detección de características	96
4.40. Procedimiento de detección de características	96
4.41. Búsqueda de correspondencia epipolar. La zona sombreada en amarillo representa el área reducida de búsqueda entorno a la proyección esperada en la imagen derecha	98
4.42. Distancia estimada de de una marca en función de la disparidad $d = u_L - u_R$. .	99
4.43. Suposición de la situación de los patches respecto a la cámara	102
4.44. Representación del ruido de medida de 1 píxel	105
5.1. Representación gráfica del método híbrido de SLAM de dos niveles jerárquicos. Cada sub-mapa tiene una huella asociada a éste	113
5.2. Estructura básica del mapa topológico. Cada huella (nodo) lleva asociado el sistema de referencia local de su propio submapa métrico. El sistema de referencia global es coincidente con el sistema local de la primera huella.	114
5.3. Representación del primer umbral de giro θ_1 , seguido del segundo θ_2	115

5.4. Gestión de alto nivel del mapa	116
5.5. Primer paso en la obtención de P_{rob}^G , en función de la covarianza local relativa del vehículo $P_{rob}^{fp_l}$ (en rojo) y de la covarianza global absoluta de la huella actual $P_{fp_l}^0$ (en gris).	117
5.6. Segundo paso en la obtención de P_{rob}^G . $P_{fp_l}^0$ se obtiene en función de la covarianza local relativa de la huella actual respecto a la anterior $P_{fp_l}^{fp_{l-1}}$ (en rojo) y de la covarianza global absoluta de la huella anterior $P_{fp_{l-1}}^0$ (en gris)	118
5.7. Proceso de creación de un nuevo sub-mapa. Las marcas pertenecientes al sub-mapa anterior visibles en el nuevo (magenta) se eliminan del primero y se incorporan como marcas iniciales en el segundo (verde).	119
5.8. Ejemplo de grafo de alto nivel. Las huellas SIFT se marcan en rojo, generándose éstas en las <i>esquinas</i> del recorrido.	121
5.9. Proceso de búsqueda de lugares previsitados. Los nodos circulares representan las diferentes huellas a lo largo del recorrido del vehículo. En el instante mostrado, una nueva huella SIFT sf_x está siendo evaluada. Dentro de la región de incertidumbre global P_{rob}^G se encuentran dos huellas SIFT previamente obtenidas, que se han destacado en amarillo, candidatas a ser reconocidas.	122
5.10. Líneas de unión entre marcas SIFT inicialmente emparejadas. Se muestran las probabilidades de <i>matching</i> sobre las mismas huellas usando RANSAC comparado con el uso del método basado en varianzas	124
5.11. Representación del incremento progresivo en la incertidumbre global del vehículo	125
5.12. Proceso de corrección en la posición de las marcas. En rojo se muestra la pose inicial del vehículo antes de la corrección. En verde se muestra la pose una vez corregida. En azul se muestra la interpolación ponderada por la huella asociada a la marca en cuestión	126
5.13. Representación del mapa global una vez corregido	126
5.14. Gráfico del proceso de resolución de la ecuación por relajación. En sentido descendente se reflejan las iteraciones de discretización y en sentido ascendente se muestran las interpolaciones.	128
5.15. Representación del mapa topológico de alto nivel. Las huellas están representadas con círculos rojos en los que el triángulo indica la orientación del vehículo.	130
5.16. Representación de las incertidumbres globales del vehículo P_{rob}^G , incrementándose a lo largo del camino del vehículo en cada una de las poses de las huellas. Las incertidumbres globales marcadas en color rojo están asociadas a las huellas SIFT. Los números denominan a cada huella. El grafo, además muestra un ejemplo de selección del camino más corto para el cálculo de la incertidumbre global después de una situación de cierre de lazo.	131
5.17. Detalle de una sección de una ruta de prueba en la que se aprecian las incertidumbres globales de las marcas	132
5.18. Estimaciones del mapa realizadas usando tanto el MLR como el método lineal, así como la estimación de referencia (<i>ground truth</i>). Por claridad sólo se muestra la estimación de la trayectoria del vehículo.	133
5.19. Errores de estimación en el eje X usando el MLR y el método lineal.	133
5.20. Errores de estimación en el eje Z usando el MLR y el método lineal.	134
5.21. Errores medios de estimación en el eje X, en el Z y en la altura en función de las pendientes medias de los diferentes recorridos analizados	135
5.22. Relación entre el error medio σ_R y el valor de UERE.	136

5.23. Fusión de información visual y GPS. Cada vez que se toma una nueva medida de GPS se resetea el filtro de bajo nivel y las medidas son, a partir de ese momento, referidas a la actualización de GPS anterior.	138
5.24. Ejemplo de fusión de distribuciones de incertidumbre. La imagen superior muestra las distribuciones originales. La imagen inferior muestra el resultado de la fusión.	139
5.25. Detalle de un sub-mapa en el que se muestra la estimación de la orientación del vehículo a partir de la fusión con los datos del GPS. La estimación del mapa local se representa a través de las estimaciones de pose locales (triángulos negros). La flecha gris indica la orientación estimada del vehículo en el segundo punto de medida de GPS	140
5.26. Diagrama de huellas MLR antes (izquierda) y después (derecha) de la recuperación de la señal de GPS. Las huellas en las que el GPS estaba disponible están expresadas en coordenadas globales (19..33). Las huellas sin GPS disponible están expresadas relativas a la huella anterior.	141
5.27. Diagrama MLR de huellas antes de la recuperación de la señal de GPS. Se muestran las incertidumbres globales de las huellas de un tramo con GPS disponible, aproximadamente constantes con el tiempo (color verde). La última huella introducida termina un tramo sin GPS disponible, apreciándose un incremento notable en el tamaño de la región de incertidumbre global (color rojo).	142
5.28. Diagrama MLR de huellas después de la recuperación de la señal de GPS. Se muestran las incertidumbres globales del tramo con GPS sin apenas variación. Tras la optimización del mapa, una vez recuperada la señal de GPS, la incertidumbre al final del tramo sin GPS se reduce notablemente. Se aprecia también que este último tramo es el que ha sufrido una mayor corrección.	143
6.1. Esquema del montaje básico, compuesto por una cámara estéreo conectada a un ordenador portátil.	147
6.2. Cámara Unibrain Fire-i utilizada en el sistema.	148
6.3. Lente de gran angular utilizada en la cámara.	149
6.4. Montaje de la cámara estéreo con conexionado en serie para interiores.	149
6.5. Sistema desarrollado para entornos interiores.	150
6.6. Esquema del montaje para exteriores, basado en el hardware básico añadiéndole un GPS de bajo coste. Como sistema de referencia se utiliza un RTK-GPS y la estación base para las correcciones de posición.	151
6.7. Vehículo utilizado para la implementación en exteriores. Abajo a la izquierda: vista frontal del vehículo con la cámara estéreo incorporada. Abajo a la derecha: Vista interior del vehículo con el PC portátil y el GPS de bajo coste.	152
6.8. Montaje de la cámara estéreo para exteriores. Abajo se muestra el PC portátil con la tarjeta de adquisición IEEE1394 conectada.	152
6.9. GPS de bajo coste utilizado en el sistema.	153
6.10. RTK-GPS utilizado como referencia. Abajo a la izquierda: Ubicación del RTK-GPS dentro del vehículo de pruebas. Abajo a la derecha: Antena receptora de señal GPS + correcciones de posición vía radio.	153
6.11. Vista aérea del camino recorrido por el robot, marcada en color verde. La flecha roja indica la posición de inicio y cierre de lazo.	154
6.12. Estimación del camino recorrido por el robot realizada por el bajo nivel. La referencia se dibuja en línea continua	154
6.13. Error acumulativo medio para el eje X y Z en función del número de frame n.	155

6.14. Representación del mapa en una situación de cierre de lazo. Las marcas de bajo nivel se representan de color amarillo, mientras que las visibles y correctamente medidas se representan en color rojo. Los números verdes muestran las ubicaciones de las huellas. La posición del robot se indica en el centro de una gran cruz azul.	156
6.15. Tiempos de cómputo por frame a lo largo del recorrido, representados en rojo. El límite de tiempo real se representa como una línea constante de color negro en 33 ms.	157
6.16. Vista aérea del camino A recorrido por el vehículo (referencia) dibujado en color magenta. El punto de partida se indica con el flag <i>S</i> . El punto de cierre del primer lazo se indica con el flag <i>1</i> . El punto de cierre del segundo lazo, y final del recorrido, se indica con el flag <i>E</i>	158
6.17. Causas del error de estimación en el último tramo. En las imágenes superiores se observa la oclusión casi total de las mismas por parte de un autobús. En las imágenes inferiores se muestra el entorno del último tramo tras la oclusión, en el que se aprecia la ausencia casi total de texturas donde extraer marcas visuales relativamente cercanas.	159
6.18. Estimación del camino recorrido. La línea rayada representa la estimación antes del cierre del lazo. La línea punto-raya representa la estimación después del cierre del lazo. La referencia está representada en línea continua.	160
6.19. Representación del error en la estimación del recorrido en los ejes X y Z respecto de la referencia después de la corrección global	161
6.20. Tiempos de procesamiento totales a lo largo de la mitad del recorrido del vehículo.	161
6.21. Vista aérea del recorrido de prueba B. Los puntos de inicio y final se indican en la figura.	162
6.22. Vista aérea del recorrido de prueba C. El punto de inicio se indica en la figura.	163
6.23. Estimación del recorrido B usando el método propuesto (rojo) y la referencia (línea rayada).	163
6.24. Error en la estimación del recorrido B usando el sistema propuesto respecto a la referencia.	164
6.25. Estimación del recorrido C usando el método SLAM visual propuesto (rojo) y la referencia (línea rayada).	165
6.26. Error en la estimación del recorrido C usando un GPS estándar (línea rayada) y el sistema de SLAM (línea continua) respecto a la referencia.	166
6.27. Error medio de estimación del recorrido C usando un GPS estándar (línea rayada) y el sistema de SLAM (línea continua) respecto a la referencia.	167
6.28. Tiempos de cómputo totales por frame para el recorrido B.	168
6.29. Tiempos de cómputo totales por frame para el recorrido C. El límite de tiempo real se representa como una línea constante en 33 ms. Los frames en los que se produjeron cierres de lazo se marcan con líneas verticales rojas.	168
6.30. Vista aérea del camino recorrido por el vehículo. Se muestran las marcas de bajo nivel como puntos en rojo. El lazo que se muestra en la parte inferior del mapa contiene diversos edificios que aún no aparecían en el mismo.	170
6.31. Vista en perspectiva del camino recorrido por el vehículo (línea continua). Las marcas de bajo nivel se muestran como puntos.	171
6.32. Estimación del recorrido D usando únicamente un GPS de bajo coste (línea punteada), sistema SLAM combinado (línea punto-raya) y la referencia (línea rayada). Los tramos de pérdida de GPS se muestran en línea gruesa de color rojo.	171
6.33. Error en la estimación del recorrido D usando un GPS estándar (arriba) y el sistema de SLAM combinado (abajo) respecto a la referencia.	172

6.34. Vista aérea del camino recorrido por el vehículo. Los puntos inicial y final de recorrido se indican con una estrella. Los tres tramos numerados carecían de cobertura GPS.	173
6.35. Estimación del recorrido E usando únicamente un GPS de bajo coste, el sistema SLAM combinado y la referencia. Los tramos de pérdida de GPS se muestran en línea gruesa de color rojo.	174
6.36. Error en la estimación del recorrido E usando un GPS estándar (arriba) y el sistema propuesto (abajo) respecto a la referencia. El tramo sin referencia se marca en color rojo.	175
6.37. Vista aérea del camino recorrido por el vehículo. El punto inicial de recorrido se indica con una estrella verde, mientras que el final se muestra con una estrella roja.	176
6.38. Estimación del recorrido F usando únicamente un GPS de bajo coste, el sistema SLAM combinado y la referencia. Los tramos de pérdida de GPS se muestran en línea gruesa de color rojo.	176
6.39. Error en la estimación del recorrido F usando un GPS estándar (arriba) y el sistema de SLAM combinado (abajo) respecto a la referencia.	178
6.40. Diagrama funcional del sistema de asistencia a la navegación propuesto, basado en visión y GPS.	180
6.41. Mobile PC empleado para emular el dispositivo navegador.	181
6.42. Detalle de los tramos 2 y 3 del recorrido D utilizado para las pruebas. Se indican los dos giros realizados por el vehículo en ausencia de señal de GPS	182
6.43. Secuencia de imágenes del display del navegador en el momento del giro del tramo 2. El orden temporal es de arriba a abajo. La posición y orientación del vehículo estimadas se muestran con una flecha azul.	183
6.44. Secuencia de imágenes del display del navegador en el momento del giro del tramo 3. A la derecha se muestran las imágenes del vehículo y su entorno correspondientes a las posiciones mostradas por el navegador (izquierda). El orden temporal es de arriba a abajo.	184

Índice de tablas

2.1. Rendimiento de diferentes algoritmos de SLAM frente a <i>treemap</i>	29
5.1. Contribuciones al UERE.	136
6.1. Resoluciones y modos de color disponibles en función de la velocidad de muestreo de la cámara.	148
6.2. Datos relevantes para el recorrido en interiores.	155
6.3. Datos relevantes para el recorrido A.	162
6.4. Datos relevantes para el recorrido B.	167
6.5. Datos relevantes para el recorrido C.	169
6.6. Datos relevantes para el recorrido D.	177
6.7. Datos relevantes para el recorrido E.	177
6.8. Datos relevantes para el recorrido F.	179
6.9. Resumen de los datos más relevantes para los diferentes recorridos presentados. .	179
6.10. Robustez de la detección de lugares previsitados ante cambios de iluminación. En la tabla se indica: % Falsos positivos / % Falsos negativos.	179

Capítulo 1

Introducción

1.1. Introducción

En los últimos años el uso de navegadores GPS se ha extendido entre los conductores de vehículos. Actualmente, la mayoría de los fabricantes ofrecen este sistema como un accesorio integrado dentro de la estructura del propio vehículo. Asimismo, existe otro mercado paralelo que vende navegadores GPS portátiles para el público en general, fundamentalmente orientados para su uso en vehículos, donde es el propio usuario el responsable de su instalación. Las causas de la gran proliferación de estos sistemas son las siguientes: por un lado se trata de una tecnología de localización madura y de uso mundial, por otro lado, ofrece una solución razonable a uno de los principales problemas de la conducción que consiste en saber cómo llegar a un destino, y por último, la progresiva reducción de los costes de los mismos ofrece una buena relación calidad precio que le ha permitido pasar de un producto de lujo, al alcance de muy pocos, a un producto comercial de uso masivo.

A pesar de las grandes ventajas que introducen los navegadores GPS de bajo coste, muestran ciertas limitaciones de uso fundamentalmente en zonas urbanas. En este tipo de entornos existen generalmente una gran variedad de elementos que obstruyen o dificultan la recepción de la señal GPS, como por ejemplo edificios altos, árboles y sobre todo túneles. Los errores provocados por estos elementos son generalmente filtrados por algoritmos que tienen en cuenta las rutas planificadas y el mapa en sí. Sin embargo, dependiendo de las dimensiones del error, el sistema de planificación puede llegar a fallar, provocando el cálculo de rutas erróneas con la consecuente distracción del conductor. Este tipo de problemas es más frecuente en situaciones en las que el vehículo circula por un túnel o paso subterráneo donde la información de localización no está disponible en absoluto.

Otra de las limitaciones de los navegadores GPS actuales es el uso de mapas no actualizados o con errores. Hay que tener en cuenta que la responsabilidad de las redes viarias está muy fragmentada (nacional, autonómica, local) lo que provoca multitud de actuaciones de mejora y cambios de las mismas, a veces no suficientemente informadas. A esto habría que añadir los cambios temporales de trayectorias debidos a obras. Todo ello origina que algo tan básico como tener un mapa cien por cien fiable pueda ser un problema. En consecuencia, el navegador puede sugerir seguir una ruta inexistente o prohibida, lo que provoca distracción en el conductor y pérdida de confianza en el navegador.

La mejora de las limitaciones de localización y mapeado existentes en los sistemas de navegación GPS actuales y la ausencia de sistemas SLAM para entornos urbanos de grandes dimensiones suficientemente fiables han motivado la realización de esta Tesis.

Como paradigma de estas investigaciones se encuentra la posibilidad de desarrollar un sistema de guiado autónomo que, idealmente, permitiría a un vehículo circular de forma autónoma y sin asistencia del conductor. La historia de los vehículos autónomos comienza en 1977 con el Laboratorio de Ingeniería Mecánica Tsukuba en Japón. Usando un recorrido dedicado y claramente marcado, se alcanzaron velocidades de hasta 30 Km/h, por medio del seguimiento de marcas de color blanco integradas en la calzada. Para ello fue necesario un hardware especial ya que la capacidad de cómputo era mucho menor que hoy en día.

En la década de los 80, un mono-volumen guiado por visión diseñado por Ernst Dickmanns y su equipo de la Universidad de Bundeswehr en Munich, Alemania, alcanzó los 100 Km/h en calles sin tráfico. Paralelamente, la agencia DARPA en los Estados Unidos consiguió la primera demostración de un sistema de seguimiento en carretera, usando un radar láser (Instituto de Investigación de Michigan), visión artificial (Universidad Carnegie Mellon y SRI) y un sistema de control automático para controlar un vehículo sin conductor a velocidades de hasta 30 Km/h. En 1995, el proyecto Navlab de la Universidad Carnegie Mellon, consiguió realizar una conducción

autónoma en un 98.2%, en un recorrido de 5000 Km. Este vehículo, sin embargo, era semi-autónomo por naturaleza, ya que usaba redes neuronales para controlar la dirección, mientras que el freno y el acelerador eran controlados manualmente. Tras diversos proyectos llevados a cabo por diferentes instituciones de investigación, en 2002 la agencia estadounidense DARPA anunció una serie de competiciones de vehículos completamente autónomos, denominadas DARPA Grand Challenge. Estas competiciones permitieron a equipos internacionales competir en carreras en campo abierto sobre terreno no urbano. En 2007 se celebró la primera competición sobre territorio urbano, incluyendo tráfico, intersecciones, aparcamientos, etc. Esta competición se denominó DARPA Urban Challenge, llegando a completarla seis de los vehículos participantes (ver Fig. 1.1).



Figura 1.1: *Imagen superior*: Vehículos autónomos participantes en la carrera DARPA Urban Challenge. *Imagen inferior*: Hardware de control y sensores de distancia láser empleados en uno de los vehículos participantes.

En el diseño de un vehículo autónomo existen numerosos aspectos a tener en cuenta, como son: tipos de sensores, sistemas de control, métodos de navegación, sistemas de evitación de obstáculos y generación de mapas, que forman la base sobre la que se implementan los sistemas de navegación. Dentro de estos últimos, los métodos de Localización y Mapeado Simultáneo (SLAM) de entornos han acaparado el interés de los investigadores en los últimos años, no sólo por su evidente aplicación en sistemas de navegación autónomos encargados de planificar rutas, fundamentalmente en el área de la robótica, sino por su utilidad en otros sistemas de asistencia a la conducción, como los de ayuda a la navegación. Dentro de éstos últimos, los métodos de SLAM permitirían, por ejemplo, la generación de un mapa de forma *online*, corrigiendo los errores en el mapa predefinido.

1.2. Problemática asociada a la localización y mapeado simultaneos de un vehículo en grandes entornos

Existen diversas dificultades asociadas a resolver el problema de la localización y mapeado de grandes entornos urbanos, de una manera fiable y suficientemente precisa. Tomando como base las causas que los generan, los problemas se pueden clasificar en tres grandes grupos:

1.2.1. Derivados del sensor utilizado

El uso de visión artificial para la localización y reconstrucción de entornos ha sido ampliamente utilizado, sobre todo recientemente. A pesar de las numerosas ventajas de este tipo de sensores, éstos están sujetos a diversos problemas. Uno de ellos es el efecto de la iluminación, esto incluye problemas de contraluz, cambios del nivel de luminosidad, falta de textura debido al ángulo de incidencia de la luz, etc. La lluvia, por otro lado, condiciona la visibilidad disponible por la cámara, debido a las gotas depositadas en el parabrisas. Por otra parte, la dificultad en la calibración de las cámaras, sobre todo si éstas poseen lentes con una elevada distorsión, puede conllevar un determinado error en las medidas realizadas. Adicionalmente la elección de marcas visuales adecuadas para su seguimiento, presenta problemas en entornos urbanos, en los que la variedad de objetos y características dificulta esta elección. Además, el uso de cámaras estéreo conlleva la posibilidad de aparición de fallos en las correspondencias entre imágenes.

En cuanto al uso de sensores GPS, la problemática fundamental de éstos está asociada a la disponibilidad y calidad de la señal de los satélites. En primer lugar, como se ha explicado anteriormente, la existencia de edificios altos, árboles o túneles en los entornos urbanos conlleva una pérdida de precisión e incluso ausencia de la estimación de la posición por parte del GPS. Otro aspecto a tener en cuenta es la frecuencia de actualización de estos dispositivos, generalmente no superior a 1 Hz. En cuanto a la integración de la información proporcionada por el GPS en el sistema, la principal problemática es la imposibilidad de obtener una estimación precisa de la propia incertidumbre de los datos, dependiente de factores que no son completamente medibles. Por último, hay que tener en cuenta que la información de orientación del vehículo no es proporcionada por los sensores GPS.

1.2.2. Derivados del entorno

El primer problema a abordar, derivado del entorno, es el tamaño de éste. A medida que se pretende localizar y, sobre todo mapear, en tiempo real entornos de mayores dimensiones, aparecen problemas de consistencia del mapa, cada vez más importantes. El uso de sensores y métodos que conllevan errores acumulativos provoca que, a medida que se recorren nuevos lugares la estimación de la pose (posición y orientación) del vehículo, así como del mapa incrementa su error absoluto. Si se pretende no incorporar información a priori del entorno de trabajo, estos errores serán inevitables, a menos que se implementen métodos que permitan reconocer localizaciones anteriores, con un error asociado menor al actual.

Por otro lado, el hecho de destinar el sistema a entornos de tipo urbano conlleva una serie de problemas derivados de la impredecibilidad de éstos. Los sistemas de estimación de la pose basados en visión artificial, generalmente asumen que el entorno observado es estático. De esta forma, todo movimiento percibido por la cámara será debido al movimiento de ésta misma. La realidad, fundamentalmente en los entornos urbanos, es bien distinta, ya que existen numerosos elementos móviles en dicho entorno, como por ejemplo, peatones, otros vehículos, árboles movidos por el viento, etc. El efecto perjudicial de estos objetos es mayor cuanto mayor es el porcentaje de

la imagen ocupada por éstos, ya que el sistema no dispone de información suficiente para estimar el verdadero movimiento propio. En la Fig. 1.2 se muestran dos ejemplos de variabilidad del entorno. En el caso del autobús, se observa como el objeto ocupa un porcentaje considerable de la imagen, mientras que en la imagen inferior se muestra cómo un segundo automóvil adelanta al vehículo de pruebas, circulando a una velocidad cercana a la de éste, incumpliendo la suposición de entorno estático.



(a) Secuencia de imágenes que muestra el momento de oclusión parcial del campo visual por un vehículo de grandes dimensiones en movimiento



(b) Secuencia que muestra dos instantes en los que un vehículo en movimiento, a la izquierda de la imagen, circula a una velocidad cercana a la del vehículo de pruebas y en el mismo sentido. Esta situación incumple la suposición de un entorno estático.

Figura 1.2: Situaciones de entorno variable.

1.2.3. Derivados de los requerimientos del sistema

Los sistemas de asistencia a la navegación, en general, y de navegación autónoma, en particular, deben ser capaces de responder con la mayor rapidez posible ante situaciones imprevistas. Este tipo de situaciones puede darse cuando el recorrido por una calle determinada, que previamente había sido mapeada, esté cortado temporalmente. En ese caso es importante conocer la posición actual del vehículo para poder tomar la decisión de modificar la ruta por una determinada calle u otra en función de esta posición. Para ello el sistema debe ser capaz de funcionar en *tiempo-real*, de forma que se obtenga una nueva pose cada 33 ms, tomando éste como límite asociado a una velocidad de captura estándar de 30 imágenes por segundo.

Por otro lado, el sistema empleado deberá almacenar los diversos mapas en función de la ruta en cuestión. Dado que los dispositivos de almacenamiento de datos tienen una capacidad limitada, la cantidad de memoria necesaria para almacenar los mapas deberá ser la mínima posible.

En cuanto a la distribución física del sistema, éste no deberá ser demasiado voluminoso ni aparatoso, permitiendo una integración lo más fácil posible en el vehículo. Además, de cara a los sistemas de ayuda a la conducción, los equipos y sensores necesarios no deberán entorpecer la maniobrabilidad del conductor, así como obstruir la visión del mismo.

1.3. Planteamiento y objetivos iniciales

El objetivo fundamental del sistema a desarrollar en esta Tesis será la localización de un vehículo en grandes entornos fundamentalmente urbanos, así como la creación de un mapa no denso consistente basado en marcas visuales. Para lograr este doble objetivo, se dispondrá de sensores tanto de tipo visual como GPS, así como equipos para realizar las tareas de procesamiento.

Como subobjetivos se plantean los siguientes:

1. Un primer subobjetivo es el diseño de un sistema SLAM, utilizando los sensores anteriormente mencionados. Se tendrán en cuenta todos los aspectos necesarios tanto para la generación del mapa como para la localización del vehículo dentro de él.
2. Adicionalmente, otro subobjetivo planteado consiste en la mejora de los sistemas de asistencia a la navegación, es decir, los conocidos como navegadores GPS actuales. El sistema deberá poderse integrar con un navegador GPS estándar sin modificación alguna. Como resultado de la integración, el navegador dispondrá de información de pose más precisa y fiable, siendo capaz de planificar y mostrar al conductor en tiempo real rutas, de una manera continua y sin errores. Esto debe ser posible a pesar de circular por calles con edificios altos, túneles con intersecciones, etc. Además, en caso de encontrar inconsistencias entre el mapa almacenado y el creado por el sistema de mapeado, se modificará el primero y se almacenará de nuevo.
3. Un tercer subobjetivo secundario, que ha surgido como consecuencia de la metodología utilizada, ha sido la evaluación del sistema propuesto en entornos interiores. Para ello, el requerimiento será la localización de un robot móvil en un entorno interior y el consiguiente mapeado del mismo. Posteriormente, el sistema será adaptado para ser embarcado en un automóvil y realizar las consiguientes pruebas en entornos exteriores, urbanos y semi-urbanos, de grandes dimensiones.

Como requerimientos generales adicionales, cabe destacar la necesidad de aportar la capacidad de un tiempo de reacción suficientemente bajo. Este requerimiento es necesario fundamentalmente para los sistemas de navegación autónomos, en los que es de suma importancia conocer la posición actual del vehículo para poder tomar la decisión de modificar la ruta en función de las necesidades. Por este motivo, se impone el requerimiento de proporcionar la información de pose en tiempo real.

Además, como se ha explicado anteriormente, el sistema deberá estar adaptado para poder localizar un vehículo y realizar un mapa en un entorno de grandes dimensiones. Esto implica el desarrollo de métodos que garanticen el mantenimiento del límite de tiempo real en todo el recorrido. Además, el sistema deberá ser capaz de almacenar todo el mapa en memoria, teniendo en cuenta el tamaño de éste.

En cuanto a la precisión del sistema, ésta debe ser lo suficientemente alta como para distinguir si el vehículo se encuentra en una calle o en otra, sobre todo en el ámbito de la mejora del sistema

de asistencia a la navegación. Esto será de vital importancia especialmente en situaciones de toma de decisiones, por ejemplo ante un cruce o desviación, de forma que sea posible discernir la elección tomada lo antes posible. Además, esto implicará una mayor dificultad en los tramos en los que el sensor GPS no esté disponible o carezca de la suficiente precisión.

Con el objeto de una posible implantación comercial asociada a sistemas de ayuda a la conducción, el coste total del sistema debe ser lo más bajo posible. Esto implica la utilización de cámaras de baja gama, ordenadores personales de tipo estándar y sensores GPS de bajo coste.

Por último, el sistema deberá estar diseñado de forma que permita su integración dentro de otro sistema de navegación autónoma. Embarcado en un vehículo, el propósito sería realizar recorridos repetitivos en entornos urbanos. El ejemplo más claro de este tipo de sistemas es una línea de autobús regular urbano. Este tipo de vehículos realizan el mismo recorrido de una forma iterativa, con lo que el sistema sería capaz de refinar el mapa estimado a medida que se recorren los mismo lugares. El sistema debería poder implantarse sin información previa y realizarse las primeras estimaciones del mapa desde el principio.

1.4. Estructura de la Tesis

La estructura de esta Tesis se divide en los siguientes apartados/capítulos:

- Una vez presentada la introducción, en el Capítulo 2 se describen y analizan los trabajos más relevantes dentro del estado del arte, convenientemente clasificados. Además se realizará un análisis global de las aportaciones de los distintos trabajos, extrayendo los aspectos más beneficiosos para el sistema a desarrollar en esta Tesis y explicando las posibles aportaciones sobre las carencias detectadas. Por último, se detallarán los objetivos específicos identificados para el sistema presentado.
- Posteriormente, en el Capítulo, 3 se explicará de forma general la estructura del sistema presentado, indicándose las principales funcionalidades.
- El Capítulo 4 presenta las técnicas, estudios e implementaciones llevadas a cabo para realizar las tareas de localización y mapeado de forma local. En él fundamentalmente se describe la implementación del filtro de Kalman Extendido (EKF), teniendo en cuenta los sensores utilizados y el modelo del vehículo.
- Posteriormente, en el Capítulo 5 se presentan los métodos desarrollados e implementados para adecuar el diseño del sistema a su uso en grandes entornos. Además, se describirá la estrategia empleada para la integración del sensor GPS, en los diferentes niveles jerárquicos.
- En el Capítulo 6 se procederá a validar los diferentes aspectos del sistema desarrollado. Para ello se presentan diversas gráficas y comparativas, producto de los datos obtenidos en diversos ensayos en entornos reales; tanto interiores como, fundamentalmente, exteriores. Se muestran, además, resultados prácticos de la aplicación del sistema a la mejora de un sistema de asistencia a la navegación.
- Por último, en el Capítulo 7 se muestran las conclusiones finales y futuras líneas de investigación.

Capítulo 2

Estado del arte y objetivos de la Tesis

2.1. Estado del arte

El estudio del problema del SLAM comenzó a gestarse en 1986 en el Congreso de Robótica y Automática del IEEE celebrado en San Francisco, California. Un determinado grupo de investigadores habían estado considerando la posibilidad de aplicar métodos teóricos de estimación para resolver los problemas de mapeado y localización. Como resultado de las diversas discusiones se llegó a la conclusión de que el mapeado probabilístico consistente era un problema fundamental en robótica, con problemas conceptuales y computacionales importantes que debían ser resueltos. Años más tarde, se presentaban los primeros trabajos destacables en este área. Los trabajos de Smith y Cheesman [1] y Durrant-Whyte [2] establecieron la base estadística para describir las relaciones entre las marcas y la manipulación de la incertidumbre geométrica.

Hoy en día los métodos de SLAM han alcanzado un considerable estado de madurez. Los retos más importantes se centran actualmente en métodos que permitan la implementación en entornos de gran escala y, por otra parte, entornos cada vez menos estructurados, especialmente en situaciones donde las soluciones basadas en otras tecnologías como el GPS no están disponibles o son poco fiables: zonas urbanas con edificios altos, bajo los árboles, en minas, en robots de interiores, en robots de rescate, en aplicaciones subacuáticas o en robots espaciales (exploradores lunares, en Marte, etc.). En los trabajos [3] y [4] se presenta un compendio de las investigaciones más relevantes llevadas a cabo hasta la fecha en el área del SLAM.

En este capítulo se realiza una revisión de los trabajos más relevantes presentados recientemente en esta línea de investigación. Existe diversas alternativas de clasificación en función de diversos parámetros como pueden ser: tipo de sensores utilizados, métodos empleados o entornos objetivo. Teniendo en cuenta la evolución de los sistemas de SLAM, en primer lugar se efectuará una revisión de los trabajos orientados a pequeños entornos, generalmente interiores. Posteriormente se repararán los nuevos sistemas orientados a entornos de mayores dimensiones, analizando las soluciones aportadas con respecto a los sistemas de pequeños entornos. Por último, se focalizará el estudio en los sistemas basados en sensores GPS, analizando fundamentalmente los métodos de fusión de los mismos con otros sensores.

2.1.1. Sistemas SLAM para pequeños entornos

Originalmente los métodos de SLAM fueron concebidos para su implementación en robots móviles, los cuales debían reconocer un entorno nuevo a la vez que localizarse en él. Este tipo de entornos eran generalmente de pequeñas dimensiones, en la mayoría de los casos interiores. A continuación se realiza una revisión de los trabajos más relevantes, algunos de los cuales han servido de base para esta Tesis.

a) *Sistemas basados en EKF.*

Uno de los métodos más ampliamente utilizado a la hora de modelar dinámicamente el comportamiento de un móvil es el EKF (*Extended Kalman Filter*, “Filtro de Kalman Extendido”) [5]. Como es sabido, el filtro de Kalman intenta solucionar el problema general de estimar el estado del un proceso gobernado por una ecuación estocástica *lineal*. Sin embargo, en la mayoría de los casos, incluido el SLAM, dicho proceso o las medidas a realizar no son lineales. El filtro de Kalman extendido afronta este hecho a través de la linealización en torno a la media y la covarianza actuales.

Uno de los trabajos más importantes en SLAM visual para pequeños entornos es el llevado a cabo por Andrew J. Davison [6]. Esta investigación comenzó inicialmente usando un sistema de visión estéreo activo con el objeto de estimar la posición 2D de un robot

móvil. En este trabajo se presenta un sistema completamente automático, capaz de detectar, almacenar y seguir determinadas marcas visuales naturales a medida que el robot navega hacia su objetivo. El robot elige cuál de las marcas disponibles deberá seguir, en un determinado momento para mejorar el conocimiento de su propia pose, decidiendo además cuándo es el momento de buscar nuevas marcas. Los resultados de la localización mejoran con respecto a los alcanzados usando únicamente odometría, mostrando ventajas significativas sobre técnicas denominadas SFM (*Structure From Motion*, “Estructura a partir del Movimiento”). La propagación de la incertidumbre en la estimación de las posiciones del robot y de las marcas a medida que el robot se mueve por el entorno, se tiene en cuenta de una forma rigurosa, mostrándose su importancia para evitar la deriva en el movimiento. En la Fig. 2.1 se muestra la arquitectura del robot empleado en el sistema.



Figura 2.1: Robot equipado con cámara estéreo de visión activa.

Posteriormente, en el trabajo [7] la línea de investigación del mismo grupo de trabajo se ha centrado en el SLAM 3D en tiempo real usando visión monocular, utilizando el filtro EKF. En este trabajo se presenta un método Bayesiano para la localización de una única cámara (ver Fig. 2.2), a través del mapeado de un conjunto de marcas naturales dispersas. Para ello, se hace uso de un modelo de movimiento aleatorio, adecuado para el movimiento manual de la cámara, así como de una estrategia basada en la información para la medida de marcas. Particularmente difícil es la tarea de inicializar marcas en tiempo real con una única cámara, tarea que se realiza usando una estrategia de muestreo con filtro de partículas. Una de las mayores dificultades a la hora de resolver el problema del SLAM, es el proceso de construcción del mapa. Debido al error en las medidas, es normal la aparición de una deriva acumulada durante la construcción del mapa. Esto conduce a una correspondencia incorrecta del mapa cuando la cámara captura lugares ya visitados. Por lo tanto, para conseguir una localización de la cámara correcta y estable, es necesario incluir el mapa completo dentro del vector de estado del EKF. De esta forma, el mapa irá aumentando en número de marcas a medida que la cámara se mueve por el entorno y visita nuevos lugares. El manejo de la incertidumbre en tiempo real permite la reidentificación y correcta asociación de marcas previamente medidas, lo que conduce al reconocimiento de lugares ya visitados, permitiendo la correcta localización de la cámara después de periodos sin medidas disponibles. En la Fig. 2.3(a) se muestran resultados de mapeado y localización de la cámara en un entorno reducido. En la Fig. 2.3(b) se muestra

el momento de inicialización de una nueva marca, en el que se comienza estimando una recta para, finalmente, converger hacia una marca con una distribución de incertidumbre asociada.

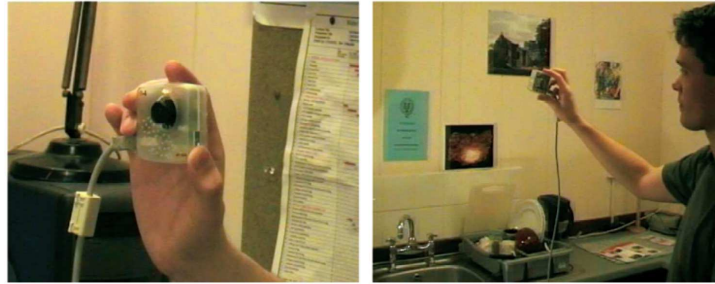
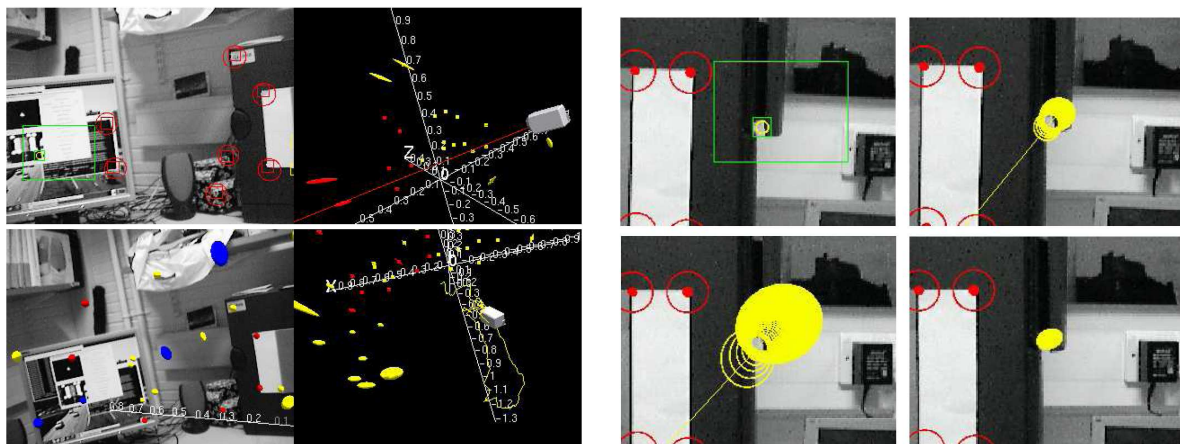


Figura 2.2: Cámara utilizada para los experimentos.



(a) Dos instantes del proceso de SLAM, partiendo de la imagen superior y terminando por la inferior. En ellas se muestran las proyecciones de las marcas y sus covarianzas sobre la imagen capturada por la cámara, a la izquierda. A la derecha se muestra el mapa de marcas estimado, junto con la posición de la cámara.

(b) De izquierda a derecha y de arriba a abajo se muestran instantes consecutivos del proceso de inicialización de una nueva marca. Se aprecia la recta inicial de estimación y la convergencia final en una marca.

Figura 2.3: Resultados de SLAM monocular.

El principal problema de este sistema es consecuencia de que, a partir de un modelo de perspectiva estándar de la cámara, se pueden obtener dos de las tres coordenadas de las posiciones relativas de las marcas, pero no es posible obtener directamente la coordenada de profundidad. Esto conlleva dos limitaciones fundamentales:

- En el momento de inicializar una nueva marca, este proceso no puede llevarse a cabo en un sólo paso, ya que no se conoce la tercera coordenada. Para ello, dicha marca se modela como una línea semi-infinita que representa todas las posibles profundidades. Después, por medio de un *Filtro de Partículas*, en cada frame la distribución de probabilidad de la profundidad tiende a concentrarse en su valor final.
- Al comienzo del proceso no existe un conocimiento a priori de la posición/orientación de la cámara, por lo tanto no es posible obtener el valor final de profundidad requerido para las primeras marcas capturadas. Esto conlleva la necesidad de disponer de un cierto número de marcas conocidas a priori, que deberán introducirse manualmente.

En el trabajo [8], desarrollado en el *Centre for Autonomous Systems (Royal Institute)* se describe una alternativa en la representación de marcas para llevar a cabo el proceso de SLAM. Ésta es una representación general para las características de dichas marcas que tiene en cuenta las simetrías y relaciones en las coordenadas de la marca en cuestión. Concretamente, la representación permite añadir las marcas al mapa siendo éstas parcialmente inicializadas. Esta es una propiedad importante cuando se usan marcas orientadas de tipo visual, en las que la información de ángulo puede ser útil antes de conocer sus poses completamente. El número de dimensiones para la característica de una marca puede crecer con el tiempo a medida que se adquiere más información. Del mismo modo que se tienen en cuenta las propiedades particulares de cada tipo de marca, se aprovechan los aspectos comunes de las mismas para permitir el intercambio de los algoritmos de SLAM, así como la elección de sensores y características. Es decir, no es necesario cambiar la implementación de SLAM cuando se cambien sensores o tipos de marcas y viceversa. Para probar el método se implementa un sistema basado en dos tipos de marcas: *líneas* y *puntos*. Las pruebas consisten en utilizar una cámara, enfocada hacia el techo, montada en un robot móvil, equipado además con un sensor de distancia láser. Las marcas de tipo línea se asocian con diversas formas rectas ubicadas en el techo, capturadas con la cámara, así como paredes detectadas con el sensor de distancia, definidas ambas usando la transformada de Hough. Las marcas tipo punto estarían identificadas en este caso con las lámparas del techo (ver Fig. 2.4).

En la Fig. 2.5 se muestran resultados experimentales usando marcas obtenidas con visión.

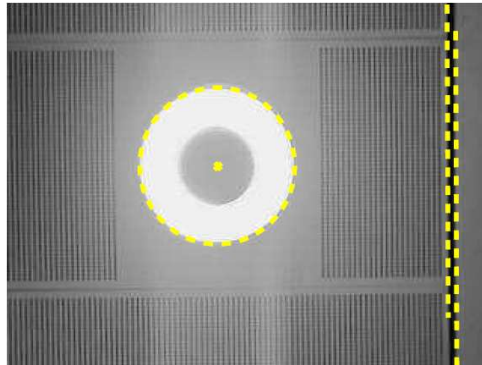


Figura 2.4: Ejemplo de imagen obtenida con la cámara con dos marcas tipo línea y una tipo punto.

En [9], el mismo equipo presenta un método de SLAM visual basado únicamente en información de ángulo de las marcas, dado que sólo se usa una única cámara como sensor. El método se centra en la gestión de las marcas para conseguir un funcionamiento en tiempo real, en su extracción, seguimiento y detección de lazos. Para asociar las marcas del mapa con sus correspondientes características en la imagen, se usa una variante rotacional del descriptor SIFT (*Scale Invariant Feature Transform*, “Transformación de Características Invariante a la Escala”), en combinación con un detector Harris-Laplace [10]. En la Fig. 2.6(a) se muestra el proceso de comparación de características usando los descriptores SIFT mencionados. Para reducir la complejidad de estimación del mapa, manteniendo un buen comportamiento en la detección de correspondencias, sólo se utilizarán unas pocas marcas visuales de gran calidad para formar el mapa. El resto se usarán para las tareas de comparación en la detección de lazos. En cuanto al modelado dinámico, se hace uso de un filtro EKF. En la Fig. 2.6(b) se muestra la reconstrucción del mapa utilizando el método propuesto.

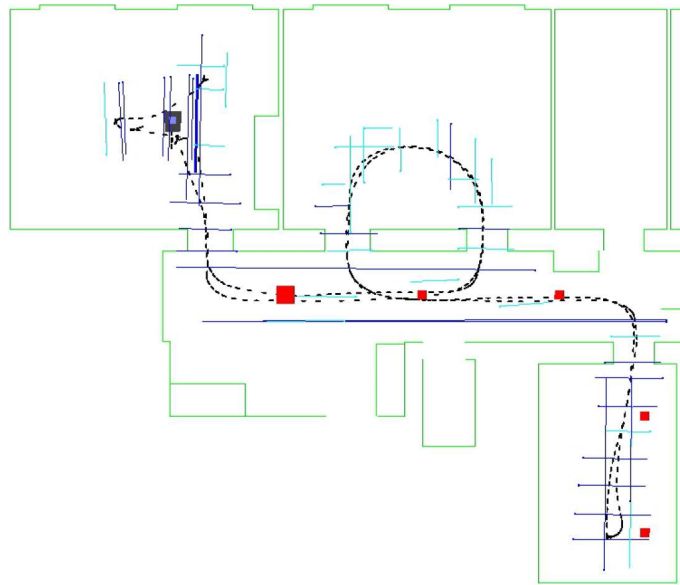
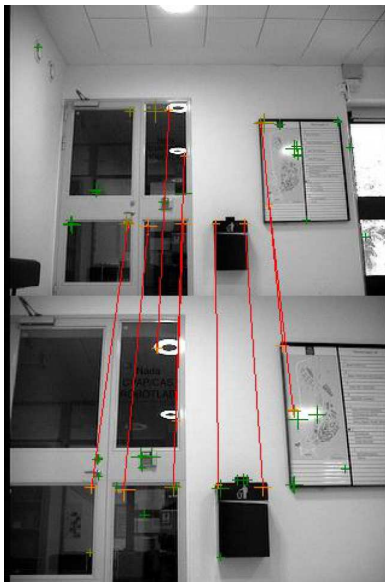
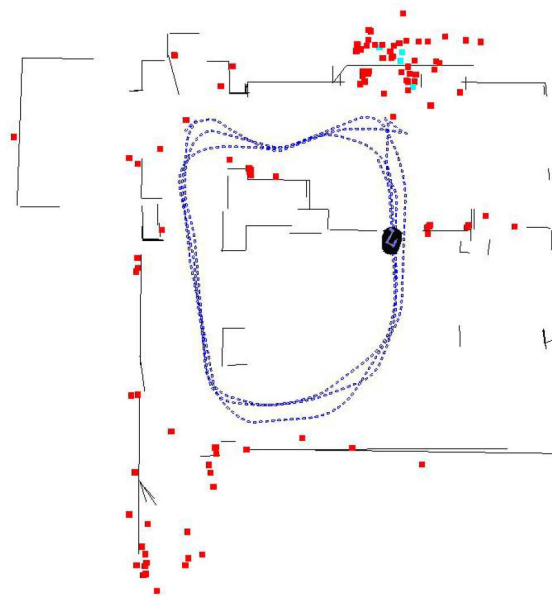


Figura 2.5: Estimación del recorrido del robot junto con las marcas tipo línea (en azul) y las marcas tipo punto (en rojo).



(a) Correspondencias obtenidas usando la variante de descriptores SIFT dependiente de la orientación.



(b) Mapa estimado por el sistema. Las marcas se muestran en rojo.

Figura 2.6: Resultados de SLAM monocular.

En [11] se presenta un sistema para integrar información de bordes en segmentos para implementar un SLAM basado en marcas, con un coste computacional bajo. El sistema también se modela dinámicamente usando un EKF estándar, donde las marcas están constituidas por segmentos, es decir, tramos de pared continua. Estos segmentos pueden o no contener bordes, correspondientes a los finales de las paredes. A medida que se realizan medidas, se integran nuevos segmentos en marcas ya existentes, detectándose si se ha introducido un nuevo borde. A la hora de cerrar lazos, la información adicional de bordes juega un papel importante en su detección. La implementación está basada en el uso de un sensor de distancia láser embarcado en un robot móvil. En la Fig. 2.7 se muestra la estimación de recorrido del robot en un entorno interior. Además se muestra un sencillo ensayo para comprobar el beneficio de usar información de bordes en las marcas a la hora de cerrar un lazo.

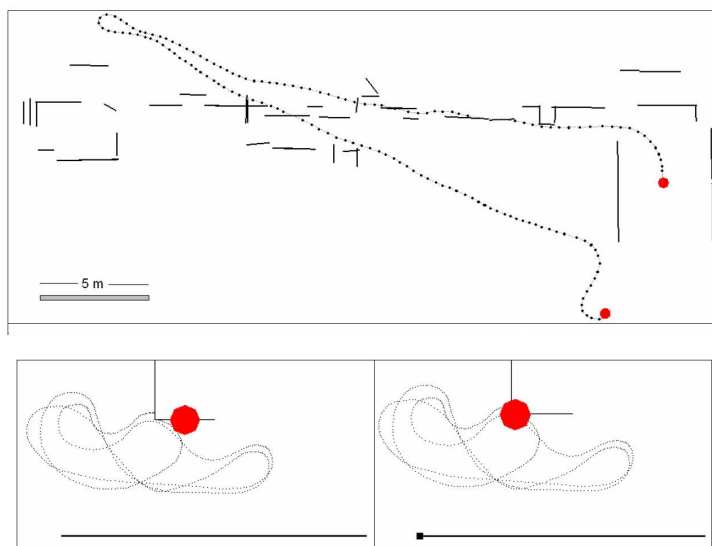


Figura 2.7: En la imagen superior se muestra la estimación de un recorrido realizado por el robot, con una longitud de 30 m y constituido por un mapa de 37 segmentos. En la imagen inferior se muestra un sencillo experimento para comparar la precisión en el retorno del robot al punto de partida, sin usar la información de bordes (izquierda) y usando esta información (derecha)

b) *Sistemas basados en Filtro de Partículas.*

Como es sabido, el EKF supera la limitación de su versión básica para poder ser aplicado en sistemas no lineales. Eso se lleva a cabo linealizando, por ejemplo, la función de próximo estado en torno a un punto de trabajo. A pesar de ello, esta linealización provoca inconsistencias a medio y largo plazo durante la estimación. Una alternativa al uso del EKF es el empleo de un *filtro de partículas* [12] para describir la distribución de probabilidad de la posición de un robot. La idea fundamental de un filtro de partículas es la de que cualquier distribución de probabilidad se puede representar como un conjunto de muestras (partículas) de la misma. Cada partícula tiene asociado, por un lado el vector de estado y por otro lado un determinado peso que se obtiene a través del modelo de observación.

En el trabajo [13] se presenta un método de SLAM basado en el filtro de partículas. El algoritmo, denominado *FastSLAM* trata de estimar recursivamente la distribución de probabilidad total a posteriori de la pose del robot y de las localizaciones de las marcas. La complejidad computacional, sin embargo, crece logarítmicamente con el número de marcas del mapa. Concretamente, el método usa un filtro de partículas modificado para

estimar la distribución a posteriori de los posibles recorridos del robot. Cada partícula contiene un determinado número de filtros de Kalman que se encargan de estimar las localizaciones de las marcas correspondientes, condicionadas al recorrido estimado. En cuanto al coste computacional, éste aumenta significativamente en función del número de partículas empleadas, ya que se multiplica el número de filtros de Kalman implementados. El método se ha probado en entornos simulados y reales, empleando un robot móvil y usando sensores de distancia, tal como se muestra en la Fig. 2.8. En la Fig. 2.9 se muestra un estudio de la precisión del algoritmo en función del número de marcas y del número de partículas. En él se observa que a medida que se usan más marcas el error, tanto en la estimación de la posición del robot como de las marcas, disminuye. Esto también sucede a medida que se aumenta el número de partículas del filtro.

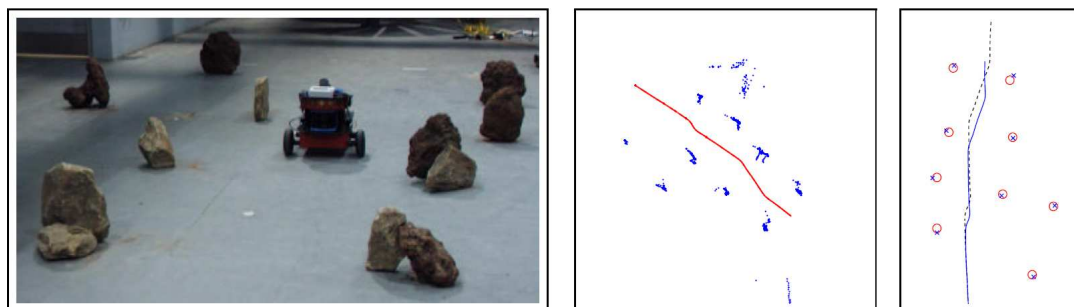


Figura 2.8: En la figura de la izquierda se muestra un robot empleado en las pruebas en un entorno con rocas. En el centro se muestra el recorrido realizado por el robot y las lecturas del sensor de distancia. En la figura de la derecha se muestra la estimación usando FastSLAM (línea punteada) y las localizaciones de las rocas, determinadas manualmente.

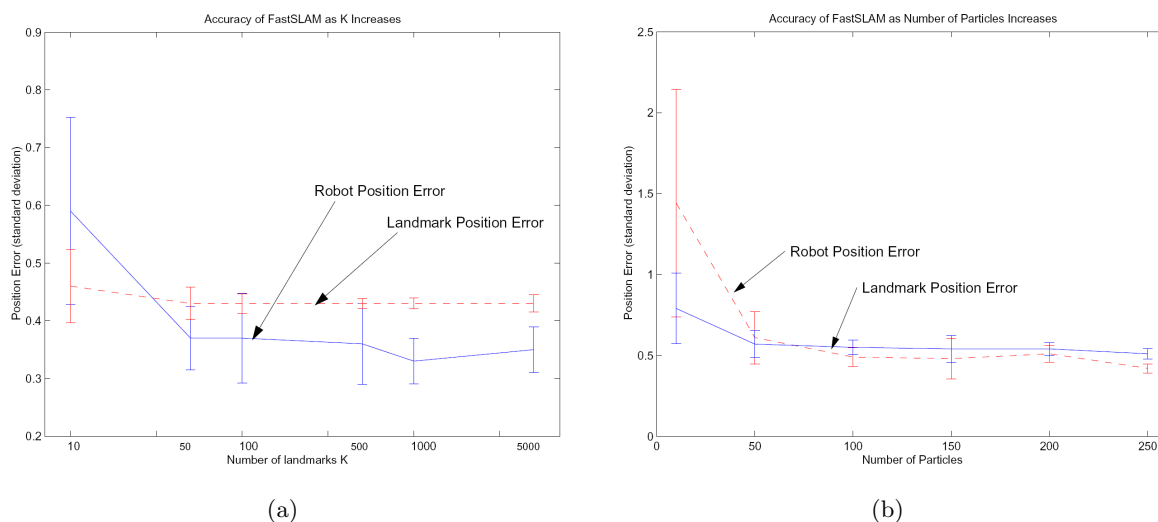


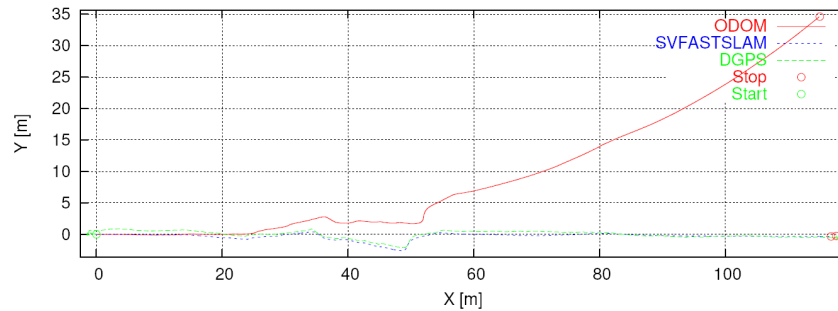
Figura 2.9: Precisión del algoritmo FastSLAM en función del número de marcas N (a) y del número de partículas M (b).

Una alternativa, usando FastSLAM combinado con marcas SIFT es la que se presenta en [14]. En él se describe una técnica para estimar el movimiento 3D de un vehículo usando odometría y una cámara estéreo. Para ello se usan las marcas visuales de tipo SIFT, anteriormente descritas. En este caso, a la hora de realizar la asociación de datos, únicamente se utilizan los correspondientes descriptores SIFT, excluyéndose así la posición 3D

de las marcas. En cuanto al comportamiento dinámico, éste se modela usando una variante del algoritmo FastSLAM presentado anteriormente, denominado *FastSLAM 2.0* [15]. Algunas de las modificaciones implementadas para adaptar el algoritmo al sistema desarrollado son consecuencia del elevado número de marcas visuales observadas simultáneamente (alrededor de 500) y la necesidad de implementar un algoritmo de detección de aquellas que no cumplen con un modelo determinado (*outliers*), debido a la posibilidad de correlaciones incorrectas entre marcas. Por otra parte, debido a la no linealidad del modelo de observación de la cámara empleada, el algoritmo FastSLAM se modifica para adaptar algunos pasos de linealización. A pesar de utilizar un robot móvil con ruedas, el vector de estado de la cámara se define como una pose con seis grados de libertad. En cuanto al modelo de movimiento, para predecir la evolución del estado del robot se hace uso de la odometría incorporada en el mismo. En la Fig. 2.10 se muestra el robot utilizado en el entorno de pruebas. Además, se muestra también una comparativa entre las estimaciones realizadas con el método propuesto, utilizando sólo odometría y la referencia (*ground truth*) realizada con un DGPS.



(a)



(b) Robot en el sub-mapa B1

Figura 2.10: Robot en el entorno de pruebas (a). Estimación del recorrido usando SVFastSLAM y sólo odometría, comparado con la referencia obtenida con un DGPS (b).

Una alternativa adicional en el uso del método FastSLAM se describe en el trabajo [16]. Éste presenta un sistema de SLAM basado en visión, el cual usa de nuevo marcas tipo SIFT, aplicando una técnica de selección para centrarse en un grupo reducido de características distinguibles y estables. Estas marcas SIFT son seguidas a lo largo de frames consecutivos, obtenidos con la cámara estéreo. De estas marcas sólo se seleccionan aquellas que resulten suficientemente estables desde diferentes puntos de vista. Cada vez que se selecciona una marca, esta se observa durante varios frames consecutivos, obteniendo diferentes puntos de vista con sus correspondientes descriptores SIFT asociados. El comportamiento dinámico del sistema se modela utilizando un filtro de partículas de tipo *Rao-Blackwellized*, es decir el método FastSLAM. En las pruebas realizadas se tomaron 1500 marcas y se probaron

implementaciones del método FastSLAM usando entre 10 y 100 partículas. Los ensayos se realizaron con un robot móvil en interiores, a lo largo de un recorrido de 80m de longitud. En la Fig. 2.11 se muestra la estimación del recorrido realizado por el robot usando 10 partículas y usando 100 partículas.

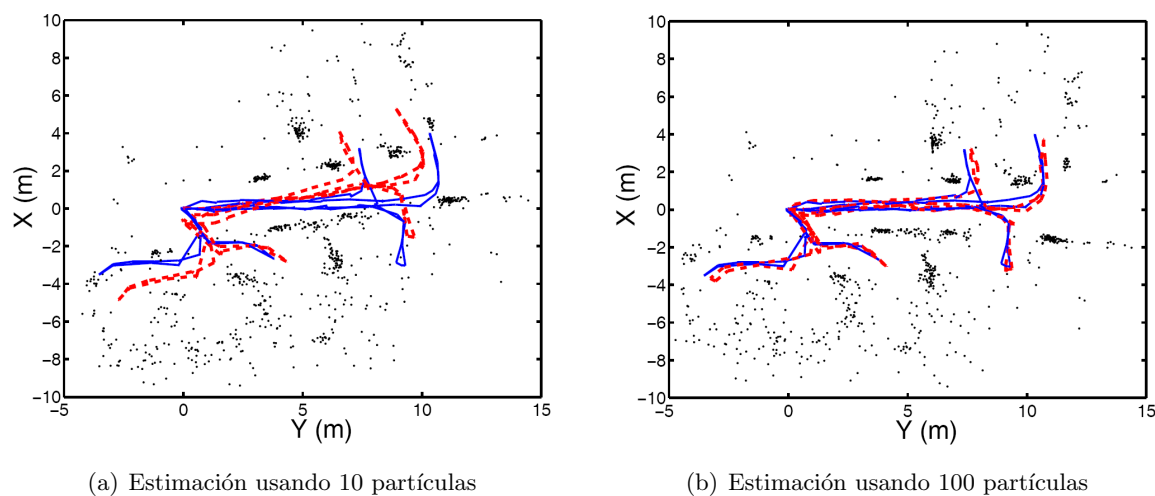


Figura 2.11: Estimación del recorrido usando el método presentado (línea rayada) comparada con el recorrido real (línea continua).

c) *Sistemas basados en marcas SIFT.*

Uno de los principales problemas a resolver en los sistemas de SLAM basado en marcas, es la correcta asociación de éstas con sus correspondientes características. Éstas deben ser identificables en la mayor cantidad de situaciones posibles y, además, deben ser distinguibles unas de otras con la mayor probabilidad posible. Existen en el estado del arte numerosos estudios focalizados en este aspecto.

En el trabajo [17], David G. Lowe presenta un método de obtención de características de marcas visuales, las cuales se caracterizan por seguir siendo identificables y distinguibles en situaciones de rotaciones, translaciones, cambios de escala y, parcialmente, en cambios de iluminación y distorsiones afines. Estas marcas se denominan como SIFT (*Scale-Invariant Feature Transform*). El enfoque de este trabajo se centra en el desarrollo de un sistema de reconocimiento de objetos, usando este tipo de marcas visuales. Estas marcas se detectan eficientemente a través de un método de filtrado en etapas que se encarga de identificar puntos estables en el denominado *espacio de escala*. Se crean la denominadas *claves* de imagen, que tienen en cuenta las deformaciones geométricas locales, representando los gradientes de la imagen *suavizada* en múltiples orientaciones y escalas. Estas claves se usan como entrada a un método de indexado de tipo *nearest-neighbor* que identifica posibles candidatos de objetos reconocidos. Finalmente se realiza una verificación final en cada candidato, buscando una solución por mínimos cuadrados para los parámetros desconocidos del modelo. Se muestran resultados experimentales en los que se reconocen objetos en condiciones de oclusión parcial y sobre fondos muy ricos en textura, alcanzándose tiempos de cómputo por debajo de los dos segundos (ver Fig. 2.12).

Posteriormente, autores del mismo grupo de trabajo han desarrollado la aplicación de este tipo de marcas a la localización y mapeado de un robot móvil [18]. La invarianza de estas marcas a la translación, escalado y rotación las hace apropiadas para tal efecto. En este trabajo se usa un sistema trinocular denominado Triclops (ver Fig. 2.13(a)). Con él se

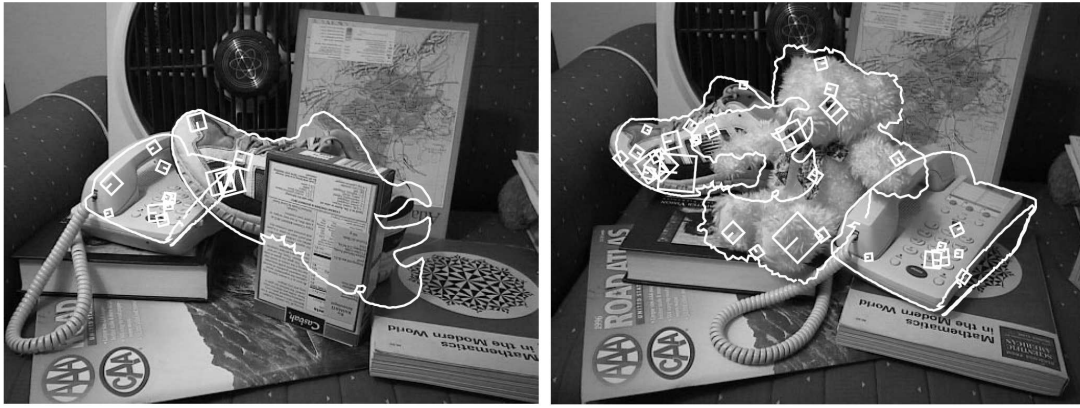
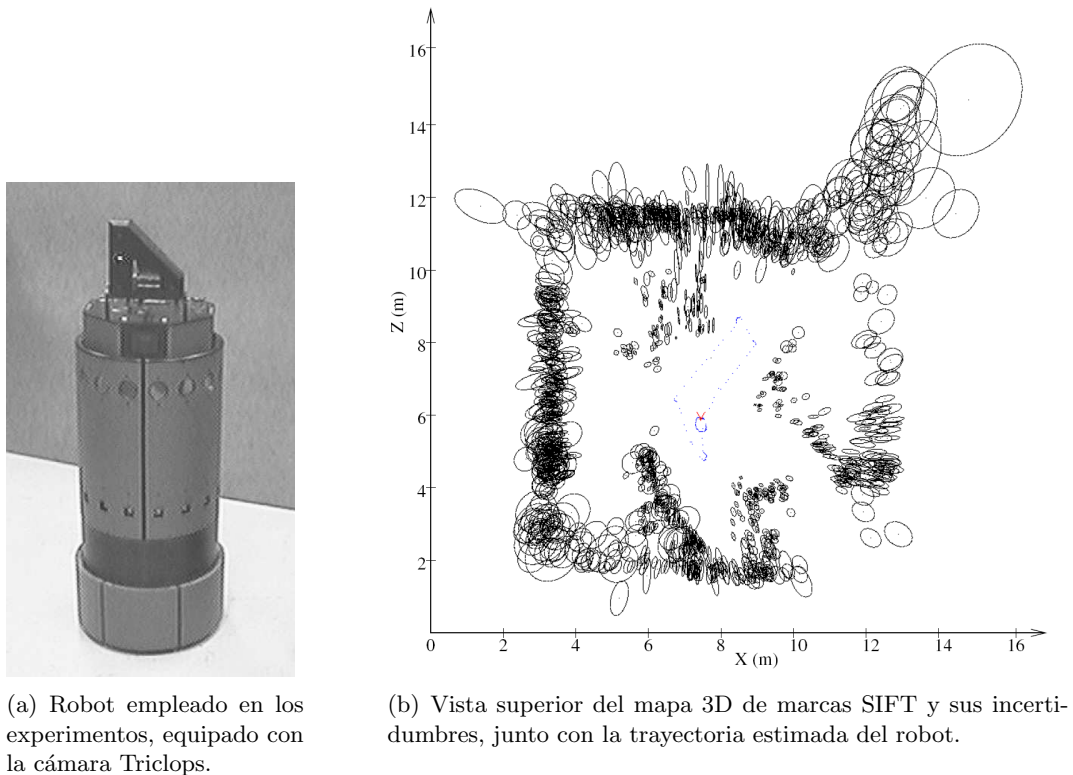


Figura 2.12: Ejemplo de reconocimiento de objetos tridimensionales con oclusiones.

localizan este tipo de marcas y se estima el movimiento propio del robot, haciendo uso de la minimización por mínimos cuadrados de las marcas identificadas. Para tener en cuenta las variaciones en los puntos de vista de las marcas, así como posibles oclusiones, se almacena la información de la dirección de captura de cada marca. Para estimar la posición de las marcas se realiza un proceso de comparación (*matching*) entre las imágenes obtenidas por la cámara izquierda y la derecha. Para ello se tienen en cuenta las restricciones epipolar, de disparidad y de unicidad, añadiendo además las propias de las marcas SIFT, es decir, las restricciones de orientación y escala. Posteriormente se buscan las correspondencias restantes con las proyecciones en la imagen de la cámara superior, siguiendo los mismos criterios anteriormente descritos.



(a) Robot empleado en los experimentos, equipado con la cámara Triclops.

(b) Vista superior del mapa 3D de marcas SIFT y sus incertidumbres, junto con la trayectoria estimada del robot.

Figura 2.13: SIFT aplicado a la navegación de un robot.

El sistema se ha implementado en una plataforma robótica móvil, de la cual se extraen las medidas de odometría para predecir el próximo estado. En la etapa de medida se utilizan las marcas SIFT para corregir la estimación. En la figura 2.13(b) se muestra la trayectoria estimada del robot junto a las marcas SIFT y sus covarianzas. Una limitación asociada a las marcas SIFT es un tiempo de cómputo relativamente alto a la hora de extraer los descriptores de características. Los resultados muestran un funcionamiento del sistema a una frecuencia de 2Hz, alejándolo del tiempo real.

Otro enfoque a la hora de usar marcas SIFT para la navegación de robots se presenta en el trabajo [19]. En este artículo se propone una alternativa basada en el empleo de la memoria adaptativa para el control y aprendizaje en robots. La alternativa se basa en la premisa de que el razonamiento espacial cualitativo es apropiado para solventar los problemas de cálculo de pose, aprendizaje y tareas de navegación. Más concretamente, la idea propuesta consiste en adquirir marcas puntuales potenciales para, posteriormente, asociarlas en grupos de forma que sean utilizadas para representar objetos “de interés” que sean mantenidos a lo largo del recorrido. Estos grupos deben ser consistentes, de forma que puedan servir como partes de la memoria del sistema. Las marcas utilizadas para formar estos grupos son también de tipo SIFT. Para asociar las marcas entre sí, el procedimiento consiste en una dilatación morfológica y posterior etiquetado. Este método es sencillo y permite unos tiempos de cómputo muy bajos. Las pruebas se realizaron llevando una cámara en la mano por una zona de campo abierto, tal como se muestra en la Fig. 2.14. El máximo número de agrupaciones mantenido en un sólo frame fue de 10, con una media de 300 marcas en cada grupo, aproximadamente.



Figura 2.14: Agrupaciones de marcas obtenidas y mantenidas en una secuencia en exteriores.

Además de las alternativas presentadas, existen otros enfoques diferentes en el uso del método SIFT, como el presentado anteriormente en [9]. Otro trabajo que utiliza marcas SIFT es el que se muestra en [20], en el que se hace uso de su capacidad para representar la apariencia de colecciones de imágenes. Asimismo, en [21] también se emplean marcas SIFT para identificar lugares previsitados con una determinada probabilidad y poder así cerrar lazos. Ambos trabajos son evaluados más adelante debido a otras características más representativas sobre su implementación.

2.1.2. Sistemas SLAM para grandes entornos

A la hora de extender la aplicación del EKF hacia entornos de mayores dimensiones, se demuestra que el tamaño de la matriz de covarianza total se incrementa ilimitadamente en función del tamaño de dichos entornos. La complejidad en el cálculo de esta matriz aumenta cuadráticamente en función del número de marcas n introducidas en el filtro, es decir, la complejidad es de orden $O(n^2)$.

Para solucionarlo, algunas alternativas se basan en modificar los principios intrínsecos del EKF, en cuanto a la manera de calcular la matriz de covarianza [22] [23]. La mayoría de ellos consiguen reducir el problema a un orden de complejidad lineal $O(n)$. Otra alternativa consiste en dividir el mapa global en otros más pequeños usando diversas jerarquías que serán explicadas a continuación. Por último, se presentarán algunos trabajos que centran el problema en obtener únicamente la trayectoria del vehículo, sin construir ningún tipo de mapa basado en marcas.

a) *Métodos jerárquicos de tipo Métrico-Métrico.*

Una primera solución al problema de los grandes entornos, desde el punto de vista jerárquico, es aplicar el denominado enfoque *Métrico-Métrico*. Éste aborda el mencionado problema dividiendo el mapa completo en otros más pequeños, aplicando un nivel métrico superior sobre los sub-mapas métricos resultantes.

Uno de los primeros métodos que aplicaron técnicas de división del mapa se presenta en el trabajo de J. D. Tardós et al [24]. La idea de este sistema es la de crear un mapa y localizar un robot dentro de él, haciendo uso de medidas de s3nar. Estas medidas dan como resultado unas marcas que se dividen en dos tipos de características, bien de tipo *línea* o bien de tipo *punto*. Una de las contribuciones más importantes fue la creación de sub-mapas locales y aplicar filtros EKF dentro de ellos. Los sub-mapas independientes se unen posteriormente usando las denominadas *composiciones*. En la figura 2.15 se muestra en primer lugar la representación de un robot tomando una serie de lecturas de sonar que, denotan marcas tanto de tipo línea como de tipo punto. En segundo lugar se muestra la representación del robot en un primer sub-mapa y posteriormente en otro sub-mapa. Un inconveniente importante de este método, así como del SLAM visual jerárquico, que se explicará más adelante, es el hecho de que los mapas locales deben ser estadísticamente independientes. Esto impide que se comparta información valiosa entre éstos.

Una solución al problema del método anteriormente explicado ha sido recientemente publicada por el mismo grupo de investigación en [25]. En éste se describe una técnica, desarrollada en el espacio de la matriz de covarianza, que permite que los mapas locales correlados entre sí puedan compartir información de una manera consistente, tal como se muestra en la Fig. 2.16. Este método está bien adaptado a aplicaciones con sensores que requieren tiempos de inicialización de marcas muy largos, como es el caso de los sistemas de visión monocular presentados anteriormente. Esto es debido a que, para estimar correctamente una marca puede requerirse su medida desde varias poses del robot. También sería apropiado para sensores inerciales (IMU), donde la velocidad y pose de dicha IMU en el sub-mapa actual depende de sus valores en los sub-mapas anteriores.

Con el objeto de extender el método de SLAM monocular (*MonoSLAM*) hacia entornos de mayores dimensiones, en [26] se presenta un trabajo de SLAM visual jerárquico (*Hierarchical Visual SLAM*). Este sistema, asimismo, hace uso de una única cámara y, por lo tanto acusa el mismo problema de imposibilidad de estimar la escala absoluta del mapa. Como consecuencia, la escala debe fijarse observando objetos conocidos, de forma que se evite la deriva de la propia escala con el tiempo. Este método puede usarse para el mapeado de

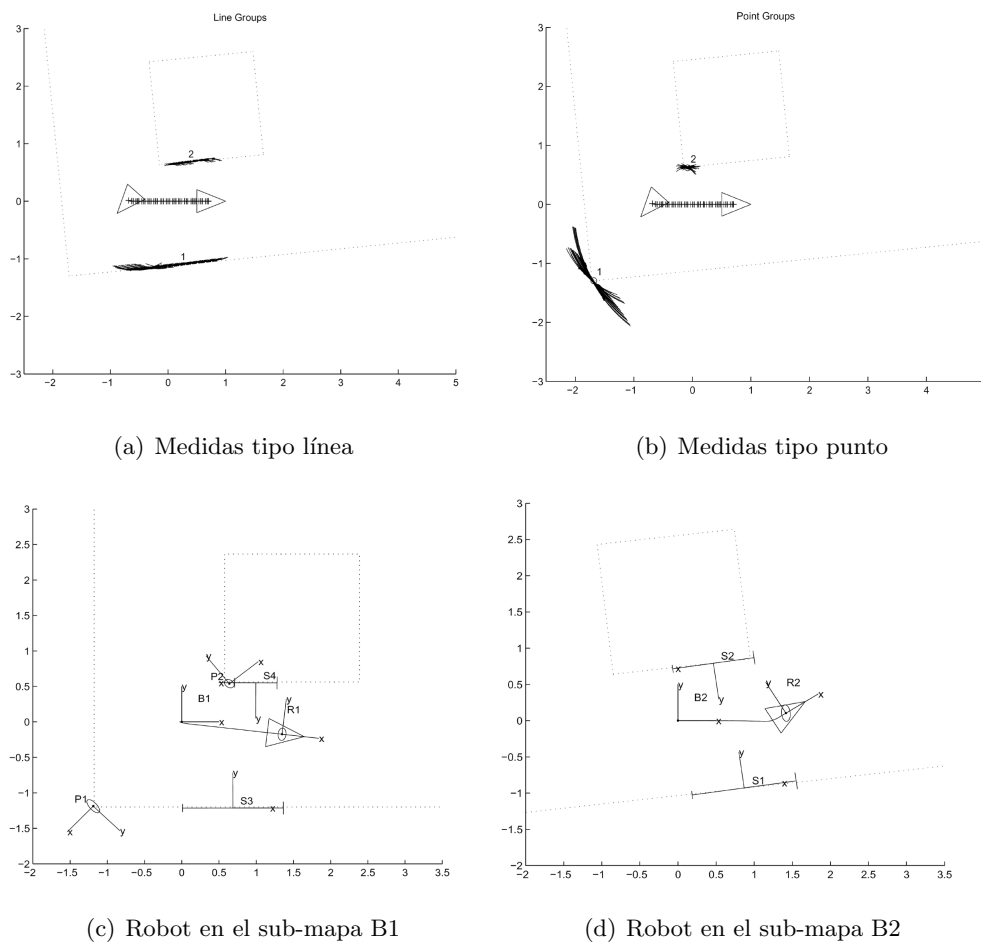


Figura 2.15: Resultado de las medidas de sonar para marcas tipo línea (a) y tipo punto (b). Representación del robot en dos sub-mapas diferentes (c) y (d), compuestos por marcas tipo línea Sx y tipo punto Px

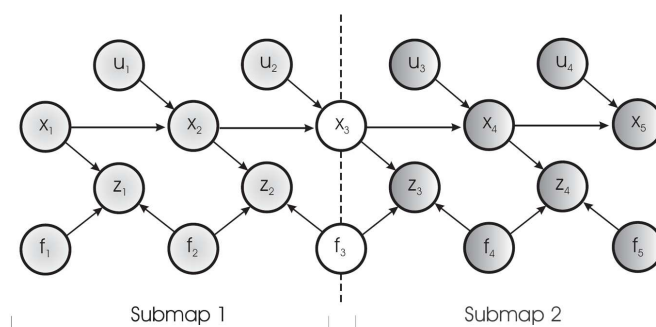


Figura 2.16: Red Bayesiana representando dos sub-mapas independientes y sus dependencias probabilísticas. Las variables u_i representan las acciones introducidas al sistema, x_i las diferentes poses del robot, f_i las marcas de cada sub-mapa y z_i las observaciones realizadas

grandes entornos, dividiendo el mapa global en sub-mapas locales de tamaño limitado y consiguiendo, por tanto, tiempos de ejecución casi constantes. Un ejemplo de una situación de cierre de lazo, empleando esta técnica se muestra en la Fig. 2.17.

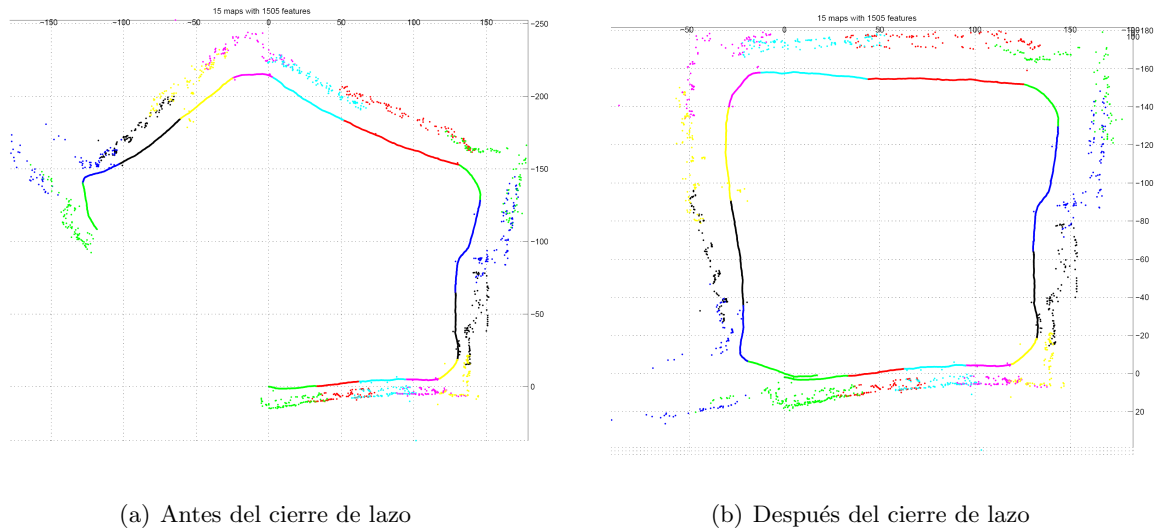


Figura 2.17: Representación del mapa global antes y después de una situación de cierre de lazo. Cada submapa se muestra de un color diferente. Las marcas obtenidas se muestran como puntos.

Una de las últimas contribuciones de este grupo al estado del arte, es el trabajo presentado en [27]. En éste caso, se usa un sistema estéreo de seis grados de libertad, que puede moverse con la mano, para capturar marcas visuales. El sistema está basado en la cámara estéreo comercial *Bumblebee* [28]. En la Fig. 2.18 se muestran las dos imágenes capturadas por la cámara, así como las proyecciones de las marcas y sus correspondientes covarianzas. Al tomarse la cámara derecha como sistema de referencia, se aprecia que la proyección de las covarianzas antes mencionadas, es más alargada en la imagen izquierda. La particularidad de este sistema es que estas marcas pueden clasificarse, bien como *cercanas* o bien como *lejanas*. Dependiendo de esta clasificación, la información proporcionada por el par estéreo será, bien la localización completa de la marca en cuestión, o bien simplemente la información del ángulo de dicha marca relativo a la cámara. Esta metodología es una evolución de una versión monocular previa desarrollada por el mismo grupo de investigación en [25]. La estrategia aplicada aquí es también el uso de sub-mapas basados en EKF. Los resultados muestran un mapeado preciso, así como la capacidad de cerrar lazos en entornos relativamente grandes, tal como se muestra en la Fig. 2.18. Sin embargo, debido al movimiento lateral de la cámara asumido para el sistema, se impone una filosofía de *matching* continuo, de cara a reducir el número de frames necesarios para detectar situaciones de cierre de lazos. El uso de un sistema de cámaras con un rango relativamente corto, lo hace inapropiado para entornos muy grandes o en espacios muy abiertos en los que la mayoría de las marcas estarán situadas muy lejos de la cámara. Por otra parte, el comportamiento en tiempo real no se llega a conseguir completamente.

b) *Métodos jerárquicos de tipo Topológico-Métrico.*

Otra alternativa para solucionar el problema de los entornos de grandes dimensiones, consiste en aplicar un nivel topológico superior sobre los sub-mapas métricos. Esta alternativa se denomina enfoque *Topológico Métrico*.

En [29] se presenta el método denominado *Decoupled Stochastic Mapping* (DSM), en el cual

el mapa global se divide en *celdas* más pequeñas conteniendo partes del mencionado mapa global. Todas las marcas, así como las poses del vehículo están referenciadas al sistema de referencia global en cualquiera de las celdas. Pasar de una celda a otra implica una transferencia de información, que será resuelta por métodos de *inflación* de incertidumbre, los cuales son cuestionables. Por otro lado, el problema de la optimización del cierre de lazos no se tiene en cuenta en este trabajo. El sistema se ha implementado en un robot acuático equipado con sonar. En la Fig. 2.19 se muestra el recorrido a realizar por el robot, así como los sub-mapas obtenidos por el método.

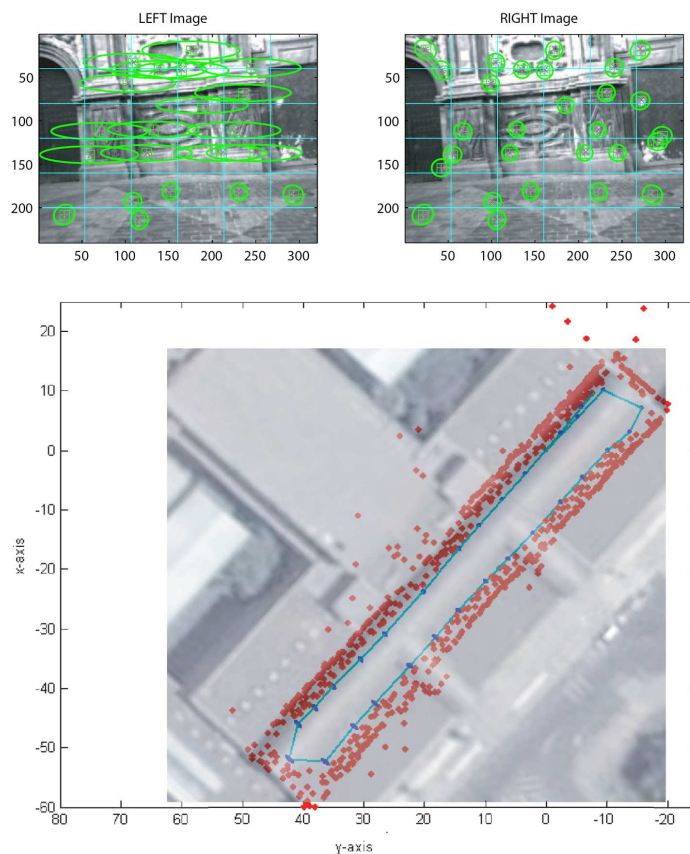
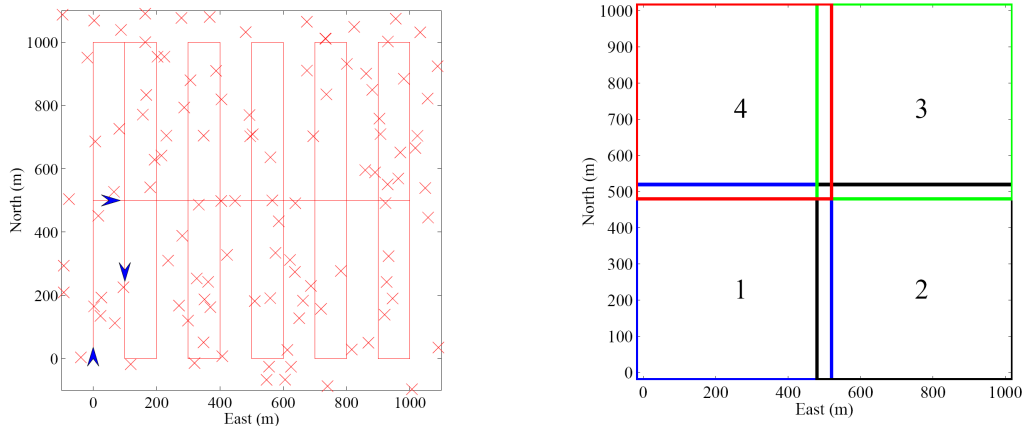


Figura 2.18: *Imágenes superiores*: Imágenes capturadas por la cámara junto a las proyecciones de las marcas y sus correspondientes covarianzas. *Imagen inferior*: Reconstrucción de la trayectoria del móvil junto con las marcas obtenidas en el entorno de prueba.

En [30] se presenta el método *Hierarchical Local Maps* (HLM). Éste consiste en un conjunto jerárquico de sub-mapas, referenciados localmente en este caso. Cada vez que se añade un nuevo sub-mapa implica el almacenamiento de la estimación de la pose local del vehículo y su covarianza en ese momento. Todas las estimaciones se almacenan en un *árbol de acoplamiento*, en el que las relaciones entre cualquiera de los sub-mapas se puede calcular usando las fórmulas denominadas *coupling summation*. A nivel físico, el sistema se basa en un robot móvil equipado con un *array* de sensores sonar, tal como se muestra en la Fig. 2.20. Uno de los principales inconvenientes está en el hecho de que las estimaciones acopladas de todos los sub-mapas permanecen estáticas a lo largo de todo proceso. Esto implica que no se podrá realizar una reducción de la incertidumbre tras el cierre de lazos.

El trabajo [31] presenta el método denominado *Constrained Relative Submap Filter* (CRSF). Éste es esencialmente igual al HLM, previamente explicado, pero introduce mejoras en la



(a) Camino a recorrer por el robot junto con las diferentes marcas del entorno aleatoriamente distribuidas

(b) Particionado del mapa en los diferentes submapas

Figura 2.19: Representación del mapa y el recorrido realizado por el robot.

forma en la que las estimaciones acopladas son almacenadas. Esto permite reinicializar la estimación del vehículo usando restricciones geométricas, en caso de que el vehículo retorne al sub-mapa anterior. Además, permite la reducción de la incertidumbre del siguiente sub-mapa, dado que el anterior ya había convergido también. Sin embargo, debido a los enlaces monótonos entre sub-mapas, no es posible realizar una optimización global en caso de situaciones de cierre de lazos. Experimentalmente, el método se probó con un vehículo submarino equipado con diversos sensores: sonar, giróscopos, compases, barómetros y una cámara. Para el experimento se hizo uso únicamente del sonar, donde las marcas del mapa son unas balizas artificiales suspendidas en el fondo del mar. En la Fig. 2.21 se muestra una imagen del vehículo empleado, así como una representación del mapa estimado y la trayectoria del móvil.

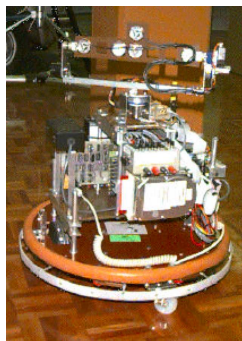
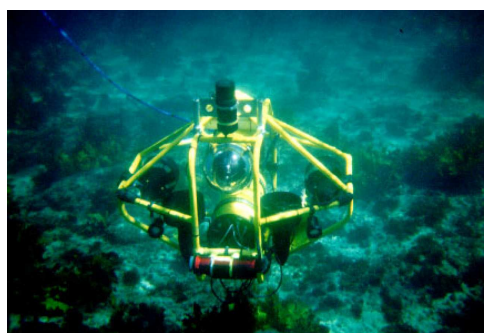
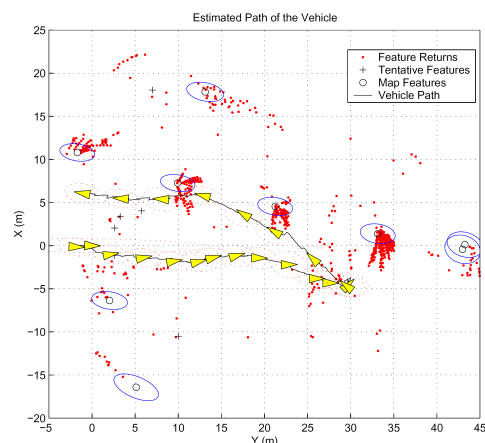


Figura 2.20: Robot utilizado para probar el método HLM, equipado con un *array* de sensores sonar en la parte superior

El método denominado *Network Coupled Feature Maps* (NCFM) se presenta en [32]. Éste está basado también en el CRSF, sin embargo, el NCFM no restringe las estimaciones acopladas a enlaces monótonos, permitiendo una mejor optimización en situaciones de cierre de lazos. Las pruebas realizadas se llevaron a cabo con un vehículo dotado de sensores de distancia láser. En la Fig. 2.22 se muestra el vehículo utilizado para las pruebas, así como la estimación realizada de uno de los recorridos de ensayo. Una ventaja del método es que permite optimizar los acoplamientos de diferentes sub-mapas cuando el vehículo



(a) Vehículo de pruebas



(b) Mapa y recorrido del vehículo estimados

Figura 2.21: Vehículo de pruebas y mapa estimado por el sistema. Las poses del vehículo se representan en amarillo. Las medidas del sonar aparecen como puntos rojos, siendo la estimación final de las marcas las señaladas como círculos rodeados por su incertidumbre asociada.

recorre las zonas comunes entre esos sub-mapas. Por otro lado, esto implica tener una red relativamente densa de sub-mapas fuertemente solapados para aprovechar esta ventaja y poder reducir la incertidumbre global. Además, el NCFM requiere un método robusto de asociación de datos para poder relacionar las marcas visuales entre sub-mapas adyacentes, en caso de ser aplicado en sistemas de SLAM visual.

El *Atlas Framework* [33] se basa, asimismo, en grafos de sistemas de referencia locales. Cada nodo en el grafo, por tanto, representa un sistema local y cada transición representa la transformación entre los sistemas locales adyacentes. En cada sistema de referencia se construye un sub-mapa que engloba el entorno local y la pose actual del robot junto con las incertidumbres de cada uno. Las incertidumbres de cada uno de los mapas se modelan con respecto a su propia referencia. Por otro lado, se pueden calcular las incertidumbres relativas de cualquier nodo respecto a cualquier sistema de referencia siguiendo el camino más corto, formado por las transiciones entre nodos de dicho camino. Este camino se calcula usando el algoritmo de Dijkstra para tal efecto. El cierre de lazo se consigue usando un algoritmo eficiente de comparación (*matching*) entre sub-mapas. Los resultados muestran un funcionamiento en tiempo real en un entorno estructurado en interiores de grandes dimensiones (aproximadamente 2.2 Km de longitud) incluyendo múltiples lazos anidados. El sistema se basa en el uso de sensores de distancia láser y de ultrasonidos. Un ejemplo de un cierre de lazo en un entorno aproximadamente rectangular se muestra en la Fig. 2.23. Como inconveniente se puede mencionar el hecho de que el sistema no realiza una optimización en el mapa global cada vez que se actualiza uno de los nodos.

En el trabajo [34], Udo Frese presenta el algoritmo denominado *TreeMap*. La idea es construir un mapa jerárquico compuesto de varios niveles. Las medidas están basadas en marcas. La idea fundamental, cuando el vehículo se encuentra en un sub-mapa, es manejar sólo la información necesaria acerca de otros sub-mapas, como las distribuciones de las marcas compartidas con éstos, condicionadas a las observaciones llevadas a cabo en sus correspondientes sub-mapas. Las dependencias entre sub-mapas se organizan como un mapa en árbol (*Treemap*), de tal forma que la integración de nuevas medidas implica únicamente la actualización del sub-mapa actual y de todos los de un nivel superior que pertenezcan a la misma rama. En cada nivel de la jerarquía, cada región almacena

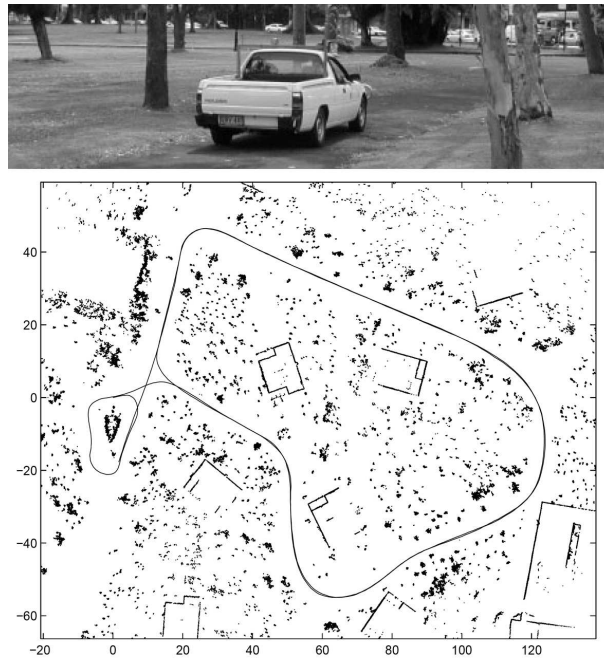


Figura 2.22: En la imagen superior se muestra el vehículo utilizado para las pruebas. En la imagen inferior se muestra la reconstrucción del recorrido realizado por el vehículo, así como las medidas realizadas con el sensor láser

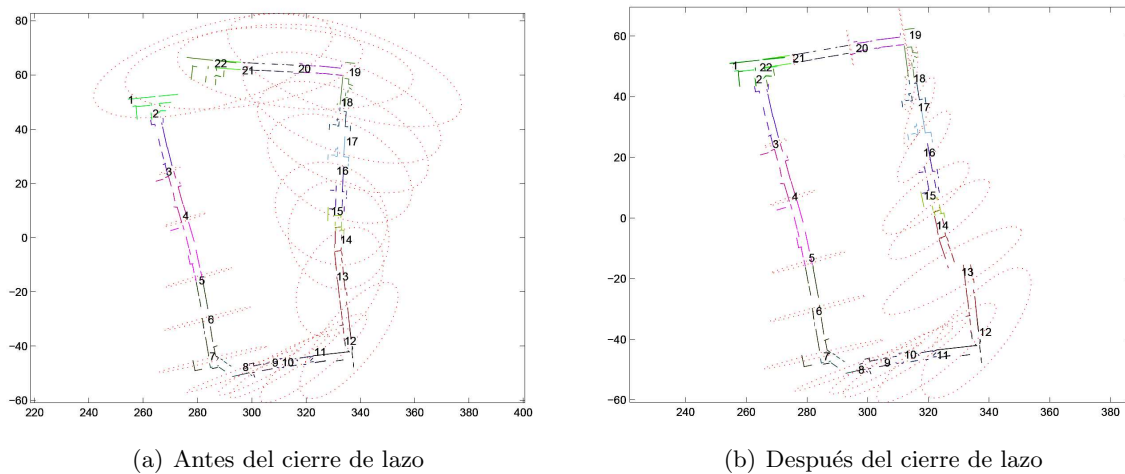


Figura 2.23: Representación del mapa global dividido en diversos sub-mapas. Se muestran también las covarianzas de incertidumbre de los sub-mapas (nodos) relativas al sub-mapa número 1, indicado en la parte superior izquierda.

una matriz representando algunas de las marcas contenidas en dicha región. Para mantener esas matrices con un tamaño reducido, sólo se representan aquellas marcas que son observables desde fuera de la mencionada región. Esta optimización marginal se realiza sin aproximaciones, de forma que la incertidumbre permanece reducida. En la Fig. 2.24 se muestra un entorno de ejemplo que ha sido dividido en dos niveles jerárquicos: L1 y L2. También se muestra la división en sub-mapas y sus nodos asociados en cada uno de los niveles. Como consecuencia de la implementación eficiente, se demuestran mejoras en cuanto al tiempo de procesamiento comparado con el EKF, FastSLAM y otros conocidos métodos, consiguiéndose una dependencia lineal con el número de marcas. En la tabla 2.1 se muestra una comparativa del método *treemap* frente a otros métodos muy extendidos, donde n representa el número de marcas, m el de medidas, p el número de poses del robot y k las marcas locales al robot.

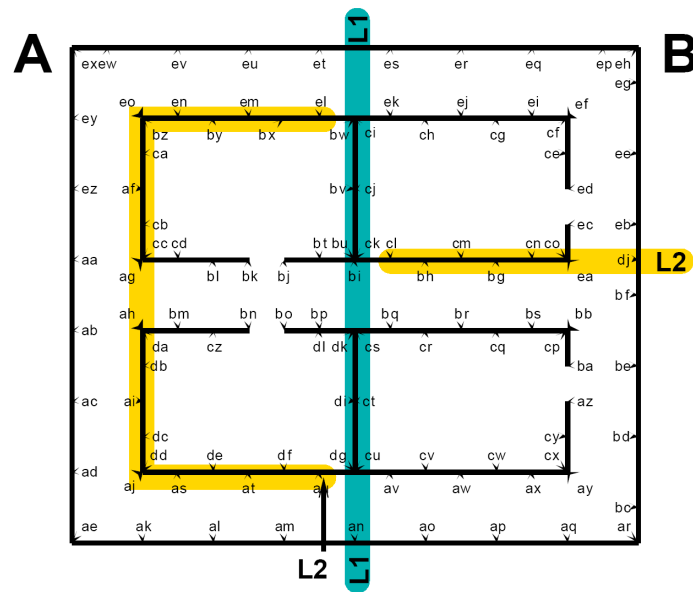
	memory	update	global update	loop
ML	m	$\dots\dots(n+p)^3\dots\dots$		
EKF	n^2	$\dots\dots n^2\dots\dots$		
CEKF	$n^{\frac{3}{2}}$	k^2	$\dots kn^{\frac{3}{2}}\dots$	
Relaxation	kn	$\dots kn\dots$	kn^2	
MLR	kn	$\dots kn\dots$		
FastSLAM	Mn	$\dots M \log n\dots$		
SEIF	kn	$\dots k^2\dots$		
w. full update	kn	$\dots kn\dots$	kn^2	
TJTF	k^2n	k^3	$\dots k^3n\dots$	
Treemap	kn	k^2	$\dots k^3 \log n\dots$	
w. global map	kn	$\dots kn\dots$		

Tabla 2.1: Rendimiento de diferentes algoritmos de SLAM frente a *treemap*.

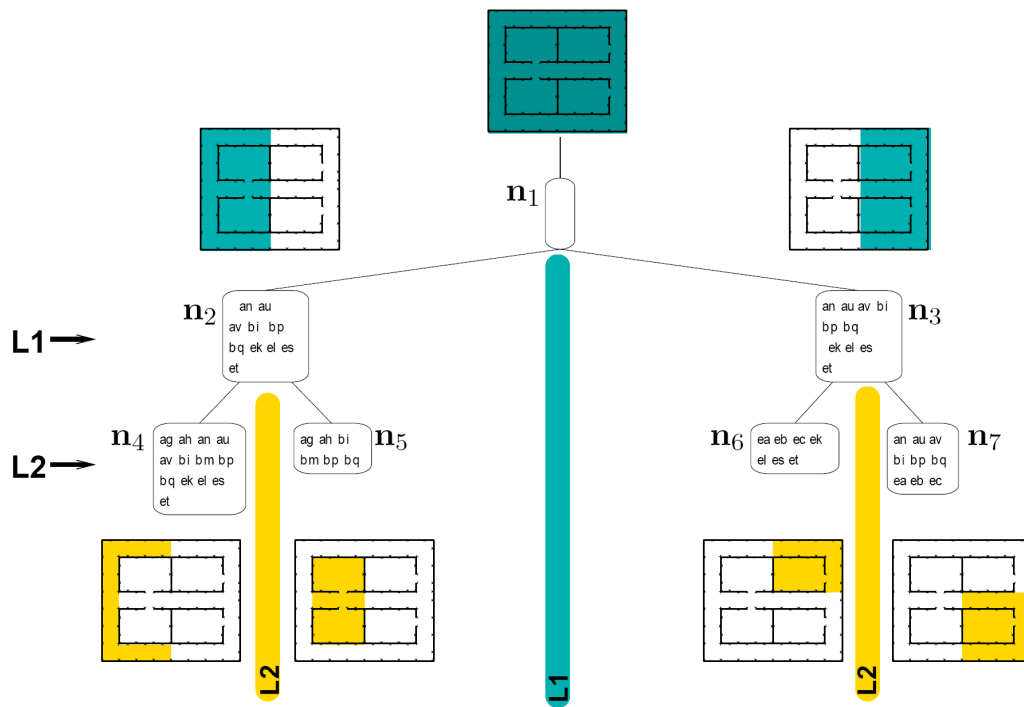
En términos de eficiencia computacional, en [35] los mismos autores presentan una mejora del método *Treemap* capaz de cerrar un lazo conteniendo alrededor de un millón de marcas, en un entorno simulado, empleando unos cientos de milisegundos. La mejora se basa en el uso de factores *Cholesky* para representar a las distribuciones de probabilidad gaussianas y del algoritmo HTP (*Hierarchical Tree Partitioning*, “Particionado jerárquico del árbol”). Este algoritmo se ejecuta en paralelo, comprobando continuamente la distribución óptima del árbol. El HTP, por tanto, se encargará de agrupar o dividir ramas del árbol, de forma que no haya demasiadas o demasiado pocas marcas en cada nodo.

Como inconveniente del método *Treemap* y sus derivados, se puede mencionar que el problema de asociación de datos se asume ya resuelto. Este problema es uno de los más importantes a tener en cuenta en entornos exteriores usando marcas naturales. Esto es crucial de cara a conocer qué sub-mapa está asociado a cada marca. Por otro lado, la compleja estructura topológica del algoritmo lo hace apropiado para ciertos tipos de edificaciones, donde las ventajas del proceso usando múltiples niveles jerárquicos puede ser aprovechada.

La alternativa presentada en [36] consiste en un SLAM visual monocular basado en el método NCFM. Éste consiste en un conjunto de nodos interconectados, conteniendo sub-mapas estimados usando el filtro de Kalman. Las incertidumbres y estados de los sub-mapas se



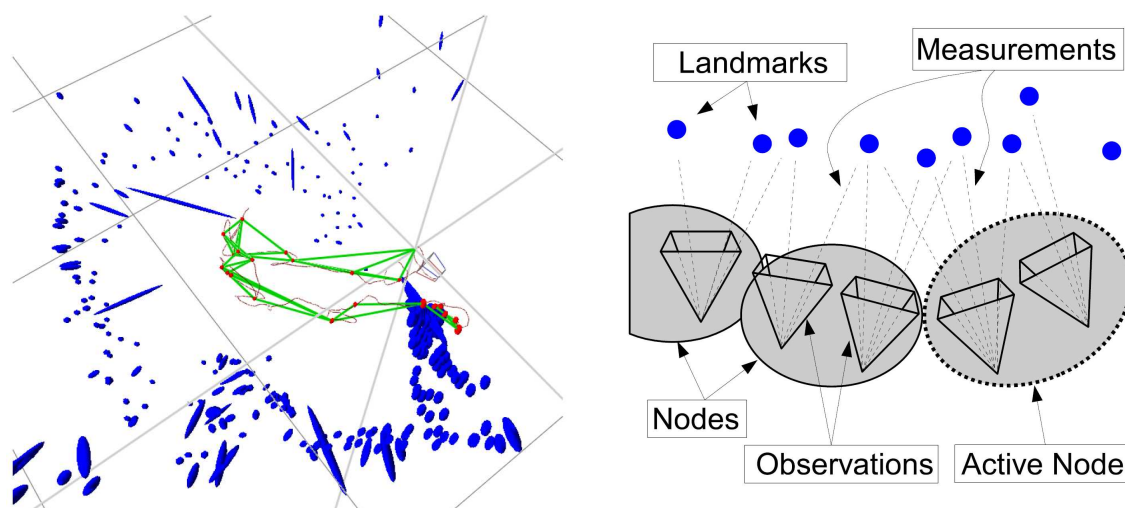
(a) Edificio dividido jerárquicamente en dos niveles (L1, L2). Las marcas artificiales ubicadas se muestran como: aa, ab, ac, etc.



(b) Representación asociada en forma de árbol, con 7 nodos ($n_{1..7}$). La región correspondiente a un nodo se muestra junto a dicho nodo. Las marcas dentro de una región que sean visibles desde fuera, se listan en el nodo asociado. Sólo serán necesarias las distribuciones marginales de estas marcas si el robot se encuentra fuera de la región asociada a ellas.

Figura 2.24: Representación geométrica del algoritmo.

calculan en sus propios sistemas de referencia locales. Para reducir los errores de linealización, las medidas se expresan usando la representación de profundidad inversa (*inverse depth*). Las transiciones del grafo almacenan las relaciones entre nodos, definidas por una transformación de *similaridad*. Esta transformación está constituida por las matrices de rotación y translación correspondientes, añadiendo la información de escala, la cual es necesaria debido a la implementación monocular. El nodo activo se selecciona basándose en las marcas visibles disponibles y en la estimación de linealidad del modelo de medida. En la Fig. 2.25 (a) se muestra una representación del mapa de marcas visuales y sus covarianzas junto con la trayectoria de la cámara, así como los correspondientes nodos. La representación en profundidad inversa muestra una mejora en la linealización de la medida en desplazamientos limitados en comparación con la distancia de las marcas. Sin embargo, esto es cierto debido a la configuración del movimiento de la cámara, que implica principalmente desplazamientos laterales en paralelo, manteniendo las distancias de las marcas casi constantes, tal como se muestra en la Fig 2.25 (b). Esta suposición se hace también en el trabajo [27]. Por otra parte, el algoritmo es apropiado para entornos interiores con fuertes relaciones entre las diferentes regiones (nodos), pero no es de esperar que mejore significativamente la estimación en grandes entornos exteriores.



(a) Mapa de marcas visuales y trayectoria de la cámara (b) Representación del proceso de medida de marcas y de la creación de nodos, según un movimiento lateral de la cámara. Se indica el nodo activo.

Figura 2.25: Representación del mapa y del proceso de medida.

En [37] se presenta un sistema, que funciona casi en tiempo real, basado en una cámara estéreo enfocada hacia el suelo y en un sensor inercial (ver Fig. 2.26). El sistema extrae marcas visuales, estimando su localización usando la geometría correspondiente. Las marcas se definen utilizando características invariantes a la rotación y a la escala denominadas SURF (*Speeded Up Robust Features*) [38], con un tiempo de detección y comparación reducido en comparación con el método SIFT. Para buscar correspondencias entre las marcas se usa una variante del algoritmo denominado *PROSAC* [39]. La técnica de mapeado se basa en construir un grafo de relaciones en el espacio. Para la optimización del mapa se aplica un proceso eficiente basado en el descenso del gradiente, obteniendo así el mapa del entorno más probable. En la Fig. 2.27 se muestra una representación del mapa, reconstruido a partir de las imágenes capturadas, antes y después de un cierre de lazo. El sistema ha sido evaluado en exteriores e interiores, además de ser embarcado en un dirigible. El principal problema es consecuencia del reducido campo de visión del sistema. Esto implica una

fiabilidad baja, a la hora cerrar lazos, cuando se está capturando una textura altamente repetitiva.



Figura 2.26: Sistema estéreo empleado en las pruebas del sistema, constituido por dos cámaras web y un sensor inercial de bajo coste.

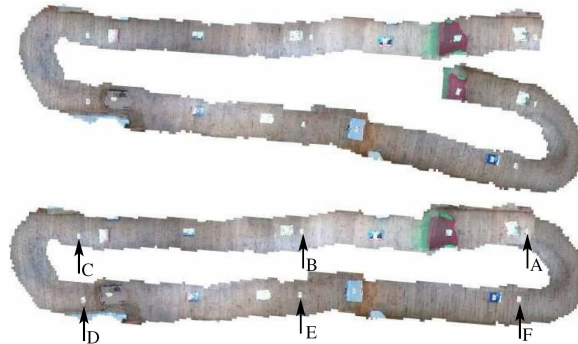


Figura 2.27: Vista superior del mapa en un entorno interior. La imagen superior muestra la estimación del mapa basado en las medidas visuales. La imagen inferior muestra esta estimación después de la optimización basada en mínimos cuadrados. Las etiquetas A - F representan marcas de control.

c) *Métodos jerárquicos de tipo Topológico-Topológico.*

Una tercera alternativa para solventar el problema del SLAM en grandes entornos consiste en usar únicamente mapas topológicos, sin sub-mapas asociados a sus nodos. Estos mapas carecen de los detalles métricos del entorno, pero proporcionan buenos resultados en ciertas aplicaciones.

En el trabajo [40] se presenta un sistema de SLAM visual orientado de una forma *minimalista*, es decir, con un reducido número de sensores y requerimientos de cómputo. Esta alternativa se basa en una representación gráfica de las poses del robot y de los enlaces entre ellas, basándose en odometría y en la similitud de las imágenes de una cámara omnidireccional. Para generar un mapa globalmente consistente se hace uso del método MLR [41]. Para estimar las matrices de covarianza de los enlaces obtenidos por el sensor de visión, se introduce un método novedoso basado en la similitud relativa entre imágenes cercanas entre sí. De esta forma, se evita el cálculo de distancias a marcas de imagen o la geometría de múltiples vistas (estéreo, trifocal, cuadrifocal, etc.). Se muestran resultados de experimentos, con el sistema embarcado en un robot móvil, en los que esta aproximación se adapta bien para grandes entornos, produciendo mapas topológicamente correctos y geoméricamente precisos con un coste computacional mínimo. En la Fig. 2.28 se muestra el robot empleado en los experimentos, así como el recorrido estimado por el sistema,

en un entorno mixto de interiores y exteriores, comparado con una señal de *ground truth*. Como línea futura se planean la inclusión de un exhaustivo análisis del tiempo de cómputo, sustituir la cámara omnidireccional por una estándar e incorporar odometría visual.

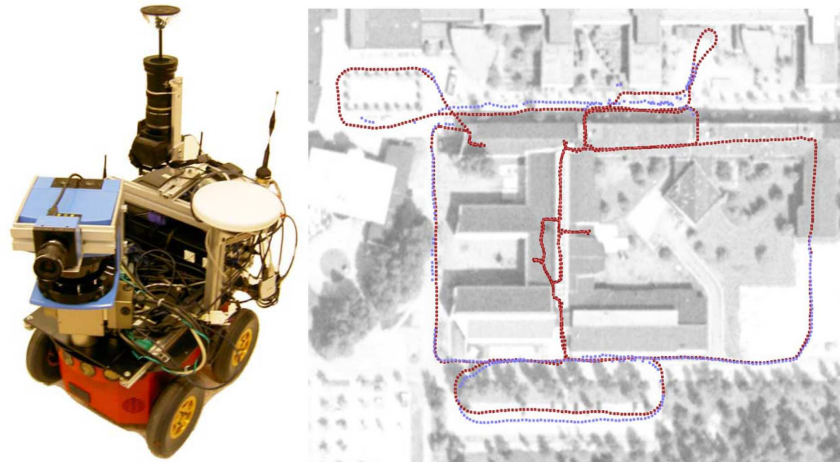
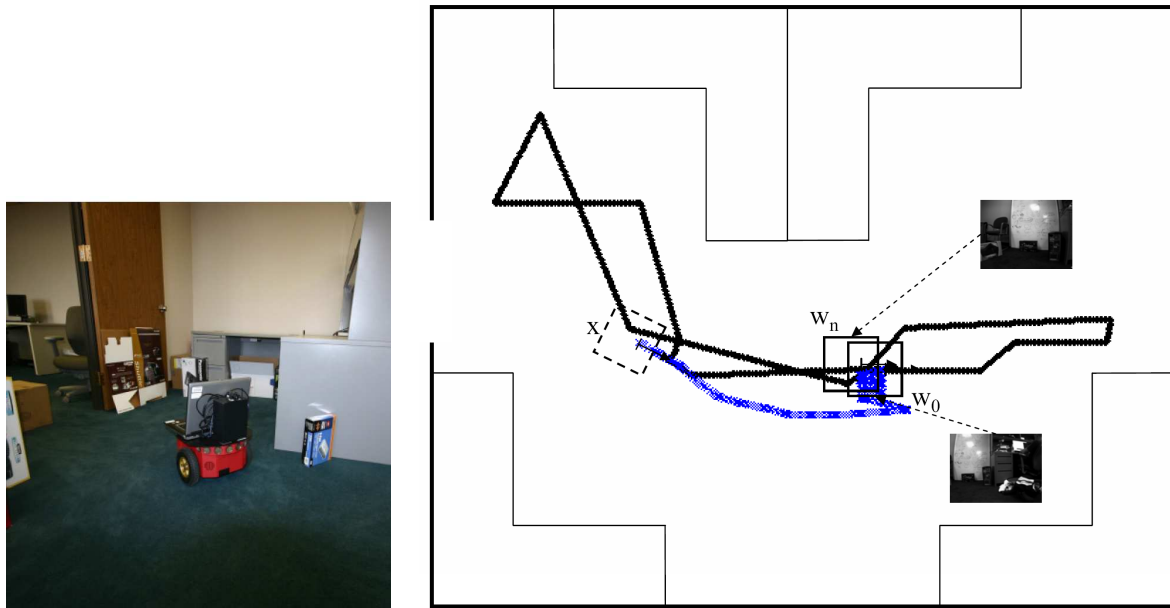


Figura 2.28: A la izquierda se muestra el robot móvil utilizado para los ensayos. A la derecha, en color rojo, se muestra el recorrido estimado por el sistema y, en color azul, la referencia obtenida con un DGPS (cuando está disponible)

Otra alternativa se presenta en [20], en la que se construye un mapa topológico capturando y almacenando colecciones de imágenes, frame a frame, y comparándolas con las anteriormente almacenadas. A medida que el robot explora el entorno se va creando simultáneamente una representación topológica de éste, añadiendo las imágenes a una base de datos y manteniendo un grafo de relaciones. El sistema se basa en un esquema eficiente de comparación y almacenamiento de descriptores SIFT. Esto permite realizar un mapeado casi en tiempo real además de una localización global. El formato compacto de representación de imágenes permite crear colecciones de millones de imágenes, permitiendo la creación de mapas de entornos de grandes dimensiones. Se han realizado ensayos en los que la búsqueda de una imagen de tamaño 640x480 entre un millón de imágenes almacenadas se lleva a cabo en 20 ms. También se propone un método de planificación de recorrido empleando la búsqueda en grafos. Para la navegación en el mapa topológico se utiliza información geométrica local. Los experimentos se han llevado a cabo usando un robot móvil equipado con una cámara, capturando imágenes de 640x480 a 15 Hz. El sistema se ha probado en un entorno interior, mostrado en la Fig. 2.29 en el que se han tomado 533 imágenes y se ha cerrado un lazo. A pesar de su eficiencia, finalmente el tiempo de cómputo termina por exceder el tiempo real cuando se han capturado un gran número de imágenes.

En el trabajo [21] también se usan descriptores SIFT para construir un mapa topológico basado en la apariencia. Dado que éste es sólo topológico, no se obtiene ningún tipo de información del movimiento propio del vehículo desde un punto de vista métrico. La principal contribución está en la forma en la que se manejan gran cantidad de frames clave (*keyframes*) para distinguir si el vehículo está recorriendo nuevos lugares o revisitando lugares antiguos, es decir, cerrando un lazo. Desde un punto de vista probabilístico, la comparación entre *keyframes* se basa en la probabilidad de que dos imágenes diferentes provengan del mismo lugar. Esta estimación depende también de las propiedades individuales del *keyframe*, como la repetitividad de patrones en la imagen, es decir, qué nivel de correlación existe entre los descriptores dentro de la imagen. Esto es muy interesante en



(a) Robot utilizado en el entorno de pruebas

(b) Experimento para el cierre de lazo. El robot comienza explorando aleatoriamente la habitación, comenzando en w_0 . Se detecta un cierre de lazo en la posición w_n mostrándose una correspondencia con el punto inicial. En ese momento el robot ya se encuentra en la posición X, iniciándose posteriormente la planificación del recorrido para volver a w_0 . Este camino es óptimo debido a la detección del cierre de lazo.

Figura 2.29: Robot utilizado y mapa del entorno de pruebas.

casos en los que el campo visual es habitualmente estrecho, por ejemplo si la cámara está enfocada lateralmente al desplazamiento del vehículo. En ese caso la riqueza en la textura de la imagen tiende a ser baja, como ocurre por ejemplo con imágenes de paredes, hojas de los árboles, etc. Las pruebas realizadas incluyen un recorrido en un entorno exterior de grandes dimensiones, tal como se muestra en la Fig. 2.30. En éste se muestran los puntos en los que el sistema reconoce lugares previsitados con una alta probabilidad, es decir, detecta un cierre de lazo.

d) *Sistemas basados en Odometría Visual.*

Algunas de las últimas contribuciones para la estimación de recorridos en grandes entornos usando sensores visuales se han concentrado en recuperar sólo el recorrido local del vehículo usando odometría visual, añadiendo además un nivel topológico para obtener una solución globalmente consistente. Estos métodos no necesitan estimar la posición de marcas externas dado que usan otras estrategias para el cierre de lazos y la corrección de la posición global.

En el trabajo [42] se describe un método basado en imágenes para estimar la trayectoria, con seis grados de libertad ($6DOF$), de una cámara estéreo usando la correspondencia entre pares de imágenes de referencia en lugar de la reconstrucción explícita de las marcas 3D de una escena. El proceso consiste en tomar un par de imágenes de referencia en un frame determinado, para posteriormente estimar un nuevo par de imágenes usando un modelo de movimiento de la cámara y distorsionar (*warping*) el par de referencia, según la geometría cuadrifocal, tal como se muestra en la Fig. 2.31(a). Para ello se utiliza un método de minimización densa, que usa directamente toda la información de escala de grises del par de imágenes. Después se busca el mínimo de una función de coste, resultado

de la diferencia entre el par de imágenes predicho y el actual, esto es el proceso de medida.

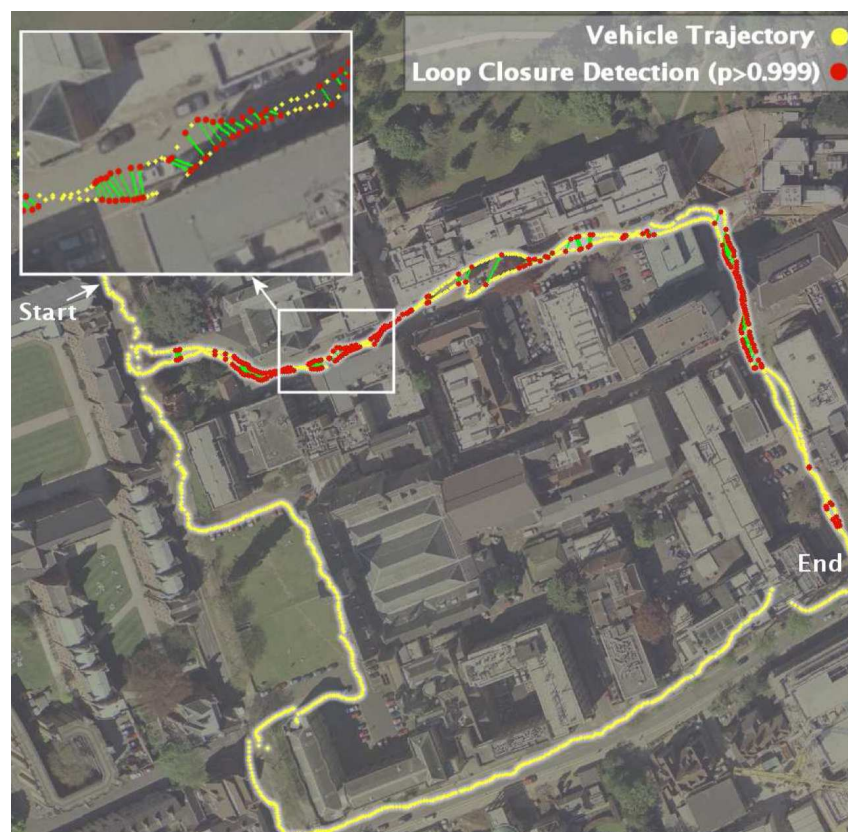
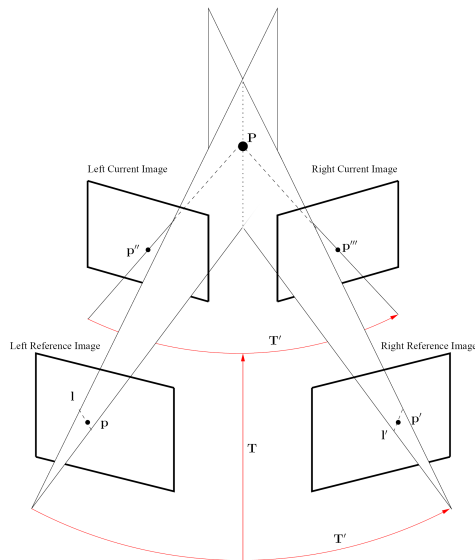


Figura 2.30: Resultados de las pruebas de *matching* basado en apariencia. El vehículo circula en el sentido de las agujas del reloj formando un lazo con una longitud de 1.6 Km, empezando por la parte superior izquierda y terminando en la parte inferior derecha. El cierre de lazo comprende dos de los lados del cuadrado. Las posiciones, obtenidas con un GPS, en las que el vehículo obtiene una imagen se marcan con un punto amarillo. La obtención de estas posiciones no forma parte del sistema y sólo se muestran como ayuda en la representación gráfica. Los pares de imágenes cuya probabilidad de provenir del mismo lugar es mayor de 0.999, se marcan en rojo y se unen con una línea verde.

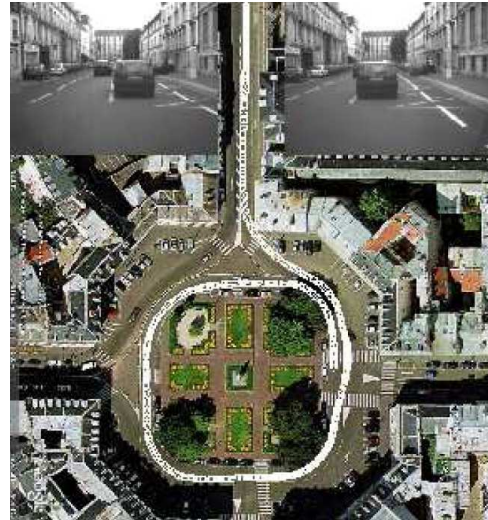
Por otro lado, se hace uso de una técnica, basada en estimadores M, para rechazar *outliers* correspondientes a objetos móviles en la escena, oclusiones o cambios de iluminación. Se muestran resultados de reconstrucción muy precisos de trayectorias en entornos urbanos exteriores, tal como se muestra en la figura 2.31(b). Una limitación del sistema es la suposición de planos paralelos para solucionar el problema de las correspondencias estéreo densas, válida para entornos de tipo urbano. Por otro lado, en cuanto a los tiempos de cómputo, los resultados muestran unos tiempos muy altos, desde 1 segundo por imagen hasta diez segundos por imagen, dependiendo de la precisión a obtener.

En el trabajo [43] se presenta un método para estimar la información de velocidad de translación y rotación de una pequeña cámara web integrada en un PC portátil (ver Fig. 2.32(a)), montado en un robot móvil, de una forma aproximada, evitando así el proceso computacionalmente costoso del seguimiento de marcas visuales. Para realizar esta estimación se selecciona una porción central de la imagen para posteriormente convertirla en un gráfico de columnas de intensidades. Éste se obtiene sumando los niveles de gris por columnas. Buscando la máxima correlación de los gráficos de frames consecutivos se puede

estimar la información de rotación.



(a) Geometría cuadrifocal en dos instantes de tiempo. Se inicializan de una vez dos puntos P y P' , en correspondencia, al comienzo del proceso de seguimiento. La pose central T se estima a través de una función de *warping* no lineal que se encarga de modificar todos los puntos en el par de imágenes de referencia transformándolos en los puntos del par actual P'' y P''' . La función de *warping* cuadrifocal se define eligiendo dos líneas l y l' que pasen por los puntos correspondientes en el primer par. Los parámetros extrínsecos T' se suponen conocidos a priori.



(b) Trayectoria estimada alrededor de una rotonda. Ésta se marca en blanco sobre la imagen de satélite, apreciándose la exactitud de la estimación sin implementar métodos de detección de cierre de lazos.

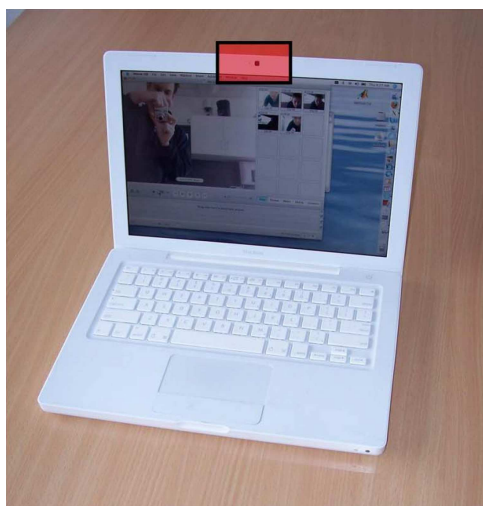
Figura 2.31: Representación de la geometría cuadrifocal junto con resultados de estimación obtenidos.

Para estimar la translación se comparan, de una forma heurística, los gráficos una vez desplazados hasta el punto de máxima correlación. Para evaluar el funcionamiento del método, éste se ha utilizado para proporcionar la información odométrica y visual al sistema de mapeado denominado *RatSLAM* [44] y así generar el mapa de una red de carreteras suburbana de gran tamaño. El método *RatSLAM* es capaz de generar un mapa coherente del entorno durante un recorrido de 18 Km con tráfico urbano a velocidades de hasta 60 Km/h. En la Fig. 2.32(b) se observa una gran cantidad de lazos en el camino recorrido por el vehículo. El algoritmo es muy simple y requiere un hardware muy sencillo. Sin embargo, es aproximado y se basa en ciertas suposiciones empíricas. Por otro lado, se muestra la necesidad de cerrar una gran cantidad de lazos para obtener unos resultados cualitativamente aceptables, pero cuantitativamente poco precisos.

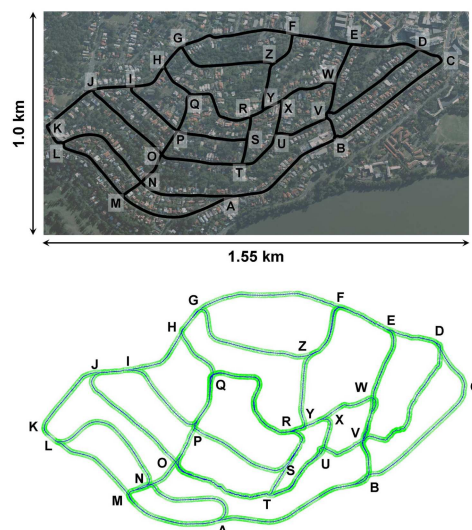
Otro ejemplo del uso de odometría visual se muestra en el trabajo [45], empleándose en este caso un montaje con dos pares de cámaras estéreo *Bumblebee*, embarcadas en un robot móvil. Este trabajo se describe en el próximo apartado, debido a otras características más significativas del mismo.

e) *Sistemas basados en Bundle Adjustment (Ajuste en conjunto).*

La técnica del *Bundle Adjustment* ha sido estudiada fundamentalmente dentro del campo



(a) Cámara web embarcada en un ordenador portátil, utilizada en el sistema

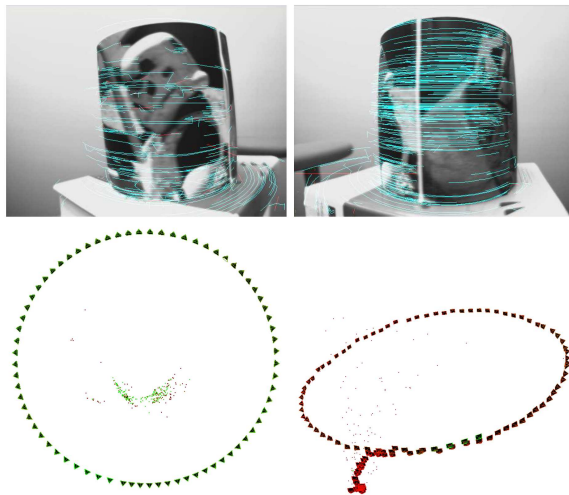


(b) La imagen superior muestra la trayectoria realizada por el vehículo. La imagen inferior muestra la estimación realizada por el sistema.

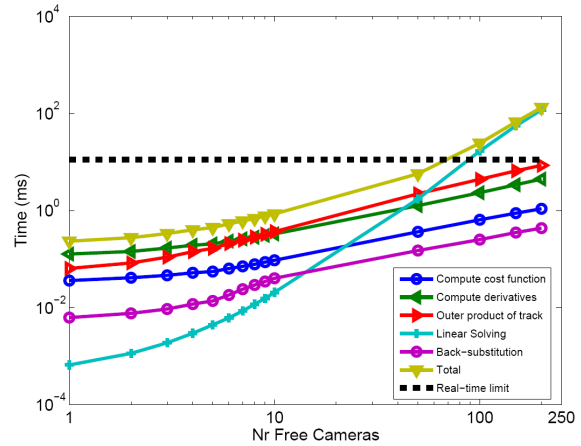
Figura 2.32: Hardware utilizado y resultados de estimación.

de la visión artificial, siendo ésta esencialmente *offline* en su origen. La técnica consiste en refinar una reconstrucción visual para producir una estructura 3D *conjuntamente óptima*, así como la estimación de la pose y/o calibración de una cámara. El término *óptima* se refiere a que la estimación de los parámetros se obtiene minimizando una función de coste que cuantifica el error en el ajuste a un modelo. *Conjunta* se refiere a que la solución es simultáneamente *óptima* con respecto, tanto a la variación de la estructura como a la de la cámara. El método también es conocido como SFM, sobre todo en aplicaciones orientadas a estimar la trayectoria de una cámara embarcada en un robot móvil, así como la estructura del entorno observado, a través del análisis de una secuencia completa de imágenes.

En el trabajo [46] se presenta un estudio experimental del *bundle adjustment* en el que se trata de seguir la trayectoria de una cámara monocular moviéndose en un entorno 3D. El propósito principal del trabajo consiste en demostrar que el *bundle adjustment* no sólo incrementa la precisión en la estimación de la trayectoria de la cámara sino que reduce la frecuencia de aparición de errores en la reconstrucción. Además se demuestra la viabilidad de implementación del método en tiempo real para pequeñas secuencias. El método consiste en ir capturando frames consecutivos, de forma que en cada imagen nueva añadida se inicializa la posición de la cámara utilizando el método RANSAC (*Random Sample Consensus*, “Consenso de muestras aleatorias”) [47]. El método se aplica de forma que las hipótesis se generan por medio de *reseciones* de tres puntos [48]. Éste método permite obtener la posición de una marca, midiendo su posición relativa a otras tres cuya posición sí es conocida. Los puntos visibles en cada nueva imagen se vuelven a triangular usando la reproyección en esta nueva imagen y en el primer frame en el que el punto fue visible. Entonces se lleva a cabo el proceso de *bundle adjustment* con n iteraciones en las m imágenes más recientes. El método además, no desecha los puntos marcados como *outliers* por el método RANSAC, sino que se ponderan por medio de una función de coste robusta. En la Fig. 2.33(a) se muestran las estimaciones de trayectoria realizadas usando el método presentado y sin usarlo. En la Fig. 2.33(b) se muestra la evolución del tiempo de cómputo en función del número de imágenes de la secuencia introducidas en el algoritmo.

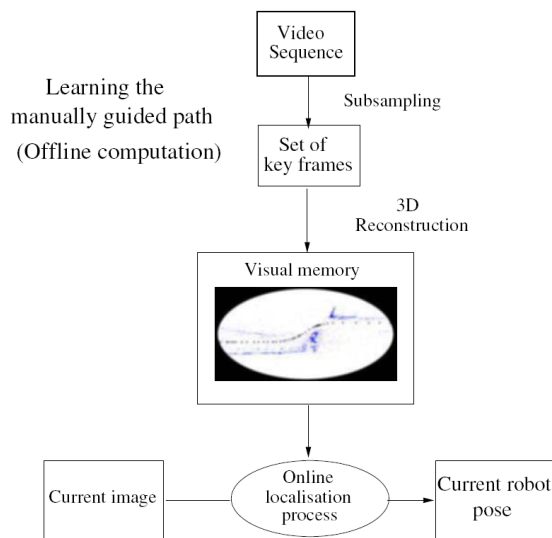


(a) En la imagen superior se muestra la secuencia de seguimiento, generada rotando un cilindro en una silla giratoria. En la imagen inferior izquierda se muestra la estimación realizada usando *bundle adjustment*, utilizando las últimas 20 imágenes, ejecutando 20 iteraciones cada vez que se añade una nueva imagen. La imagen inferior derecha muestra la estimación sin utilizar *bundle adjustment*, siendo ésta más irregular y propensa a fallos graves.

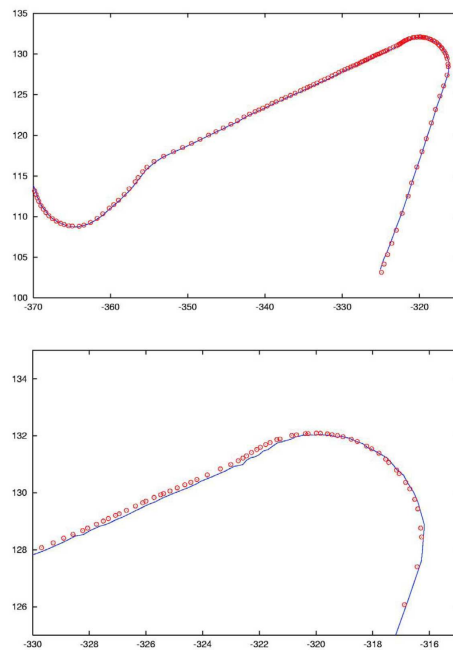


(b) Tiempos de cómputo por iteración en función del número de imágenes introducidas en el algoritmo. Se aprecia que el límite de tiempo real se pierde en torno a 60 imágenes procesadas.

Figura 2.33: Resultado de la estimación de trayectoria y tiempos de cómputo.



(a) Descripción general del sistema.



(b) Posiciones estimadas del robot (círculos rojos) en comparación con la trayectoria obtenida con un DGPS (unidades en metros). La imagen superior muestra la trayectoria completa y la inferior un detalle de la misma.

Figura 2.34: Descripción del sistema y estimación del recorrido.

En el trabajo [49] se presenta un método para calcular la localización de un robot móvil en base a una secuencia de vídeo previamente aprendida. En primer lugar, el robot es guiado manualmente por un determinado recorrido, mientras una cámara monocular graba la secuencia. Posteriormente, se calcula *offline* una reconstrucción 3D del recorrido y del entorno a partir de la secuencia aprendida. Este proceso se lleva a cabo usando una variante jerárquica de la técnica de *bundle adjustment*. Ésta consiste en dividir la secuencia original en dos secuencias de igual longitud, repitiendo el proceso en cada una de las secuencias resultantes de una forma jerárquica. El proceso de *bundle adjustment* se aplica entonces únicamente a las secciones individuales resultantes. En este punto se dispone de la secuencia completa de imágenes, estando cada una de ellas asociada a una pose determinada de la cámara. La reconstrucción 3D se usa entonces para obtener la pose del robot en tiempo real, realizándose una navegación autónoma. Este proceso se lleva a cabo comparando la imagen actual con las imágenes aprendidas utilizando puntos de interés de las mismas. En la Fig. 2.34(a) se muestra gráficamente la descripción general del método. En la Fig. 2.34(b) se muestra la estimación realizada por el método comparada con la referencia obtenida con un DGPS. La principal limitación del método reside en el hecho de que la fase de creación del mapa se realiza *offline*.

Por último, en [50] se presenta un método para estimar el movimiento de una cámara monocular calibrada en tiempo real, embarcada en un vehículo y enfocada hacia el frente, así como la geometría tridimensional del entorno, usando para ello únicamente la información visual. El procedimiento consiste en tomar cada cierto tiempo una imagen de referencia (*keyframe*) y obtener la pose del vehículo asociada a esta imagen respecto al anterior *keyframe*. Para ello se calculan las posiciones 3D de un gran número de marcas visuales, utilizando la geometría epipolar entre ambos *keyframes*. Éstos se elegirán lo más alejados posible el uno del otro, manteniendo un mínimo número de correspondencias ente marcas. El proceso de *bundle adjustment* local se ejecutará cada vez que se añada un nuevo *keyframe*. Dicho proceso se encarga de optimizar las n últimas imágenes, así como las posiciones de las marcas asociadas a éstas. Para ello se tendrán en cuenta además, las *reproyecciones* 2D de las marcas obtenidas en las últimas N imágenes, siendo $N \geq n$, tal como se muestra en la Fig. 2.35. N y n se ajustan de forma que sea posible mantener la restricción de cómputo en tiempo real. En la Fig. 2.36 se muestran los resultados de estimación de un recorrido de prueba, así como de reconstrucción del entorno asociado, de mediano tamaño. A pesar de la gran precisión mostrada en los resultados, cabe mencionar como contrapartida que para poder realizar el proceso de *bundle adjustment* local, sólo se obtiene la estimación de una pose después de un gran número de imágenes. Esto proporciona una reconstrucción del recorrido altamente dispersa. Por otra parte, el hecho de usar visión monocular implica la necesidad de tener un conocimiento a priori de la escena inicial, tal y como ocurre en otras alternativas monoculares presentadas anteriormente. Además, no se presenta ningún método para realizar una optimización global, en caso de cierre de lazos de grandes dimensiones. Por último, a pesar que los cálculos se realizan en tiempo real durante el procesado de imágenes normales, en los momentos de inserción de nuevos *keyframes*, estos tiempos de proceso exceden claramente el límite del mencionado tiempo real.

2.1.3. Sistemas basados en sensor GPS

A la hora de diseñar sistemas prácticos de localización y/o mapeado de grandes entornos, fundamentalmente exteriores, se suele recurrir a la fusión de varias fuentes de información proporcionadas por varios sensores entre los cuales tiene un papel fundamental el GPS. Algunos

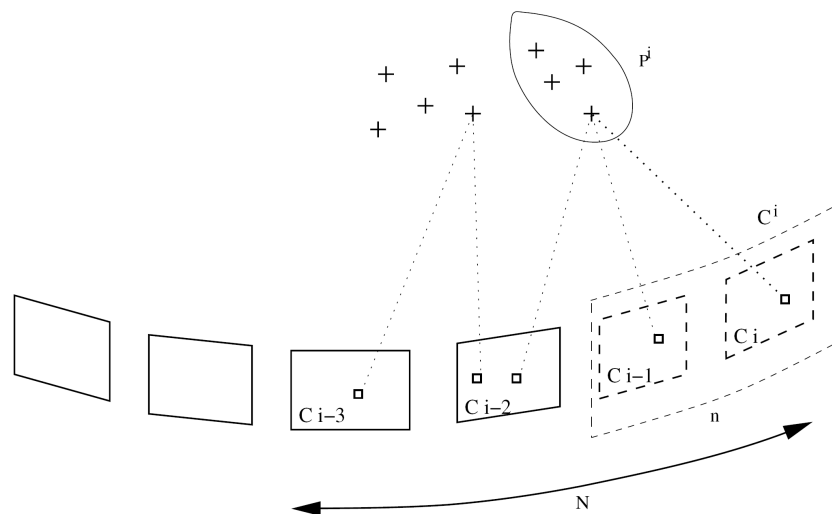
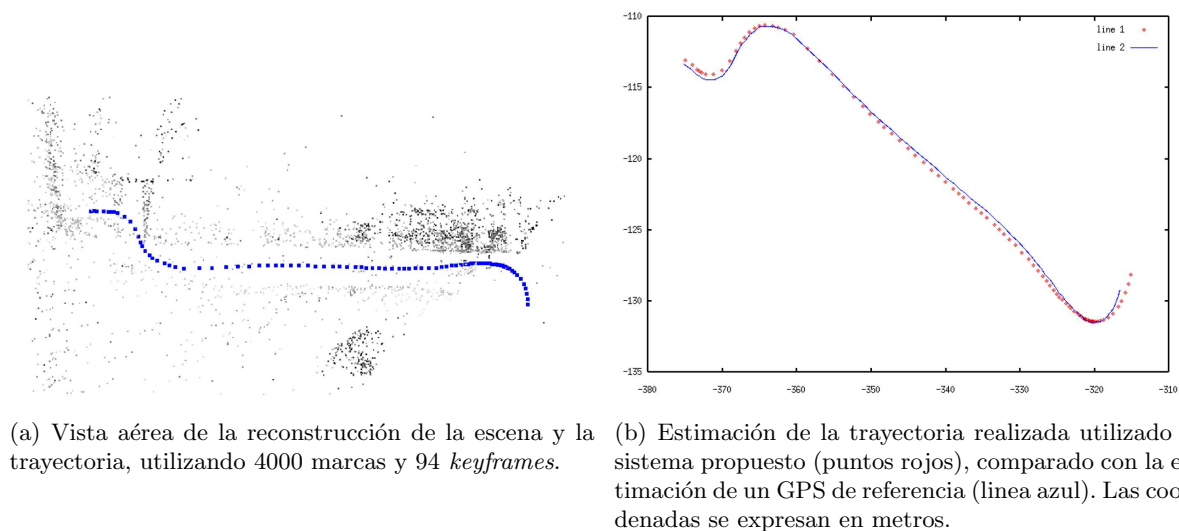


Figura 2.35: Proceso de *bundle adjustment* local cuando se añade un nuevo *keyframe* C_i . Sólo se optimizan las últimas poses y marcas asociadas. Sin embargo, se tienen en cuenta las *reproyecciones* 2D de las últimas N imágenes.



(a) Vista aérea de la reconstrucción de la escena y la trayectoria, utilizando 4000 marcas y 94 *keyframes*.

(b) Estimación de la trayectoria realizada utilizando el sistema propuesto (puntos rojos), comparado con la estimación de un GPS de referencia (línea azul). Las coordenadas se expresan en metros.

Figura 2.36: Resultados de estimación del sistema.

ejemplos destacables existentes en la literatura se describen a continuación.

En el trabajo [51] se presenta un sistema de SLAM multisensorial aplicado a un robot móvil para su navegación en exteriores. Se basa en el uso de odometría junto con medidas periódicas de un RTK-GPS y un LRF (*Laser Range Finder*, “Sensor de Distancia Láser”), integrando las medidas de ambos en un EKF. El mapa agrupa diversas marcas compuestas básicamente por paredes y aristas de edificios obtenidas a partir de lecturas del LRF, asumiéndose superficies planas y ángulos ortogonales. Sin embargo, el proceso de selección de marcas e identificación de las mismas debe ser realizado en una etapa previa por un operador. Cada vez que se obtenga una medida de GPS, ésta se fusionará con la pose actual teniendo en cuenta las incertidumbres de ambas. Entre cada medida de GPS se usará la técnica de *suavizado de Kalman* para interpolar la trayectoria del robot entre los dos puntos en los que dichas medidas han tenido lugar. Por otra parte, el vehículo debe realizar paradas periódicamente para obtener las medidas del GPS y el LRF. Esto hace al sistema inapropiado para realizar un SLAM automático y, por tanto, navegar con él en entornos desconocidos desde el principio. En la Fig. 2.37 se muestra una imagen del robot utilizado en las pruebas.

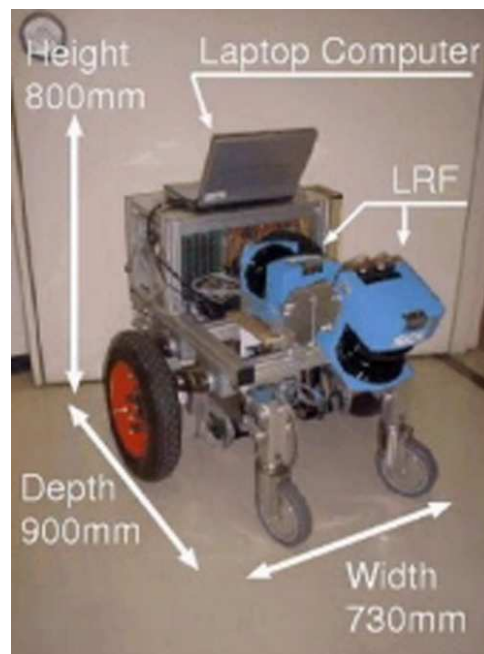
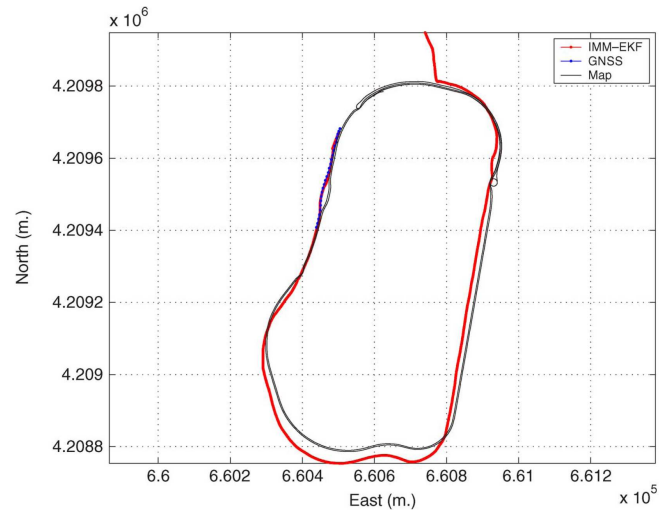


Figura 2.37: Robot móvil utilizado para las pruebas

En el trabajo [52] se presenta un método para estimar la localización de un vehículo (ver Fig. 2.38(a)) enfocado a aplicaciones de navegación. Éste se basa en un sensor GPS de altas prestaciones, de una unidad inercial INS, así como de la información de odometría del vehículo, todo ello fusionado usando un EKF combinado con un método multimodelo IMM-EKF (*Interactive MultiModel EKF*, “EKF MultiModelo Interactivo”). La idea del IMM-EKF se basa en el hecho de que el estado dinámico de un vehículo es altamente variable con el tiempo, y por lo tanto un modelo determinado no puede ajustarse perfectamente a todas las situaciones. Para solventar esto, se definen varios modelos diferentes en función de las situaciones, aplicándose el más adecuado en cada momento. El filtro IMM-EKF se encargará de calcular la probabilidad de éxito de cada modelo en cada paso de ejecución del EKF. Para evaluar el funcionamiento del sistema se realizan diversas pruebas en exteriores, en las que se muestra una gran precisión en la estimación. En la Fig. 2.38(b) se muestra un ejemplo de estas pruebas. Como inconveniente se puede mencionar el alto coste del sistema.



(a) Vehículo utilizado para las pruebas

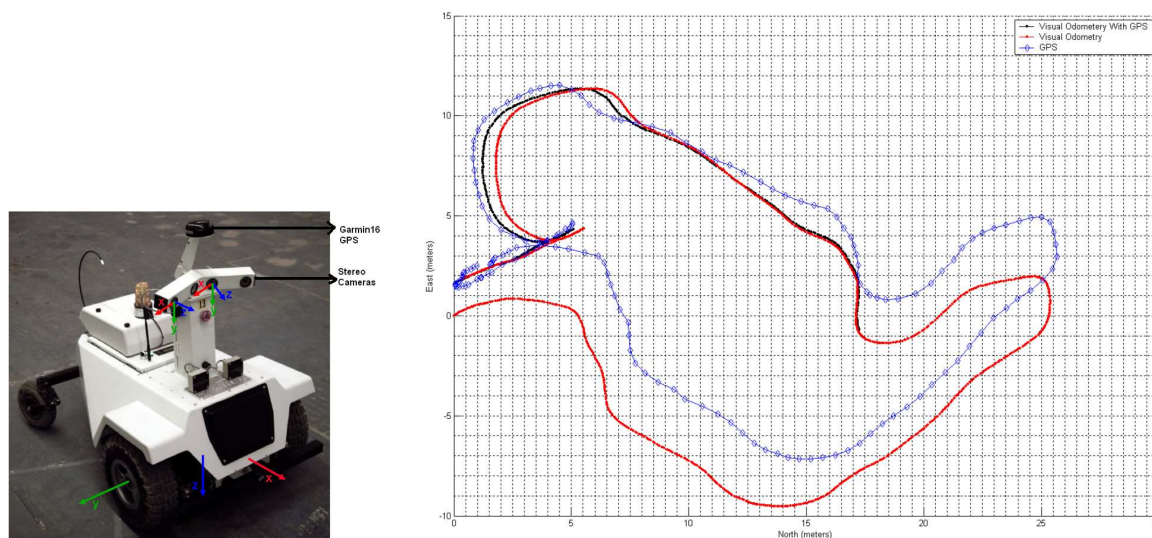


(b) Resultados de la estimación del recorrido realizado por el vehículo utilizando el método IMM-EKF, junto con las medidas del GPS GNSS, la odometría del vehículo y la referencia del mapa.

Figura 2.38: Vehículo empleado y estimación del recorrido.

Otro ejemplo de fusión de sensores se presenta en [45]. En este caso, se usa una cámara estéreo *Bumblebee*, una unidad de medida inercial (IMU) y odometría basada en encoders como sensores de tipo relativo (ver Fig. 2.39(a)). Por otra parte, se usa un GPS de bajo coste como sensor de medida absoluto. El sistema de odometría visual implementado utiliza el seguimiento de marcas visuales para estimar el movimiento relativo entre dos frames. Para inicializar las marcas se utiliza un detector de esquinas Harris [10] en la imagen izquierda de cada uno de las pares de cámaras, para posteriormente obtener sus correspondencias por triangulación. Tres de esas marcas se usan para estimar el movimiento, empleando orientación absoluta. Los movimientos relativos entre frames consecutivos se van encadenando para obtener la pose absoluta en cada frame. La pose inicial se obtiene a partir de la IMU, los encoders y el sensor GPS, moviendo el robot en una línea recta a velocidades superiores a 1 m/s. Los datos de la IMU y los encoders se usan además para suplir la información de odometría visual cuando ésta falla, complementándose así unos sensores con otros. Finalmente, se fusionan los datos de la posición local con las medidas provenientes del GPS usando un filtro de Kalman, evitando así la deriva temporal. De esta forma, se demuestran unos errores relativamente bajos en entornos de tamaño moderado, esto es, alrededor de 100 metros, tal como se muestra en la Fig. 2.39(b). El sistema se ha ensayado en exteriores en un espacio abierto de tipo no urbano. Por este motivo la precisión del GPS es buena y su disponibilidad está casi siempre garantizada. El hecho de que no se emplee ningún método específico para el manejo de grandes entornos, impide el uso del sistema en entornos mucho más grandes que los ensayados, más aún si no se dispone bien de odometría o bien de sistemas inerciales (IMU).

Por último, en un ámbito esencialmente comercial se han desarrollado varios dispositivos que complementan las estimaciones proporcionadas por el GPS usando, fundamentalmente sensores inerciales. A modo de ejemplo, el sistema Inertial+ [53] proporciona estimación de pose del vehículo utilizando sensores inerciales e incluso obteniendo información de odometría del vehículo (ver Fig. 2.40). Estos sistemas proporcionan buenos resultados para pequeñas pérdidas de señal GPS pero su error de estimación aumenta en función del tiempo de no recepción de señal GPS fiable. Presentan otros inconvenientes como su elevado coste o la necesidad de disponer de



(a) robot utilizado en las pruebas, equipado con dos pares estereo modelo Bumblebee

(b) Resultados de la estimación del recorrido realizado por el robot utilizando odometría visual fusionada con GPS, junto con la estimación usando únicamente odometría visual y únicamente GPS.

Figura 2.39: robot empleado y estimación del recorrido.

señales odométricas del vehículo, información esta no estándar y celosamente guardada por los fabricantes, lo que conlleva que sea difícil de conseguir.

2.2. Discusión y enmarque de la Tesis

Las principales conclusiones que se pueden extraer tras la descripción del estado del arte se centran en los siguientes aspectos:

- En primer lugar, cabe destacar un amplio uso de cámaras de visión monocular como sensor principal a la hora de estimar la pose de un móvil. Las limitaciones fundamentales de estos sistemas, frente a las cámaras estereo, se centran en la estimación de la posición de marcas visuales. Estas limitaciones se derivan de la imposibilidad de estimar la coordenada de profundidad de dichas marcas de una forma directa.
- Por otra parte, diversos trabajos demuestran la eficacia del uso de descriptores SIFT, tanto a la hora de identificar marcas visuales integradas en el mapa, como en su utilidad para identificar entornos en base a su apariencia. La principal limitación en el uso del método SIFT, es el relativamente elevado tiempo de cómputo necesario, demostrado en la mayoría de las aplicaciones. Este es un dato a tener en cuenta en aplicaciones de tiempo real.
- Los sistemas basados en el método FastSLAM se muestran como una alternativa al uso del EKF. Éstos muestran una mejora en las estimaciones del recorrido, combinadas con una mejora en el coste computacional. Además, se muestran como una alternativa a los métodos jerárquicos para entornos de tamaño medio-grande. Sin embargo esta eficiencia decae rápidamente en función del número de partículas utilizadas en el filtro.
- En cuanto a los métodos jerárquicos, queda patente su gran expansión en cualquiera de sus variantes. Se demuestra que la división del problema general de mapeado en un entorno de



Figura 2.40: Sistema Inertial+ basado en GPS, sensores inerciales y odometría.

grandes dimensiones reduce su problemática asociada si se divide éste en otros problemas (sub-mapas) más pequeños.

- Los métodos de tipo *métrico-métrico*, aunque son capaces de proporcionar estimaciones de gran precisión, no mantienen una estructura topológica que ayude a una optimización global en grandes entornos, así como en técnicas de planificación de recorridos para tareas de navegación.
- Las alternativas puramente *topológicas* no son capaces de proporcionar información precisa del estado del vehículo, llegando incluso a no proporcionar información absoluta de su localización. Ésta es necesaria fundamentalmente en entornos poco estructurados, en los que no existen suficientes elementos que sirvan como referencia a la hora de construir el mapa.
- Los métodos *topológico-métrico*, por el contrario, proporcionan un adecuado balance en cuanto a la información aportada por ambos enfoques. Se ha presentado una numerosa bibliografía encuadrada en este enfoque, que aporta diversos puntos de vista a la hora de estructurar el mapa topológico y enlazarlo con el nivel métrico. Una gran parte de ellas son más apropiadas para entornos muy estructurados, mientras que otras, como el NCFM permite una mayor flexibilidad de implementación en entornos más genéricos.
- Se han revisado también trabajos basados en los métodos denominados como *odometría visual*, que han recibido un gran interés reciente. La principal característica de estos métodos es la ausencia de un mapa como tal, siendo de interés únicamente la estimación de la trayectoria del vehículo. Por otra parte, para estimar la trayectoria del vehículo, sólo se tienen en cuenta dos frames individuales, comparándolos entre ellos. De esta forma, una vez utilizada la información de un par de frames, éstos se desechan. Estos métodos demuestran unos buenos resultados en sus diferentes formas de implementación, siendo una alternativa interesante para aplicaciones en las que no se necesite la estimación de un mapa.

- Como contraposición a los métodos de odometría visual, las técnicas de *Bundle Adjustment* tienen en cuenta a la vez todos los frames de una determinada secuencia, a la hora de estimar la trayectoria de un móvil. Esto proporciona a estos métodos una gran precisión en la estimación, pero como contrapartida los tiempos de cómputo quedan alejados de los requerimientos de tiempo real.
- Por último se han revisado un compendio de trabajos caracterizados por el uso de GPS. Este tipo de soluciones siguen siendo necesarias si lo que se requiere es combinar una gran precisión con reducidos tiempos de cómputo, así como un mínimo nivel de fiabilidad en grandes entornos exteriores. Esto es aplicable sobre todo en sistemas destinados a ser embarcados en automóviles. La mayor parte de los sistemas combinan sensores de tipo *relativo*, como odometría, con sensores de tipo *absoluto* como el GPS. Sin embargo, se aprecia una ausencia de sistemas, basados en métodos jerárquicos, que combinen ambos sensores en los diferentes niveles implementados. También se observa la necesidad de que estas soluciones sean implementables con un coste lo más reducido posible.

2.3. Objetivos específicos de la Tesis

Partiendo de los objetivos iniciales planteados al comienzo de esta Tesis, se han evaluado las diferentes alternativas existentes en el estado del arte, aprovechando aquellos aspectos más beneficiosos de las mismas y mejorando las áreas que contribuyan a conseguir los objetivos iniciales, anteriormente mencionados. A partir de esta información, los objetivos específicos de este trabajo se enumeran a continuación:

1. El primer objetivo a acometer, consistirá en el diseño e implementación del sistema de visión. Concretamente se estudiarán las ventajas de utilizar una cámara estéreo frente a un sistema monocular, analizándose la configuración óptima de las mismas. Adicionalmente se estudiarán las ventajas del empleo de lentes de gran angular.
2. Para la elección de las cámaras se realizará un estudio comparativo de las características fundamentales de las mismas, de cara a elegir la configuración que mejor se adapte a las necesidades del sistema
3. Se estudiará e implementará un método de calibración que permita dotar al sistema de la precisión necesaria en la obtención de las medidas correspondientes, fundamentalmente teniendo en cuenta la elección de cámaras de gran angular con una elevada distorsión.
4. Se elaborarán modelos matemáticos para solucionar el problema del SLAM basado en EKF, adaptado tanto al uso de un sistema visual estéreo de gran angular, como al hecho de embarcarlo en un automóvil para entornos exteriores.
5. Se estudiarán alternativas a la hora de elegir el tipo de marcas visuales a implementar en el sistema. Para ello se tendrán en cuenta aspectos como la adaptación al proceso de medida, teniendo en cuenta el tipo de entorno y de vehículo empleado, así como las necesidades de cómputo, teniendo en cuenta la restricción de tiempo real. También se desarrollarán mejoras en el comportamiento de las mismas, una vez elegido el tipo.
6. Partiendo de las alternativas presentadas en el estado del arte, se realizará un análisis del método global de SLAM que más se ajuste a la navegación con un vehículo en grandes entornos exteriores. Por lo tanto, dicho método será elegido de entre los destinados a grandes entornos.

7. Se diseñará un método jerárquico basado en sub-mapas independientes. Para la transferencia de información de incertidumbres entre sub-mapas se desarrollará un método con aspectos comunes al NCFM.
8. Se analizarán diferentes alternativas para la implementación de la optimización del mapa global en el nivel alto de la jerarquía. Concretamente, el estudio se centrará en los resultados del método MLR comparados con un método de corrección lineal propio.
9. Se diseñará un método de detección de cierre de lazos basado en huellas visuales. Éstas se basarán a su vez en los descriptores SIFT. Para la implementación de las correlaciones se realizará un estudio comparativo entre el método RANSAC y un método propuesto en esta Tesis.
10. Se desarrollará un método para la integración de la información proporcionada por un GPS de bajo coste. Esta integración se realizará en todos los niveles de la jerarquía.
11. Se llevará a cabo la validación de los algoritmos desarrollados e implementados a través de diversas pruebas, inicialmente con un robot móvil en entornos interiores y posteriormente con un vehículo en exteriores. Se realizará un análisis de los resultados obtenidos.
12. A modo de demostración de la utilidad práctica del sistema, se desarrollará una aplicación para la mejora de un navegador GPS. En él se evaluarán las aportaciones del sistema presentado en una aplicación práctica.

Capítulo 3

Visión general del sistema

3.1. Introducción

El propósito de este capítulo es el de justificar la elección de un método para llevar a cabo la localización y mapeado de un vehículo en grandes entornos, presentando la estructura del método a implementar y describiendo sus elementos fundamentales, antes de pasar a explicarlos en detalle en los siguientes capítulos.

3.2. Elección del método

Teniendo en cuenta las diferentes alternativas analizadas en el capítulo 2 de Estado del Arte, para solucionar el problema del SLAM en grandes entornos, las principales opciones se centran en la implementación de *odometría visual*, el uso de métodos de tipo *bundle adjustment* o el desarrollo de un método *jerárquico*.

En el caso de los métodos de odometría visual, su principal característica es la ausencia de la estimación de un mapa como tal, estimándose únicamente la trayectoria del vehículo. Este tipo de métodos no tienen la capacidad de reconocer lugares previamente visitados, por lo tanto son susceptibles de incurrir en errores acumulativos en la estimación del recorrido. Por otra parte, para realizar dicha estimación, estos métodos se fundamentan en la relación visual entre dos frames de una secuencia. Este hecho reduce el rendimiento obtenido de la secuencia, ya que la estimación que se realiza no es óptima al no tener en cuenta un determinado número de frames de forma global, lo cual reduce la precisión en la reconstrucción.

En el extremo opuesto se encuentran los métodos de tipo *bundle adjustment*. Estos métodos, en su esencia tratan de estimar la trayectoria de un vehículo así como reconstruir el entorno, utilizando para ello todos los frames disponibles en la secuencia, en un sólo bloque. De esta forma consiguen una estimación óptima del recorrido realizado. Sin embargo, como contrapartida, el tiempo de cómputo asociado a estos cálculos excede rápidamente el tiempo real, ejecutándose, por regla general, de un modo *offline*.

Teniendo en cuenta estas reflexiones, la elección más adecuada pasa por un punto intermedio, en el cual para realizar la estimación de pose local, en cada instante, se haga uso de la información visual obtenida durante un determinado número de frames. Con este objeto, se implementará un método de tipo jerárquico. Dentro de este tipo de métodos, para elegir una de las tres principales alternativas, se tendrá en cuenta que, por un lado, aunque los métodos de tipo Métrico-Métrico proporcionan estimaciones de gran precisión, éstos no mantienen una estructura topológica que contribuya a una optimización global en entornos de grandes dimensiones, así como en técnicas de planificación de recorridos en tareas de navegación. Por otra parte, las alternativas puramente topológicas no proporcionan una información precisa de la estimación del estado del vehículo. Por lo tanto, para resolver el problema de los grandes entornos se propone el uso de la alternativa jerárquica de tipo topológico-métrico, con una implementación en la línea del método NCFM.

3.3. Arquitectura general

Con el objeto de resolver el problema de SLAM en grandes entornos, la arquitectura del sistema presentado se basa en la fusión de *visión estéreo* con *GPS*. Para solventar el problema del crecimiento de la matriz de covarianza, intrínseco a los métodos de SLAM basados en EKF, se procederá a dividir el mapa global en diversos sub-mapas locales. Cada uno de estos sub-mapas posee su propio proceso de SLAM métrico, independiente de los otros sub-mapas. Sobre estos

sub-mapas locales se define un nivel de SLAM topológico superior que los relaciona, manteniendo la consistencia global del mapa, tal como se muestra en la Fig. 3.1. En esta tarea, el sensor GPS proporciona una contribución importante, ya que el error de estimación de la posición se incrementa con el tiempo en los sistemas de SLAM visual, a no ser que tengan lugar situaciones de cierre de lazos. Por otro lado, los errores en las estimaciones proporcionadas por el GPS son limitados, sin embargo, éstos pueden crecer localmente mucho más rápido de lo que la estimación visual lo hace, es decir, los sensores visual y GPS son complementarios entre sí, lo que justifica su fusión de cara a obtener un sistema de localización más preciso.

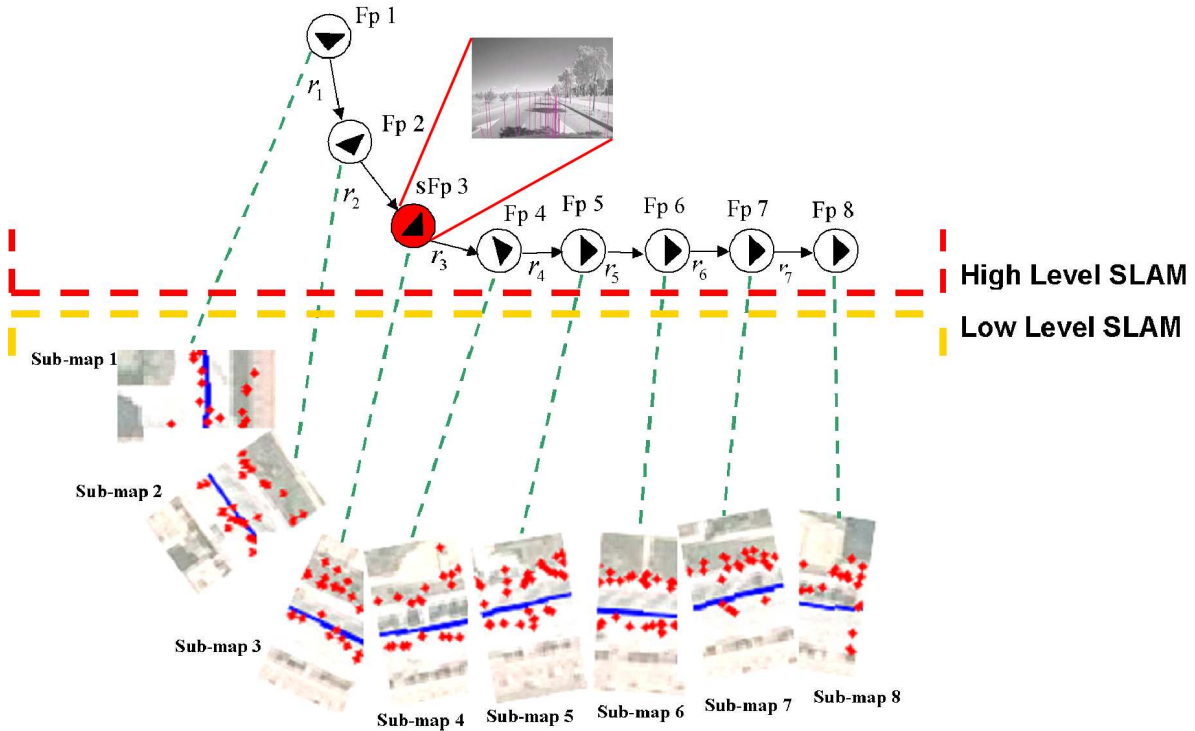


Figura 3.1: Arquitectura básica del sistema. En la parte inferior se muestran los diferentes sub-mapas, cada uno con su estimación independiente de un tramo del recorrido total, así como las marcas visuales. Todo ello constituye el SLAM de bajo nivel. En la parte superior se muestra el mapa topológico, en el que cada nodo está asociado a un sub-mapa. Estos nodos se muestran numerados del 1 al 8, así como las relaciones entre ellos. El nodo 3 tiene asociada una huella SIFT, como se indica en la imagen. Todo ello constituye el SLAM de alto nivel.

El sistema visual se basa en una cámara estéreo de gran angular, embarcada en un vehículo, orientada en la dirección de avance del mismo. En cada sub-mapa se capturan secuencialmente diversas marcas visuales que se introducen en el filtro EKF de cara a modelar el comportamiento probabilístico del sistema. Para la percepción de las marcas se utiliza un modelo de medida, mientras que para modelar el comportamiento dinámico del vehículo se utilizará un modelo de movimiento. Las medidas de GPS contribuirán a mejorar, tanto la estimación del vehículo como la del mapa.

El sistema implica el desarrollo de un método de SLAM jerárquico, el cual añade un nivel de procesamiento adicional denominado *Nivel Alto de SLAM* (High Level SLAM) al SLAM métrico, explicado anteriormente, que será denominado *Nivel Bajo de SLAM* (Low Level SLAM). De este modo, el mapa global se divide en sub-mapas locales, identificados por las denominadas *huellas*. El SLAM de bajo nivel estará asociado a cada uno de los sub-mapas, aplicando en ellos el

sistema visual de marcas y filtrado EKF que se explicará más adelante, en el capítulo 4. Las huellas almacenan la pose del vehículo en el momento de la creación del sub-mapa y definen su propio sistema de referencia local. La generación del sub-mapa se lleva a cabo periódicamente en el espacio, es decir, después de recorrer un determinado tramo del recorrido, se creará un nuevo sub-mapa, asociándose una nueva huella a éste. Si el vehículo se mueve de una huella a otra, se insertará una transición para conectar ambos nodos, lo que representa un enlace entre dos poses. Por otro lado, las transiciones almacenan las matrices de transformación e incertidumbres que describen la relación entre las huellas conectadas. Para optimizar la detección del cierre de lazos, cuando se detecta un giro significativo del vehículo, se tomará un tipo especial de huella denominado *huella SIFT*. Esto añade a la pose del vehículo una información visual que identificará el lugar en el que ésta se tomó. Entonces, para detectar zonas previamente visitadas, se realizará una comparación entre las huellas SIFT previamente capturadas, dentro de una región de incertidumbre, y la huella actual. En caso de que la comparación resulte positiva, se identificará la situación de cierre de lazo, corrigiéndose el mapa topológico. Todo ello proporciona robustez al sistema en caso de no disponibilidad del GPS.

Cada vez que está disponible una nueva medida de GPS, ésta se introduce en el sistema. Esto se lleva a cabo fusionando las estimaciones de pose 2D del vehículo obtenidas a partir del sensor visual y del GPS, teniendo en cuenta las covarianzas de incertidumbre de ambas.

En la Fig. 3.2 se muestra una tabla indicando las tareas fundamentales del sistema y su ubicación en la jerarquía.

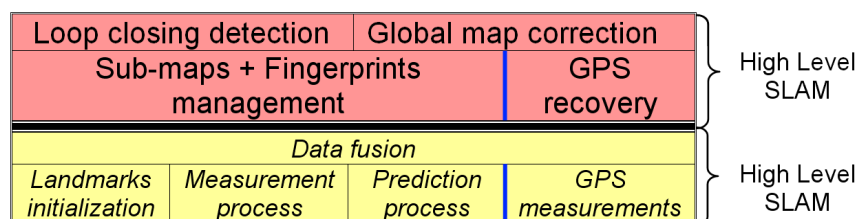


Figura 3.2: Tareas fundamentales del sistema según su ubicación en el nivel alto de SLAM o en el nivel bajo de SLAM.

En cuanto a los elementos principales del sistema, en la Fig. 3.3 se muestra la arquitectura física del sistema.

3.4. Conclusiones y aportaciones

Como conclusión del capítulo, por una parte se ha realizado un análisis cualitativo de las diferentes alternativas de implementación del método de SLAM, llegando a la conclusión de que el método de SLAM jerárquico de tipo Topológico-Métrico es el más adecuado para la aplicación planteada en esta Tesis. Por otra parte se ha presentado de manera general la arquitectura, tanto del método como del sistema físico.

Las principales aportaciones del método propuesto respecto a los métodos topológico-métricos presentados en el estado del arte, se pueden resumir en, por un lado, una estrategia de asociación de datos más robusta para el cierre de grandes lazos basada en huellas SIFT y, por otro lado, una gestión de la relación entre nodos más simple, siendo ésta más apropiada para grandes entornos urbanos. Además, gracias al uso de visión estéreo, se mantiene una correcta estimación de la escala del mapa, incluso antes del cerrar de lazos o visitar lugares.

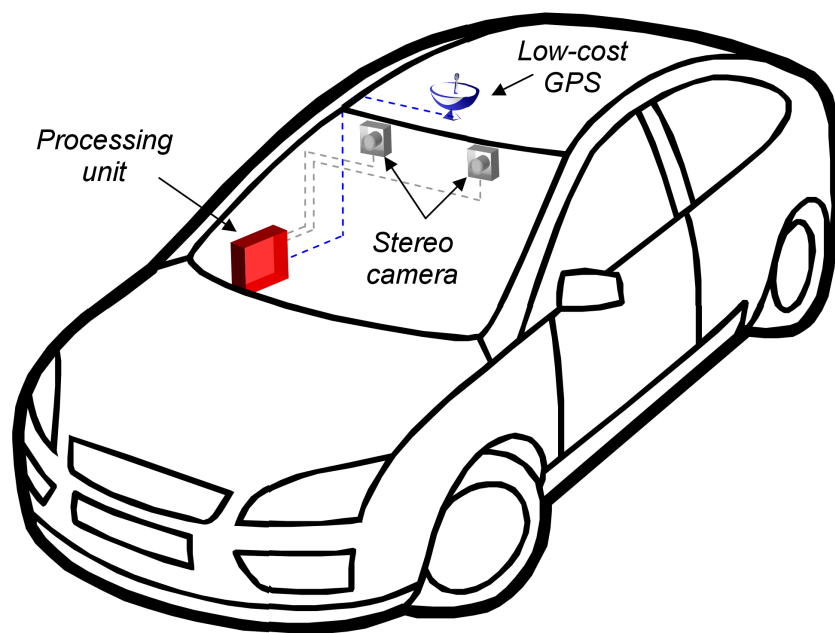


Figura 3.3: Elementos principales del sistema presentado. Se destacan el vehículo, la cámara estéreo, el receptor GPS de bajo coste y la unidad de procesamiento.

Capítulo 4

SLAM de bajo nivel

4.1. Introducción

En este capítulo se describen los algoritmos y tareas implementadas que son necesarias para localizar el vehículo y realizar el mapeado del entorno, en un área local definida por un determinado sub-mapa. Estos algoritmos y tareas se denominan SLAM de bajo nivel. El principal sensor utilizado a este nivel es la cámara estéreo. Dado que se ha escogido el mismo sistema de referencia tanto para la cámara como para el vehículo, por simplicidad se hablará de uno u otro indistintamente. En este nivel inicial de procesamiento no se dispone de información adicional de la pose ni de control sobre el movimiento de la cámara. Esto influirá decisivamente en el tipo de modelo de movimiento elegido. En este nivel el enfoque empleado para la realización del SLAM es el *métrico*.

Dado que el sistema está diseñado fundamentalmente para entornos urbanos, el mapeado del entorno se realizará en 3D mientras que la localización del vehículo se supondrá sobre terreno plano. La solución aportada para este nivel está inspirada en el trabajo de A. Davison [7], el cual utiliza como sensor una única cámara. Este trabajo consiste en el desarrollo de un método de SLAM basado en visión monocular usando un EKF.

Uno de los principales inconvenientes a la hora de resolver el problema del SLAM de bajo nivel está relacionado con la manera de construir el mapa local. Debido a los errores acumulativos en las medidas, inevitablemente se incurre en una deriva espacial durante el proceso de construcción del mapa. Esto conlleva la posibilidad de no reconocer lugares previamente visitados (problema de asociación). Por lo tanto, de cara a conseguir una buena localización por largos periodos de tiempo, será necesario incluir el mapa local completo en el proceso del filtro, es decir, en el vector de estado. Esto implica un aumento ilimitado del tamaño del vector de estado y matriz de covarianza asociada, teniendo como consecuencia el incremento del tiempo de cómputo, sobrepasando rápidamente la restricción de tiempo real. Para solucionar este problema, cada submapa deberá estar limitado en tamaño, de forma que se garantice la mencionada restricción de tiempo real.

Teniendo en cuenta lo explicado anteriormente, el mapa estará formado por una serie de marcas naturales. Estas marcas se identificarán por sus correspondientes *apariencias* una vez capturadas por la cámara. A medida que la cámara se va moviendo por el entorno, visitando nuevos lugares, se van capturando nuevas marcas que se irán añadiendo al vector de estado completo (dentro del sub-mapa).

Estas marcas no son simplemente el resultado del proceso de mapeado, sino que además son el medio por el que el vehículo es capaz de localizarse a sí mismo. Este proceso se basa en el uso de un modelo de proyección inverso, de forma que teniendo las coordenadas de proyección de cada marca en ambas cámaras es posible calcular su posición 3D respecto al sistema de referencia del sub-mapa. Obteniendo la posición relativa al sub-mapa de diferentes marcas es posible deducir la **posición** y **orientación** de la propia cámara respecto del mismo sistema del sub-mapa.

4.2. Vector de estado

El primer paso a la hora de definir los elementos del EKF será la descripción del vector de estado. Por razones de claridad la notación que identificaría el submapa respecto a los demás ha sido omitida.

Dada la estructura del sistema se ha optado por dividir el vector de estado en dos partes. Por un lado, el vector del *estado estimado del vehículo*, y por otro las posiciones de las diferentes

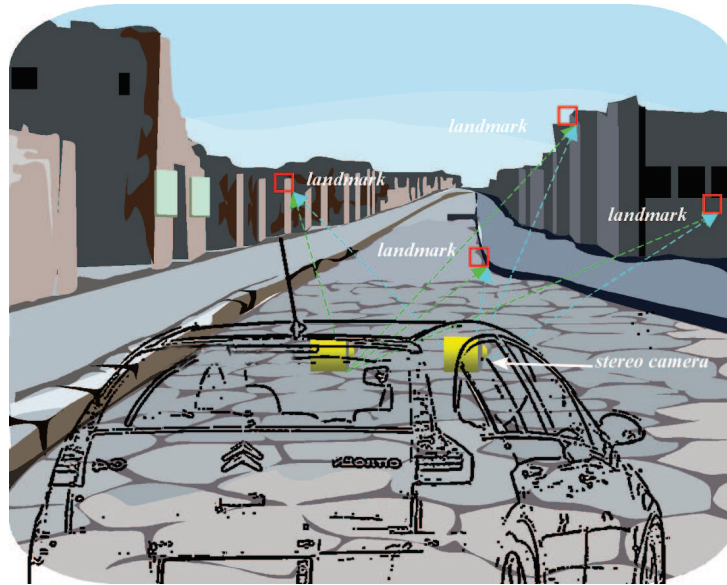


Figura 4.1: Proceso de captura de marcas

marcas del mapa. Respecto al primero de ellos, éste viene definido por la ecuación 4.1, que incluye:

- La *posición* del vehículo en el sistema de referencia del sub-mapa X_{rob} .
- La *orientación* del vehículo respecto del sistema del sub-mapa q_{rob} .
- La *velocidad lineal* del vehículo respecto del sistema del sub-mapa v_{rob} .
- La *velocidad angular* del vehículo respecto del sistema del sub-mapa ω .

$$X_v = \left. \begin{array}{c} \left. \begin{array}{c} x_{rob} \\ y_{rob} \\ z_{rob} \end{array} \right\} X_{rob} \\ \left. \begin{array}{c} q_0 \\ q_x \\ q_y \\ q_z \end{array} \right\} q_{rob} \\ \left. \begin{array}{c} v_x \\ v_y \\ v_z \end{array} \right\} v_{rob} \\ \left. \begin{array}{c} \omega_x \\ \omega_y \\ \omega_z \end{array} \right\} \omega \end{array} \right\} X_p \quad (4.1)$$

De todos los elementos del vector de estado, cabe destacar el uso de un vector de cuatro elementos, denominado *quaternion*, para describir la rotación del vehículo q_{rob} , es decir, uno más de los estrictamente necesarios para representar la orientación. Este vector define la rotación en

un ángulo de magnitud θ , alrededor de un eje definido por el vector unitario (u_x, u_y, u_z) . Por lo tanto, el vector de rotación del vehículo queda definido tal como se muestra en la ecuación 4.2

$$q_{rob} = \begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} = \begin{pmatrix} \cos \frac{\theta}{2} \\ u_x \sin \frac{\theta}{2} \\ u_y \sin \frac{\theta}{2} \\ u_z \sin \frac{\theta}{2} \end{pmatrix} \quad (4.2)$$

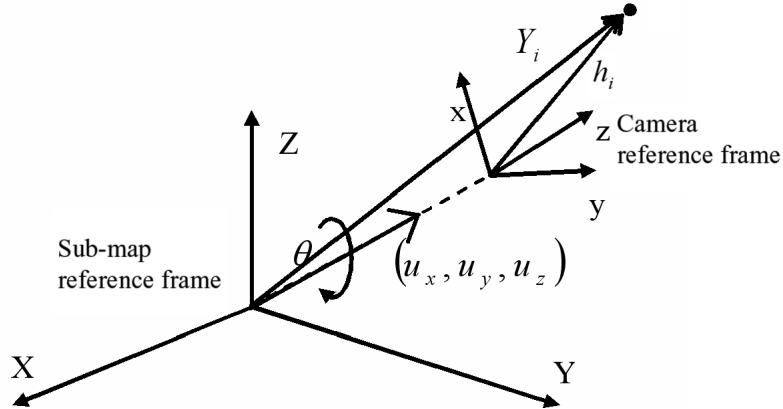


Figura 4.2: Representación del vector de rotación

La razón de usar este vector de cuatro componentes se basa en su simplicidad a la hora de realizar rotaciones concatenadas secuencialmente. Por otro lado, lógicamente existe una relación entre q_{rob} y R_{rob} , es decir, la matriz de rotación del vehículo. Esta relación se mostrará más adelante, en la ecuación 4.92.

El hecho de incluir también en el vector de estado la velocidad lineal y angular del vehículo es debido al *modelo de movimiento* elegido. Este modelo supone una velocidad constante en cada estado e impulsos de velocidad aleatorios entre cada estado. Para ello es necesario modelar también la velocidad a la hora de implementarla en el filtro, tal como se expone en el apartado 4.4.

En segundo lugar, se definen los *vectores de estado* estimados de las diferentes marcas usadas en el filtro Y_i . Estos vectores definen la posición de dichas marcas respecto del sistema del submapa, tal como se muestra en la ecuación 4.3, donde $h_i = \begin{pmatrix} h_{ix} \\ h_{iy} \\ h_{iz} \end{pmatrix}$ se corresponde con la posición de la marca medida, respecto del sistema de referencia de la cámara.

$$Y_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} \quad (4.3)$$

Por lo tanto, el vector de estado completo en el submapa, agrupando ambos elementos, se define como se muestra en 4.4, donde X es el mencionado vector de estado y P la matriz de covarianza.

$$X = \begin{pmatrix} X_v \\ Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} \quad P = \begin{pmatrix} P_{11} & P_{12} & P_{13} & \cdots \\ P_{21} & P_{22} & P_{23} & \cdots \\ P_{31} & P_{32} & P_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad (4.4)$$

4.3. Introducción al Filtro de Kalman Extendido (EKF)

El propósito del EKF en el presente trabajo consistirá en estimar la posición y orientación de la cámara en cada instante de tiempo. Dado que no se puede suponer una *función de próximo estado* $f(X)$ lineal, el EKF proporcionará la mencionada estimación linealizando $f(X)$ en cada instante de tiempo, siendo ésta la principal diferencia con respecto al filtro de Kalman estándar.

De esta manera, la implementación del filtro de Kalman se describe a continuación, donde k representa el índice de tiempo:

- a) *Etapa de predicción.* El primer paso del algoritmo consiste en predecir el vector de estado en el próximo instante de tiempo. Para ello se hace uso de la mencionada función del próximo estado $f(X)$, la cual será definida en el apartado 4.4. Así, las ecuaciones de predicción se definen en 4.5 y 4.6.

$$\hat{X}(k+1|k) = f(X(k|k)) \quad (4.5)$$

$$\hat{P}(k+1|k) = \frac{\partial f}{\partial X}(k|k) \cdot P(k|k) \cdot \left(\frac{\partial f}{\partial X}(k|k) \right)^T + Q(k) \quad (4.6)$$

A partir de la primera ecuación, se obtiene directamente la predicción del vector de estado para el próximo instante de tiempo ($K+1$). La segunda ecuación, por su parte, permite obtener la predicción de la matriz de covarianza en el próximo instante de tiempo. En 4.6, Q representa la *covarianza de ruido de proceso*, cuya obtención se explica en el apartado 4.4. De ella se deduce fácilmente que la predicción de la incertidumbre en el vector de estado tenderá a aumentar antes de la etapa de actualización.

- b) *Etapa de actualización.* Una vez completada la etapa de predicción, el siguiente paso consiste en realizar las medidas correspondientes y actualizar el filtro con la información obtenida. Para ello se aplicarán las ecuaciones 4.7 y 4.8.

$$\hat{X}(k+1|k+1) = \hat{X}(k+1|k) + W(k+1) \cdot \eta(k+1)_{tot} \quad (4.7)$$

$$P(k+1|k+1) = P(k+1|k) - W(k+1) \cdot S(k+1) \cdot (W(k+1))^T \quad (4.8)$$

De forma análoga a las ecuaciones de predicción, en este caso la ecuación 4.7 proporciona el valor del vector de estado, una vez actualizado con las medidas realizadas. En esta ecuación, η_{tot} es el *vector de innovación*, es decir la diferencia entre el vector de medida z_{tot} y el vector de predicción de dichas medidas h_{tot} . Para actualizar el vector de estado será necesario multiplicar η_{tot} por la matriz de ganancias W , la cual será definida en el apartado 4.8.5.

La ecuación 4.8 permite, en este caso, actualizar la matriz de covarianza del vector de estado total. Para ello, será necesario calcular S , que representa la *covarianza de ruido de medida* (covarianza del vector de innovación), y aplicar a ésta la matriz de ganancias W .

En este punto se actualiza el próximo instante de tiempo ($K+1$) como el instante actual, es decir, se avanza el filtro.

4.4. Modelo de Predicción o Movimiento.

En este apartado se define el modelo de predicción requerido para la implementación del EKF. Tal como se describe en el apartado 4.3, los dos elementos principales del EKF son el vector de estado X y su covarianza P . En la etapa de predicción el objetivo es estimar ambos elementos en el estado siguiente, para lo cual se hará uso del *modelo de predicción*.

En este caso el objeto a modelar es el movimiento de un vehículo del cual no se tiene información a priori. Es decir, no se dispone del conjunto de acciones \mathbf{u} que actúan sobre dicho objeto. En consecuencia, una forma intuitiva y sencilla de construir el modelo consiste en suponer una *velocidad constante* del vehículo (tanto lineal como angular) en cada estado con cambios de aceleración aleatorios entre estados.

Para predecir el vector de estado en el estado siguiente, se define la función $f_v(X_v(k|k))$ la cual, realmente, proporcionará como resultado la predicción del estado del vehículo $\hat{X}_v(k+1|k)$. Para modelar el paso de un estado a otro se supondrá un *cambio de aceleración aleatorio* gaussiano y de media 0. Es decir, se crea un vector aleatorio en el que se definen los cambios de velocidad (lineal y angular) y se le suma al vector de estado del vehículo para obtener la predicción en el siguiente estado. Dado que en principio las marcas se suponen estáticas respecto al entorno, la predicción de su posición coincidirá con la posición en el estado actual. De esta forma se puede definir la predicción del estado total según la ecuación 4.9, donde $\hat{Y}_1(k|k) = Y_1(k)$.

$$f = \hat{X}(k+1|k) = \begin{pmatrix} f_v(X_v(k|k)) \\ \hat{Y}_1(k|k) \\ \hat{Y}_2(k|k) \\ \vdots \end{pmatrix} \quad (4.9)$$

El modelo así obtenido es aplicable a una cámara con seis grados de libertad (6 DOF). Debido a las restricciones aplicables al modelo de movimiento de un vehículo frente a una cámara libre, se impone la necesidad de modificar el modelo de referencia. En primer lugar, se parte del modelo cinemático de un vehículo de cuatro ruedas, dos de ellas direccionales. En el sistema propuesto, la cámara, y por tanto el sistema de referencia de la misma, está localizada en el parabrisas, es decir, en la zona delantera superior del vehículo (ver Fig. 4.3). Por simplicidad del modelo, se asume que las cámaras están localizadas en la parte posterior del vehículo, justo sobre el eje trasero.

Teniendo en cuenta estas consideraciones, la velocidad lineal tanto en el eje X como en el Y del sistema de referencia del vehículo deben ser 0, tal como se muestra en 4.10, donde a y α representan la aceleración lineal y angular respectivamente.

Por otro lado si se asume que el vehículo circula por un terreno plano, las velocidades angulares ω_x y ω_z también deberán ser 0, tal como se muestra en 4.11.

$$V^0 = \begin{pmatrix} 0 \\ 0 \\ v_z^0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ a_z^0 \cdot \Delta t \end{pmatrix} \quad (4.10)$$

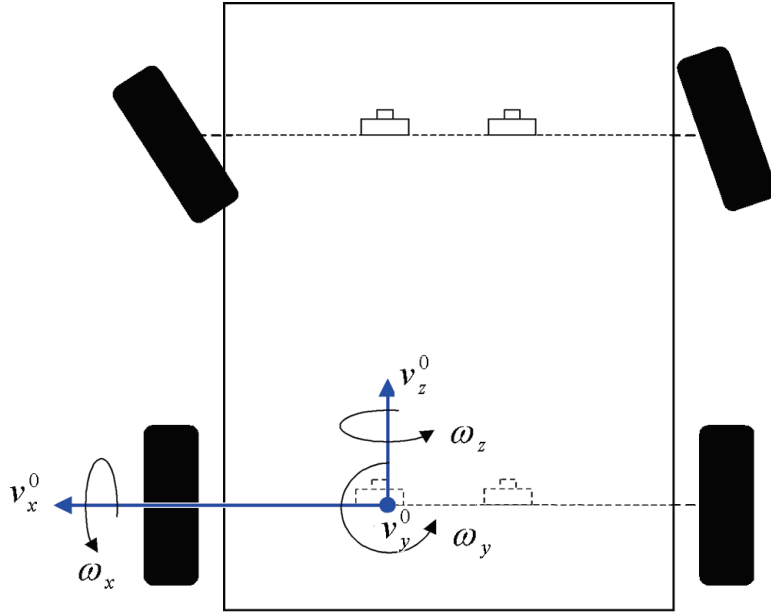


Figura 4.3: Modelo cinemático de un vehículo con cuatro ruedas, dos de ellas direccionales

$$\Omega = \begin{pmatrix} 0 \\ \omega_y \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ \alpha_y \cdot \Delta t \\ 0 \end{pmatrix} \quad (4.11)$$

Para obtener la predicción de la velocidad lineal en el próximo estado respecto a la referencia del submapa, se deben expresar las condiciones anteriormente expuestas en dicho sistema de referencia, como se muestra en 4.12.

$$V = R_{rob} \cdot V^0 + X_{rob} \quad (4.12)$$

Por lo tanto el vector aleatorio de velocidades mencionado anteriormente queda definido según 4.13.

$$n = \begin{pmatrix} V \\ \Omega \end{pmatrix} \quad (4.13)$$

Si se suponen a y α independientes se puede formar la matriz de covarianza del anterior vector como se muestra en 4.14.

$$P_n = \begin{pmatrix} \sigma_V^2 & 0 \\ 0 & \sigma_\Omega^2 \end{pmatrix} \quad (4.14)$$

Teniendo en cuenta las suposiciones anteriores, se puede obtener $f_v(X_v(k|k))$ según se muestra en 4.15, donde $q[(\omega + \Omega) \Delta t]$ representa la transformación del vector de velocidad angular en la forma de tres componentes a la forma de cuatro (q). v_{rob} y ω son la velocidad lineal y angular actuales del vehículo respectivamente.

$$f_v = \begin{pmatrix} X_{rob} + (v_{rob} + V) \Delta t \\ q_{rob} \times q [(\omega + \Omega) \Delta t] \\ v_{rob} + V \\ \omega + \Omega \end{pmatrix} \quad (4.15)$$

De cara a calcular la predicción de la covarianza total P , es necesario obtener el ruido de predicción Q_v . Para ello hay que transformar la covarianza del vector de velocidad aleatorio de la manera expresada en 4.16.

$$Q_v = \frac{\partial f_v}{\partial n} P_n \left(\frac{\partial f_v}{\partial n} \right)^T \quad (4.16)$$

Además, es necesario calcular el jacobiano $\frac{\partial f}{\partial X}$ y el ruido de predicción total Q . Para ello, en primer lugar se calculará $\frac{\partial f_v}{\partial X_v}$, con lo que el jacobiano total quedará de la forma expresada en 4.17.

$$\frac{\partial f}{\partial X} = \begin{pmatrix} \frac{\partial f_v}{\partial X_v} & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \quad (4.17)$$

De forma similar se puede formar Q como se muestra en 4.18.

$$Q = \begin{pmatrix} Q_v & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix} \quad (4.18)$$

Finalmente se puede calcular la predicción de P como indica la ecuación 4.19.

$$\hat{P}(k+1|k) = \left[\frac{\partial f}{\partial X} \cdot P(k|k) \cdot \left(\frac{\partial f}{\partial X} \right)^T \right] + Q \quad (4.19)$$

4.5. Modelado de las cámaras

Dado que el sensor principal del sistema consiste en una cámara estéreo, para poder definir un correcto modelo de medida previamente será necesario modelar dicha cámara.

La Fig. 4.4 representa el escenario de la cámara estéreo enmarcada en la denominada *Geometría epipolar*, en el que se aprecian las dos cámaras (izquierda y derecha), una marca i y el sistema de referencia del sub-mapa.

Para modelar la cámara se necesita obtener tanto sus parámetros *intrínsecos* como *extrínsecos*. Los primeros caracterizan los parámetros geométricos y ópticos de cada cámara individual:

- $FC1$ y $FC2$: Distancias focales de una de las cámaras individuales.
- $CC1$ y $CC2$: Coordenadas del punto principal de una de las cámaras individuales.

Los segundos describen las relaciones geométricas entre cada cámara y el mundo exterior, y también describen las relaciones geométricas entre ambas cámaras.

- R_{int} : Matriz de rotación entre cámaras
- T_{int} : Matriz de translación entre cámaras.

La proyección de la marca i en el plano de imagen de la cámara izquierda (u_{LS}, v_{LS}) queda definida por la intersección del vector h_i con este mismo plano. De igual forma se puede obtener la proyección en la cámara derecha a partir del vector h_{iR} . Ambos pares de coordenadas vienen definidas en unidades de píxel. Por lo tanto, para obtener (u_{LS}, v_{LS}) y (u_{RS}, v_{RS}) a partir de h_i y h_{iR} basta con realizar dicha proyección en base a los parámetros intrínsecos de cada cámara según se indica en 4.20 y 4.30 (ver trabajo [54]).

$$\begin{pmatrix} su_{LS} \\ sv_{LS} \\ s \end{pmatrix} = \begin{pmatrix} FC1_L & 0 & CC1_L & 0 \\ 0 & FC2_L & CC2_L & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} h_{ix} \\ h_{iy} \\ h_{iz} \\ 1 \end{pmatrix} \quad (4.20)$$

$$\begin{pmatrix} su_{RS} \\ sv_{RS} \\ s \end{pmatrix} = \begin{pmatrix} FC1_R & 0 & CC1_R & 0 \\ 0 & FC2_R & CC2_R & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} h_{iRx} \\ h_{iRy} \\ h_{iRz} \\ 1 \end{pmatrix} \quad (4.21)$$

Por otro lado si la información de partida es el vector de posición absoluto de la marca Y_i , antes de realizar la proyección será necesario transformar las coordenadas de la marca en el sistema del sub-mapa al sistema de la cámara. Para ello, será necesaria la construcción de sendas matrices M_L y M_R , denominadas matrices de *proyección perspectiva*, definidas en las ecuaciones 4.81 y 4.83, y en las que se engloban la transformación (translación + rotación) y la proyección explicada anteriormente. Este proceso será tratado en el apartado 4.9.2.

Otro aspecto a tener en cuenta es el denominado *plano epipolar* delimitado por los vértices formados por la marca en cuestión y los centros de los dos sistemas de referencia de las cámaras. Dicho plano intersecta a los dos planos de imagen en las denominadas *rectas epipolares*. La principal propiedad de estas rectas es que, dada una marca en el espacio tridimensional, sus proyecciones en ambos planos de imagen deberán pertenecer a dichas rectas. Por lo tanto, dada una marca y su proyección en una de las cámaras, se debe realizar la búsqueda de su correspondiente proyección en la otra cámara sobre la recta epipolar de dicha cámara. Este es el principio utilizado en la llamada *búsqueda epipolar* (ver apartado 4.9.2).

A la hora de escoger la configuración más adecuada del par de cámaras se han tenido en cuenta los siguientes criterios:

4.5.1. Resolución de las cámaras

Teniendo en cuenta que el procesado completo debe realizarse en tiempo real, cuanto mayor sea la resolución de las imágenes a procesar mayor será el tiempo empleado, tanto en su captura

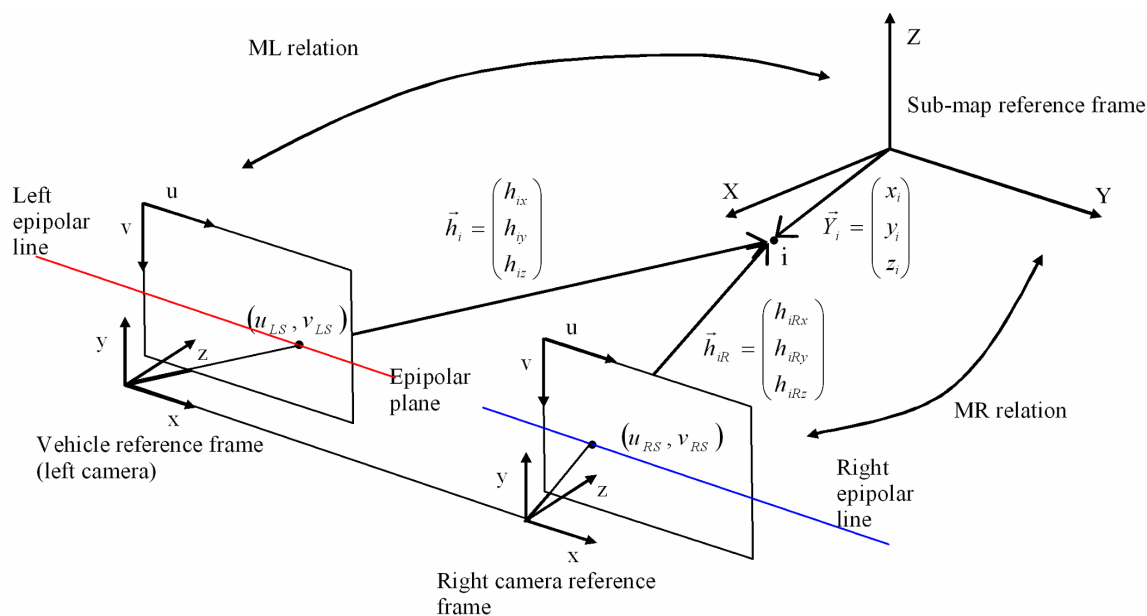


Figura 4.4: Representación de la geometría epipolar 3D y nomenclatura utilizada

como en el posterior procesado. Por otro lado, tal como se explica en 4.5.2, el error en la estimación de la posición de las marcas se reduce a medida que la resolución aumenta, ya que el tamaño del píxel disminuye. Por lo tanto, se deberá encontrar una solución de compromiso que se ajuste fundamentalmente a los requerimientos de tiempo del sistema. El estudio se centrará en cuatro resoluciones posibles, disponibles comercialmente.

En la gráfica mostrada en la Fig. 4.5 se muestran los tiempos de cómputo medios y sus varianzas para diferentes resoluciones empleadas, obtenidos en las pruebas realizadas al sistema. La mejor resolución que permite un funcionamiento en tiempo real, teniendo en cuenta la capacidad de procesamiento empleada, es 320x240. En el caso presentado en esta Tesis, el límite de tiempo real se considerará 33 ms.

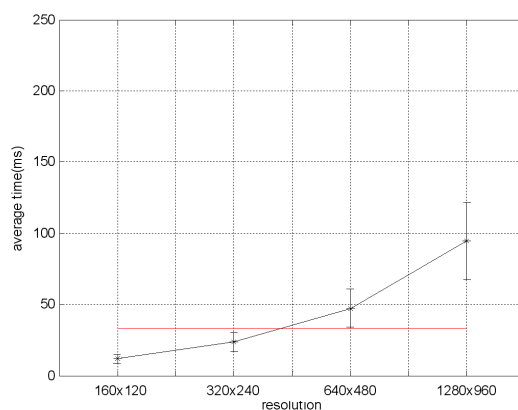


Figura 4.5: Tiempos de cómputo medios y sus varianzas para diferentes resoluciones de imagen. La línea horizontal marca el límite de tiempo real

4.5.2. Separación entre cámaras

Si se simplifica el problema de la geometría epipolar a dos dimensiones, se puede representar como se muestra en la Fig. 4.6.

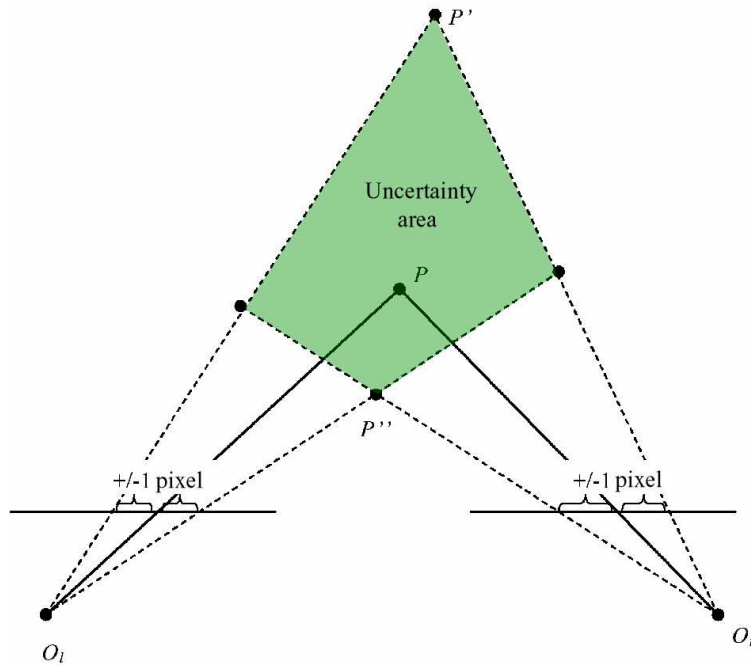


Figura 4.6: Efecto de la incertidumbre en la determinación de la posición real de la marca P a consecuencia del error de ± 1 píxel en las coordenadas de proyección. El efecto del error de 1 píxel en una marca más lejana P' provoca mayor error de profundidad que en una marca más cercana P'' .

Suponiendo como única fuente de error en la determinación de la profundidad h_{iz} la incertidumbre en la determinación de la proyección de la marca de ± 1 píxel, se puede apreciar en la Fig. 4.6 cómo a medida que dicha marca P se encuentra más alejada el error en la determinación de su posición es mayor.

Para evaluar cómo afecta al error mencionado la distancia entre cámaras (*baseline*) se asumen las siguientes simplificaciones:

- Ambas cámaras se encuentran orientadas de forma paralela, es decir, enfocando en la misma dirección.
- Ambas cámaras están separadas una distancia T_x a lo largo del eje X , existiendo un desplazamiento nulo en el resto de los ejes.
- Ambas cámaras se asumen intrínsecamente iguales, es decir, tanto la distancia focal como el punto principal son iguales.

Bajo estas condiciones, se puede obtener la profundidad h_{iz} de la marca a partir de las coordenadas de proyección horizontales de ambas imágenes tal como se expresa en 4.22.

$$u_{LS} = CC1_L - FC1_L \frac{h_{ix}}{h_{iz}}; \quad u_{RS} = CC1_R - FC1_R \frac{h_{ix} - T_x}{h_{iz}} \quad (4.22)$$

Aplicando la suposición de que ambas cámaras son iguales, se puede obtener fácilmente la profundidad en función de la separación entre cámaras T_x , la distancia focal de las cámaras FC y la disparidad de las coordenadas horizontales $d = u_L - u_R$, tal como se muestra en 4.23.

$$h_{iz} = \frac{FC1 \cdot T_x}{d} \quad (4.23)$$

A partir de esta ecuación, haciendo variar la disparidad en ± 1 píxel se obtiene el error correspondiente para diferentes separaciones entre cámaras, según la distancia a la que se encuentre la marca, tal como se muestra en la gráfica de la Fig. 4.7

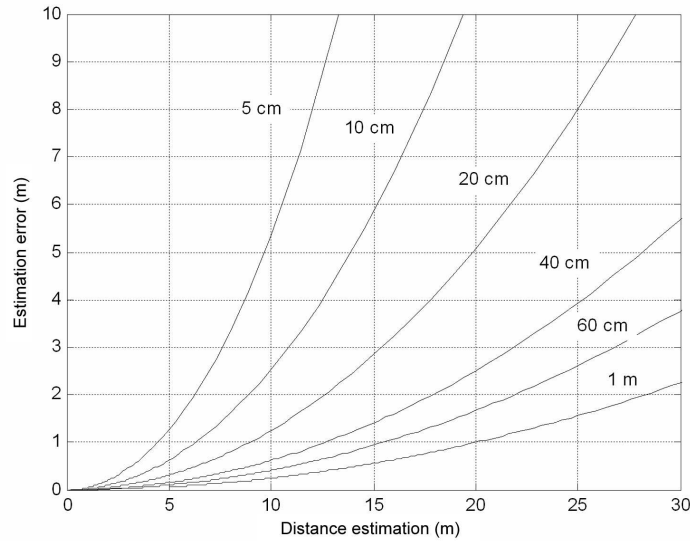


Figura 4.7: Error en la determinación de la profundidad en función de la separación entre cámaras T_x y la distancia de la marca

De los datos presentados en la Fig. 4.7 se deduce que cuanto mayor sea la separación entre cámaras, menor será el error en la determinación de la profundidad. Sin embargo, a medida que la separación entre cámaras es mayor, la *zona muerta* en la que no existe visibilidad para ninguna de las cámaras (ver Fig. 4.8(a)) o para, al menos, una de ellas (ver Fig. 4.8(b)) aumenta. Esto provoca que marcas demasiado cercanas al vehículo no sean capturadas correctamente.

Como solución de compromiso se escogerá una separación entre cámaras en un intervalo $T_x = [30, 40]cm$, ya que permite una visibilidad en perspectiva suficientemente cercana, manteniendo un error de estimación de distancia próximo a la máxima separación evaluada.

4.5.3. Apertura de las lentes

Tal como se explicó anteriormente, la cámara estéreo se ubica en la parte superior del salpicadero del vehículo mirando hacia el frente. En [27] y [26] se presentan sendos sistemas de SLAM visual en los que la orientación de la cámara es transversal al sentido de la marcha. En el sistema propuesto, esto no es posible ya que el algoritmo de detección de lugares previsitados requiere que la apariencia de la imagen no varíe demasiado desde la primera visita a la siguiente. Esto no sucede con la orientación transversal de las cámaras, tal como se explica más adelante, al ser muy reducido el campo de visión asociado a un entorno demasiado cercano.

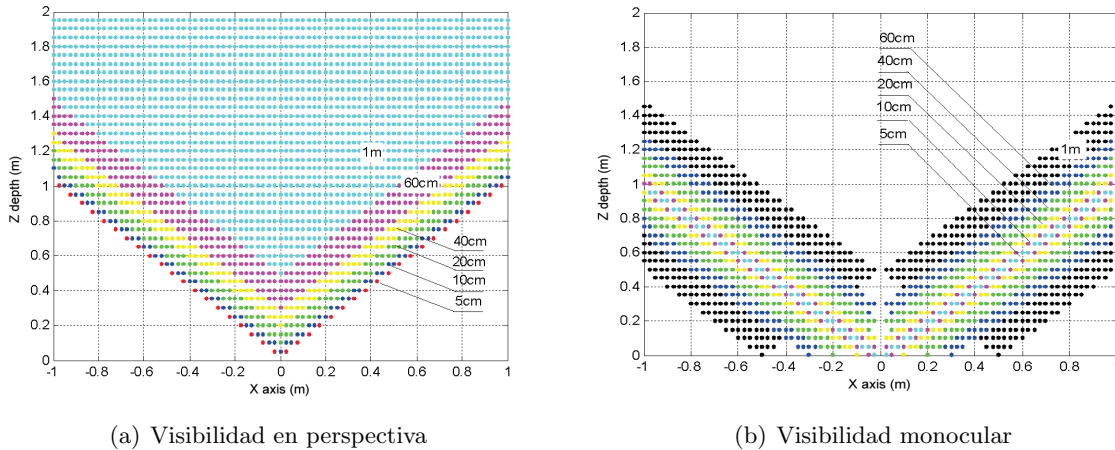


Figura 4.8: Zonas de visibilidad perspectiva y monocular en función de la separación entre cámaras. El origen de coordenadas, en este estudio, se encuentra situado en el punto medio entre cámaras.

Dada la configuración de las cámaras propuesta, el movimiento de avance del vehículo se produce en la misma dirección que el punto de vista de las mismas. En la Fig. 4.9 se puede apreciar la distribución de las marcas en entornos típicamente urbanos. Éstas se encuentran situadas generalmente en edificios circundantes, árboles, farolas, etc. Dado que el vehículo circula por calles, carreteras, etc., estas marcas se encontrarán generalmente situadas de forma lateral al sentido de la marcha. Por lo tanto, las marcas cuya proyección queda situada en el centro de la imagen izquierda o derecha estarán localizadas a una mayor distancia que las que su proyección recae sobre los laterales de la imagen. En la Fig. 4.9 se muestran las aperturas horizontales de las cámaras sobre el escenario explicado anteriormente. Se puede observar que la marca P2, que es la más cercana, queda fuera del campo visual de la cámara izquierda. Dado que las marcas más cercanas poseen un error de estimación menor, cuanto mayor sea el ángulo de apertura mayor será el número de marcas estimadas de forma precisa.

Por otro lado, el hecho de tener un campo visual grande permite tener marcas visuales dispersas, reduciendo el número de marcas necesarias para localizar correctamente el vehículo y mejorando por tanto el tiempo de cómputo, tal como se mostrará en el capítulo 6. En la Fig. 4.10. se muestra un ejemplo del número de marcas capturadas usando una lente estándar con un ángulo de apertura horizontal $\alpha_h = 42,25^\circ$ y una lente de gran angular con un ángulo de apertura horizontal $\alpha_h = 107^\circ$. Estos datos son generalizables a cualquier otro tipo de recorrido.

Teniendo en cuenta el modelo comercial de cámara elegido, se ha buscado la lente de mayor ángulo de apertura, esto es, $\alpha_h = 107^\circ$.

4.6. Calibración de las cámaras

El primer paso para poder utilizar la cámara estéreo es poder relacionar las coordenadas de proyección 2D con las correspondientes 3D en el espacio. Para esto es necesario caracterizarla, es decir, obtener tanto sus parámetros *intrínsecos* como *extrínsecos*.

Existe numerosa literatura relacionada con el proceso de calibración de cámaras [55], [56], [57], [58], [59]. Para realizar la calibración se ha utilizado el Camera Calibration Toolbox para Matlab (ver [60]), el cual está basado en [55].

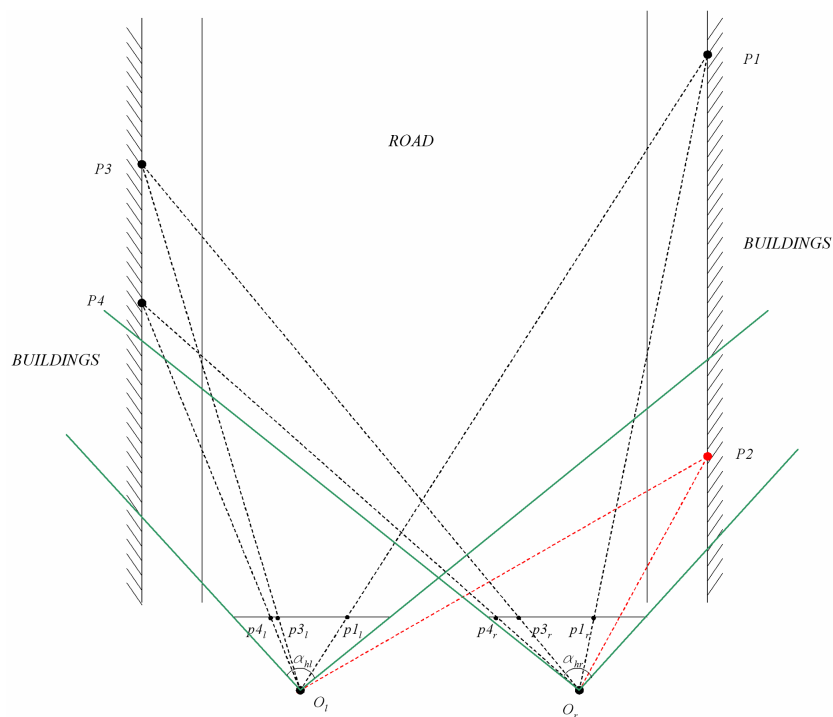


Figura 4.9: Distancia mínima de captura de marcas en función del ángulo de apertura. La marca $P2$ no es capturada, debido a que su proyección cae fuera del campo visual de la cámara izquierda.

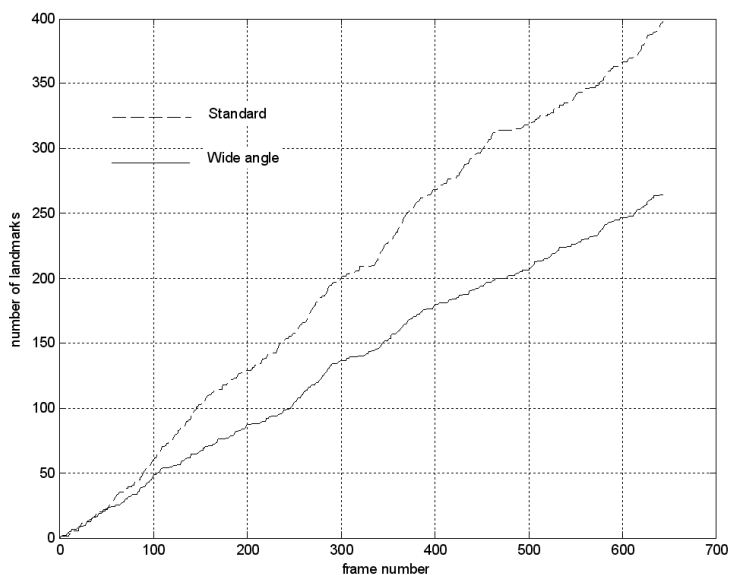
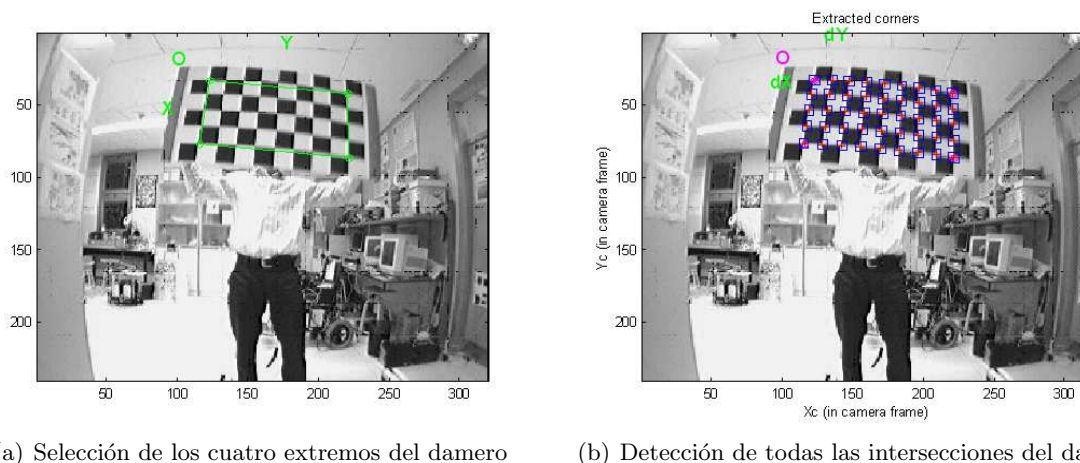


Figura 4.10: Número de marcas capturadas por el sistema a lo largo del recorrido. Para la prueba se utilizó un recorrido de prueba consistente en una rotación de 360° de la cámara sobre su propio eje.

Como se explicó anteriormente, el objetivo general del proceso de calibración es estimar tanto los parámetros intrínsecos, como los extrínsecos. Esta estimación deberá realizarse de forma óptima basándose en las observaciones de un objeto 3D conocido. Este objeto consiste en un damero de tamaño conocido y cuyos cuadros son de tamaño también conocido. De este panel se extraen los puntos 3D coincidentes con las intersecciones de los cuadros. Este proceso se realiza en una etapa previa y de forma semiautomática, es decir, que se deben introducir manualmente las posiciones de las esquinas de los bordes exteriores del damero. El sistema detecta a partir de esta información el resto de intersecciones, tal como se muestra en la Fig. 4.11.

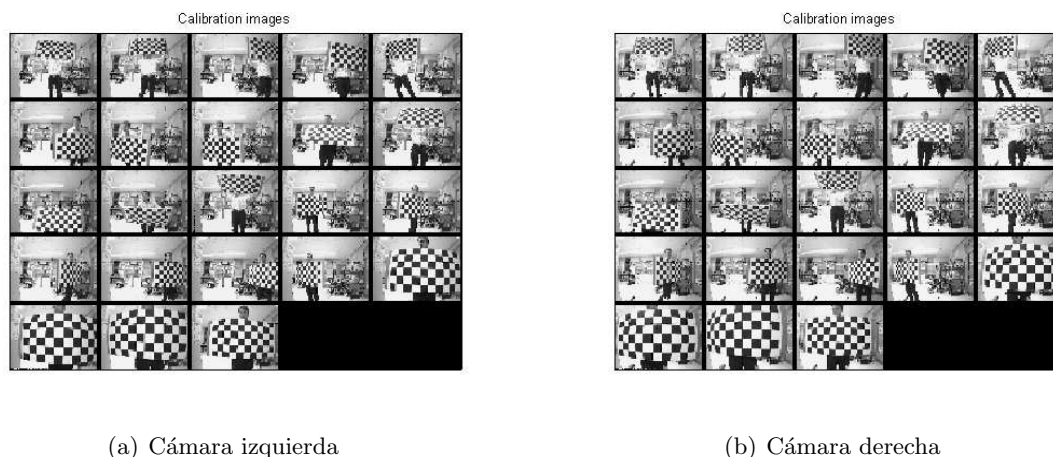


(a) Selección de los cuatro extremos del damero

(b) Detección de todas las intersecciones del damero

Figura 4.11: Proceso de selección de puntos de calibrado a partir de un patrón damero (cámara izquierda)

El mencionado proceso se repite exactamente igual para diferentes imágenes en las que el patrón se ubica a diferentes distancias, posiciones y orientaciones (ver Fig. 4.12).



(a) Cámara izquierda

(b) Cámara derecha

Figura 4.12: Imágenes de calibración de ambas cámaras

Una vez obtenidos todos los puntos comienza el proceso de calibración, el cual consiste básicamente en minimizar una función de error inicialmente lineal (etapa de modelo Pin-Hole) y, posteriormente aplicar los modelos no lineales de distorsión de lentes. Este método, descrito en [55], consiste en usar un nuevo modelo implícito que interpola los puntos correctos de la imagen basándose en los parámetros de la cámara obtenidos en pasos anteriores del algoritmo.

Tras realizar el proceso de calibración para cada una de las cámaras se procede a la calibración del sistema estéreo en conjunto, obteniendo los parámetros extrínsecos entre cámaras y optimizando los intrínsecos de manera conjunta. En la Fig. 4.13 se muestran los resultados geométricos de la calibración junto con las estimaciones 3D del patrón en sus diferentes posiciones.

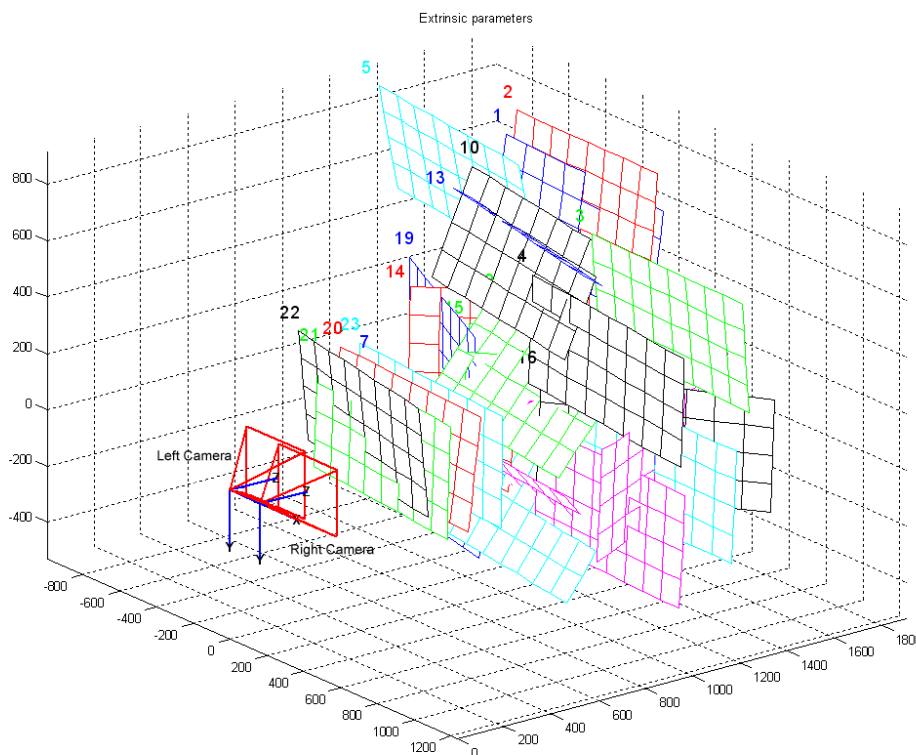


Figura 4.13: Visualización de los resultados de calibración y las diferentes muestras del patrón empleadas en el proceso

4.6.1. Modelo de distorsión

Si los componentes ópticos de la cámara fueran ideales la transformación entre los puntos tridimensionales en el espacio y sus proyecciones en la imagen podrían modelarse linealmente con el modelo Pin-Hole.

En una lente real el modelo Pin-hole no refleja exactamente el comportamiento de la misma debido a que ésta desvía los haces de forma que las proyecciones de los puntos en el espacio hacia el plano de imagen no coinciden con las esperadas. La Fig. 4.14 muestra cómo un punto, cuya proyección teórica recae en \tilde{q} , por efecto de la distorsión de la lente finalmente queda ubicado en q .

Dos de los modelos más usados para predecir el efecto mencionado anteriormente son los denominados modelos de *distorsión radial* y *distorsión tangencial*. Éstos se fundamentan en dividir el desplazamiento entre la proyección real y la ideal en dos componentes.

La primera de ellas contempla el desplazamiento producido en la dirección radial desde el punto principal, determinado por $\delta u^{(r)}$ y $\delta v^{(r)}$. En el trabajo [55] se describe la obtención de

dicho desplazamiento en función de las coordenadas de proyección sin distorsión. Esta relación se muestra en 4.24.

$$\begin{aligned}\delta u^{(r)} &= \tilde{u}(K_1 r^2 + K_2 r^4 + \dots) \\ \delta v^{(r)} &= \tilde{v}(K_1 r^2 + K_2 r^4 + \dots)\end{aligned}\quad (4.24)$$

La segunda de ellas refleja el desplazamiento de forma tangencial al punto original y está determinado por $\delta u^{(t)}$ y $\delta v^{(t)}$. El cálculo de esta componente de desplazamiento queda reflejado en 4.25.

$$\begin{aligned}\delta u^{(t)} &= 2P_1 \tilde{u} \tilde{v} + P_2 (r^2 + 2\tilde{u}) \\ \delta v^{(t)} &= P_1 (r^2 + 2\tilde{v}) + 2P_2 \tilde{u} \tilde{v}\end{aligned}\quad (4.25)$$

Las constantes K_i son los coeficientes de distorsión radial, las constantes P_i representan los coeficientes de distorsión tangencial, que se añaden al conjunto de parámetros intrínsecos, y $r = \sqrt{\tilde{u}^2 + \tilde{v}^2}$.

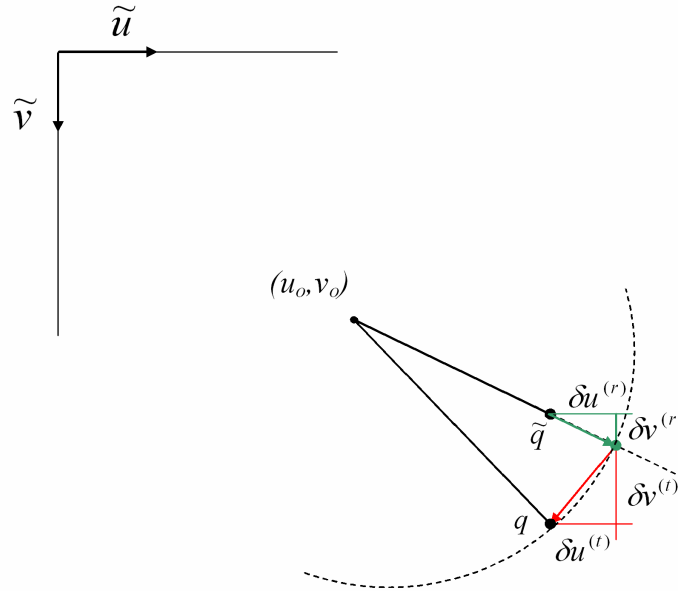


Figura 4.14: Efecto de la distorsión radial y tangencial sobre la proyección de un punto en el plano de imagen.

Finalmente, las coordenadas métricas, una vez aplicada la corrección de la distorsión, se pueden obtener agrupando las dos ecuaciones anteriores tal como se indica en 4.26.

$$\begin{aligned}u_m &= \tilde{u} + \delta u^{(r)} + \delta u^{(t)} \\ v_m &= \tilde{v} + \delta v^{(r)} + \delta v^{(t)}\end{aligned}\quad (4.26)$$

Estos modelos de distorsión son más necesarios cuanto mayor sea el ángulo de apertura de las lentes. Además, cuanto mayor sea el número de coeficientes de distorsión radial empleado mayor será la precisión a la hora de modelar dicha distorsión.

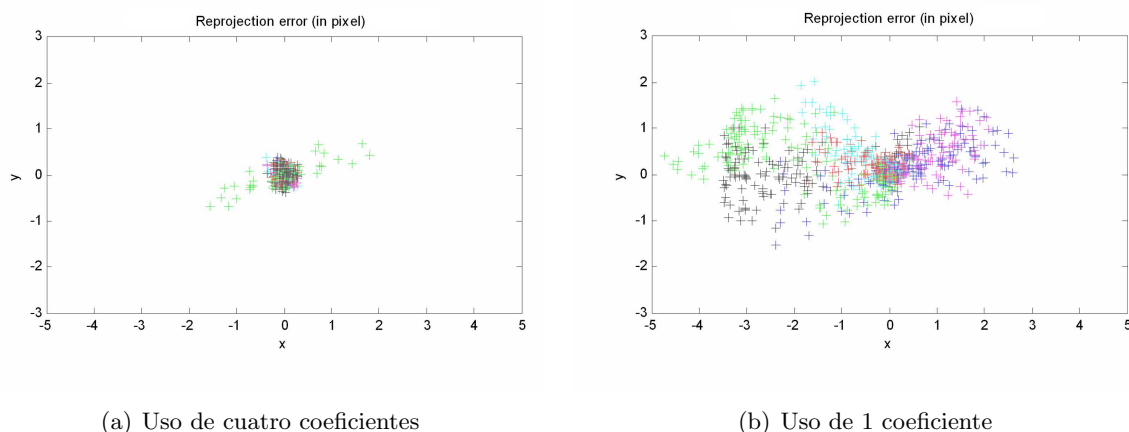
En la versión monocular de SLAM visual presentada en [61] ya se justifica el uso de lentes de gran angular debido a la reducción de la incertidumbre de las marcas capturadas. En este

trabajo se presenta un modelo de distorsión sencillo basado en una aproximación con una única constante de distorsión.

Para comparar el uso de un único coeficiente frente al uso de dos coeficientes radiales y dos tangenciales se ha analizado el error de reproyección sobre las imágenes de calibración. Es decir, tomando las coordenadas 3D de los puntos de calibración del patrón, se han calculado las coordenadas de proyección de cada uno de los puntos y se han comparado con las introducidas manualmente al inicio del proceso.

En primer lugar se han obtenido los errores asociados al uso de cuatro coeficientes. Los valores usados para éstos son los siguientes: $K_1 = -0,31307$, $K_2 = 0,09325$, $P_1 = -0,0002$, $P_2 = 0$. Los errores obtenidos se muestran en la Fig. 4.15(a). En segundo lugar, para el caso de un único coeficiente, el valor obtenido fue $K_1 = -0,21982$ y los errores asociados se muestran en la Fig. 4.15(b).

Como se puede apreciar, la dispersión de los datos muestra un error medio notablemente mayor para el caso de usar un único coeficiente. El motivo de esta clara diferencia es el uso de lentes de gran angular, ya que con lentes con un campo de visión más estrecho esta diferencia es mucho menos apreciable.



(a) Uso de cuatro coeficientes

(b) Uso de 1 coeficiente

Figura 4.15: Errores de reproyección (en coordenadas de píxel) para el caso de usar cuatro coeficientes y para el caso de usar un único coeficiente. Cada color indica una imagen del patrón diferente

Como conclusión, en el método propuesto en esta Tesis se han empleado dos coeficientes para la distorsión radial y dos coeficientes para la tangencial, mejorando el modelo de distorsión propuesto por A. Davison en [61].

4.7. Corrección de la distorsión

Antes de detallar tanto el *proceso de medida* como el de *inicialización de nuevas marcas*, es necesario analizar las diferentes alternativas a la hora de tener en cuenta el efecto de la distorsión de las cámaras. Como se verá más adelante, este apartado resume gráficamente los dos procesos fundamentales del SLAM de bajo nivel, relacionados con el sistema de visión.

Existen fundamentalmente dos alternativas a la hora de aplicar la corrección de la distorsión en el sistema, según se muestra en el diagrama de la Fig. 4.16:

- Corregir la distorsión en todos los píxeles de la imagen. En este caso, las coordenadas de proyección obtenidas previamente usando el modelo Pin-hole son directamente aplicables, ya que el desplazamiento debido a la distorsión va implícito en la imagen.
- Aplicar la corrección de la distorsión a las coordenadas de proyección de cada marca individualmente.

A la hora de comparar ambas alternativas el estudio se centrará en los tiempos de cómputo empleados en el proceso de medida y detección de nuevas marcas para ambos casos.

A priori, el hecho de corregir la posición de todos los píxeles de la imagen conlleva una carga computacional mucho mayor que el proceso de corrección aplicado únicamente a las coordenadas de las marcas.

Fijándose en el diagrama del proceso de inicialización de marcas (ver Fig. 4.16(a)), se aprecian cambios en caso de usar una alternativa u otra. En caso de usar la corrección de imagen completa se debe añadir la etapa de esta corrección, mientras que no es necesario aplicar correcciones a las coordenadas obtenidas con el modelo Pin-hole. Por otro lado, en la segunda alternativa, el hecho de no eliminar la distorsión en la imagen implica que la región de búsqueda epipolar en la imagen derecha no se deba realizar sobre una recta, sino sobre su equivalente distorsionada. Para resolver el problema de una forma eficiente se propone buscar a lo largo de la recta epipolar original, extendiendo la búsqueda en un determinado margen entorno a la misma. Esta ampliación implica un aumento del coste computacional asociado a la búsqueda epipolar.

Respecto al proceso de medida (ver Fig. 4.16(b)) también se aprecian cambios, fundamentalmente asociados a la eliminación de las etapas de corrección de las coordenadas para el caso de usar la corrección de imagen completa.

A la hora de implementar la alternativa de corrección de la imagen completa, para reducir el coste computacional de la tarea se propone usar el método presentado en [62]. Básicamente, este método consiste en precalcular las posiciones de los píxeles de la imagen una vez aplicada la corrección, y posteriormente almacenarlas en una tabla denominada LUT (*Look-up table*). A la hora aplicar la corrección a una imagen concreta sólo será necesario buscar las nuevas posiciones de los píxeles en la LUT y posteriormente aplicar un proceso de interpolación.

En la Fig. 4.17 se muestra una comparativa de los tiempos de cómputo por tareas, en función del método de corrección empleado. En la Fig. 4.17(a) se muestran los tiempos medios empleados corrigiendo la distorsión de toda la imagen directamente. En la Fig. 4.17(b) se muestra los tiempos aplicando la corrección completa, pero utilizando el método LUT en este caso. Por último, en la Fig. 4.17(c) se muestran los resultados obtenidos corrigiendo únicamente las coordenadas de proyección de las marcas.

En los dos primeros casos se aprecia un tiempo de lectura + corrección claramente superior al empleado en el último caso debido a la eliminación de la etapa de corrección. Por otra parte, debido al incremento del área de búsqueda epipolar, se aprecia un aumento de tiempo para esta tarea empleando únicamente la corrección en las marcas respecto a la de toda la imagen. Como observación final, los tiempos de cómputo totales, empleando corrección únicamente en las marcas, son claramente inferiores a los obtenidos corrigiendo la imagen completa, incluso empleando el método LUT.

Existen tres razones adicionales por las que en esta Tesis se opta por el segundo método:

- En primer lugar, como consecuencia de mantener la misma resolución tras la corrección de la distorsión en toda la imagen, se pasa de tener un ángulo de apertura muy alto a

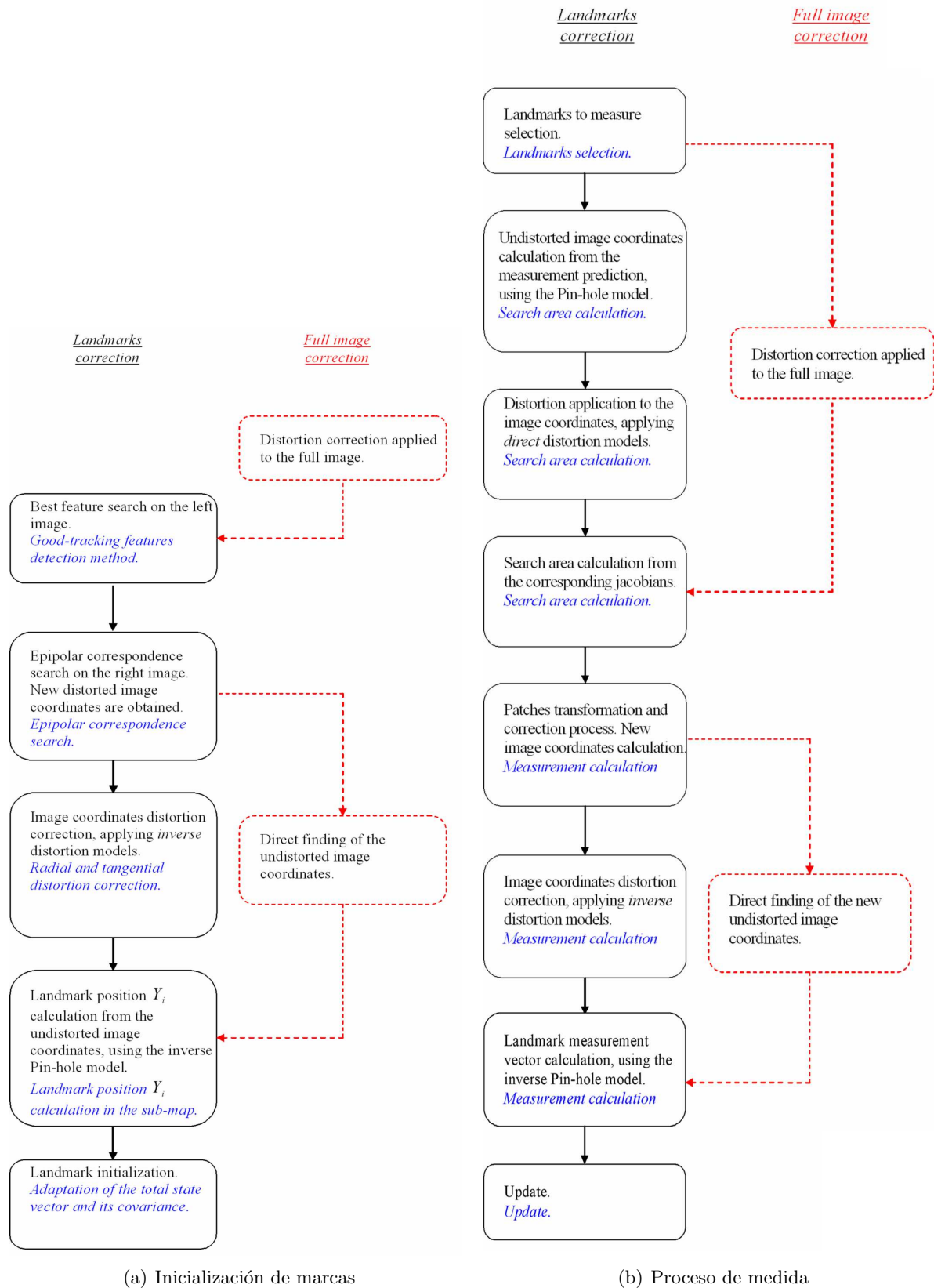
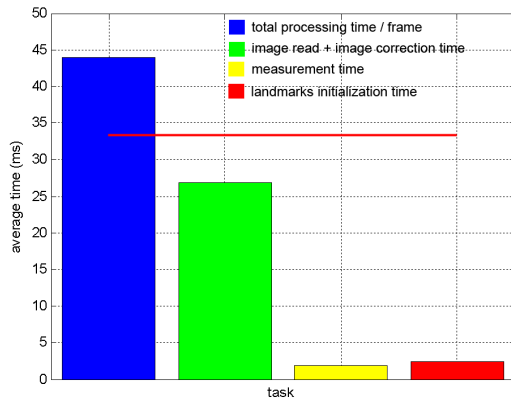
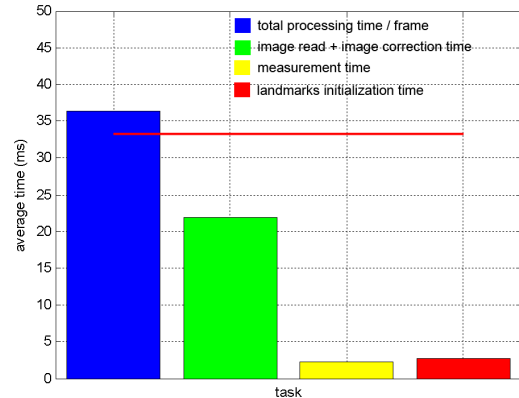


Figura 4.16: Diagrama general para los procesos de inicialización de marcas y medida de las mismas. En cursiva se muestra el apartado de la Tesis asociado a la tarea.

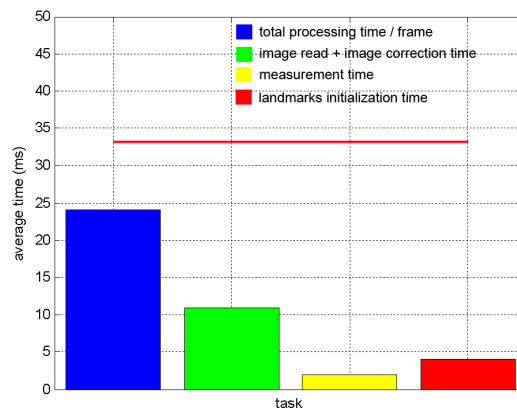
tener uno mucho más reducido. Esto implica la pérdida de las ventajas asociadas a tener un gran campo visual (ver 4.5.3).



(a) Corrección completa



(b) Corrección usando LUT



(c) Corrección sólo de marcas

Figura 4.17: Tiempos de cómputo promedio para las diferentes alternativas. Se muestra el tiempo de proceso total en cada frame (azul), tiempo dedicado a lectura de imagen y/o corrección de la misma (verde), tiempo empleado en medidas (amarillo) y tiempo dedicado a la inicialización de nuevas marcas (rojo)

- En segundo lugar, como se explicó anteriormente, el proceso de corrección implica una interpolación de píxeles, la cual es tanto más acusada cuanto más cerca del borde está cada píxel. En estas regiones, la corrección de la distorsión de la apariencia de las marcas no es buena, impidiendo en un gran número de casos una correcta correlación epipolar entre las dos imágenes.
- Por último, dado el reducido tamaño de los *patches* y la proximidad entre éstos, la similitud en la apariencia de éstos sigue siendo alta, a pesar de no eliminar la distorsión provocada por la lente. Se denomina patch a una porción de la imagen que identifica a una marca determinada.

Por lo tanto, el sistema elegido para el sistema propuesto en esta Tesis será este último método de corrección de las marcas.

4.8. Modelo de medida

El modelo de medida está basado en la obtención de las coordenadas 3D de posición de determinadas marcas. Para ello, es necesario obtenerlas en tiempo real a partir de la suposición de la posición del vehículo inicialmente. Todo ello se explicará en el apartado 4.9.

4.8.1. Selección de marcas

El primer paso a la hora de realizar la etapa de medida es la selección de marcas a medir. Dicha elección se llevará a cabo en base a los siguientes dos criterios:

1. *Criterio de visibilidad*: De entre todas las marcas capturadas previamente serán descartadas todas aquellas que no sean *visibles*. Para superar el criterio de visibilidad, la marca deberá cumplir las siguientes tres condiciones:
 - 1.1. La proyección de la marca en ambas cámaras (u_L, v_L) y (u_R, v_R) no deberá exceder el campo de visión. Es decir, estas coordenadas deberán estar contenidas en su correspondiente cuadro de imagen menos un margen igual al tamaño del *patch*, tal como se muestra en la Fig. 4.18. Por otra parte, dada la ubicación de las cámaras en el parabrisas y el amplio campo de visión obtenido por la lente de gran angular, parte del salpicadero del vehículo es capturado permanentemente. Para evitar la captura de marcas erróneas pertenecientes al salpicadero, este área será eliminada también de la búsqueda.

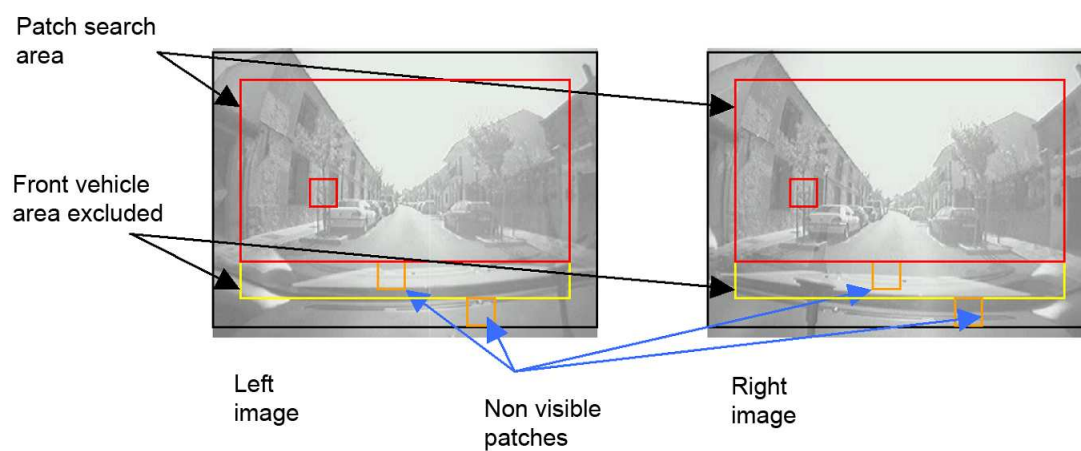


Figura 4.18: Representación de la zona de selección de marcas

- 1.2. El cambio de ángulo del punto de vista no debe exceder un cierto límite. Es decir, el ángulo formado por el vector de medida tomado cuando la marca fue inicializada y el vector de medida actual no debe exceder un valor (normalmente 45°). Esta restricción es necesaria debido al cambio de apariencia relativa al observar una imagen desde diferentes ángulos, lo cual terminaría provocando un fallo a la hora de realizar la correlación (ver apartado 4.8.3).

El cálculo de dicho ángulo se realiza según 4.27, donde h_{iorig} es el vector de observación en el momento de inicializar la marca, obtenido a partir de la pose del vehículo en dicho momento X_{porig} .

$$\beta = \cos^{-1} \left(\frac{h_i \cdot h_{iorig}}{|h_i| |h_{iorig}|} \right) \quad (4.27)$$

1.3. Por último, la distancia del vehículo a la marca puede exceder o ser inferior a la original hasta un cierto límite, es decir, el cociente entre el módulo del vector de medida tomado cuando la marca fue inicializada y el módulo del vector de medida actual debe estar entre un valor mínimo y máximo (normalmente 5/7 y 7/5, tal como se establece en [7]). La razón es similar a la anterior condición, es decir, la apariencia de la marca es diferente según la distancia desde la que se observe. Dicho cociente se calcula como muestra la ecuación 4.28.

$$T = \frac{|h_i|}{|h_{iorig}|} \quad (4.28)$$

Gráficamente se muestran las dos últimas condiciones en la Fig. 4.19.

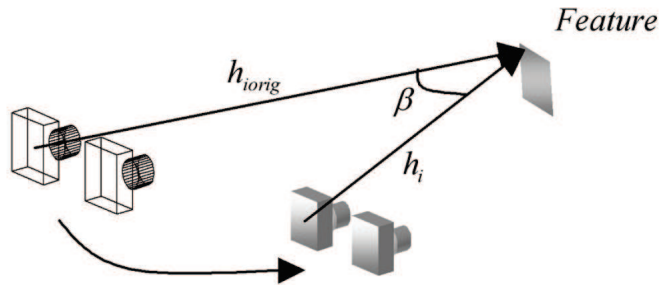


Figura 4.19: Vector de medida original h_{iorig} y actual h_i . Representación de los criterios de visibilidad

2. *Criterio de validez*: Una vez escogidas aquellas marcas que pasaron el criterio de visibilidad, el siguiente *filtro* consiste en no seleccionar aquellas marcas cuyas medidas hayan sido fallidas más de la mitad de los intentos realizados. Se entiende por marcas cuya medida ha resultado fallida aquellas cuya correlación con su imagen en el estado anterior no resultara suficientemente buena (ver apartado 4.8.3). Las marcas que no superen esta condición serán eliminadas. La razón es mantener, a lo largo del tiempo, sólo las marcas cuya apariencia sea relativamente constante y así evitar problemas de reflejos, obstrucciones, etc.

En lo que concierne a la implementación, los patches correspondientes a cada marca llevan asociados un código de colores para identificar su propio estado. Este código es el siguiente:

- **Rojo**: Indica que la marca es visible y fue correctamente medida en la iteración anterior.
- **Azul**: Indica que la marca es visible pero no fue correctamente medida en la iteración anterior.
- **Amarillo**: Indica que la marca no es visible.

Una vez elegidas las marcas a medir, se procederá a obtener la predicción de sus vectores de medida, es decir, h_i . A la hora de predecir dichos vectores para cada una de las marcas seleccionadas, es necesario conocer dos datos:

- La posición de cada marca Y_i en el sistema del sub-mapa.

- La pose del vehículo X_p en el sistema de referencia del sub-mapa (ver apartado 4.2).

Si se obtiene la matriz de rotación del vehículo $X_p \Rightarrow R_{rob}$, tal como se describirá más adelante en la ecuación 4.92, se puede calcular el vector de medida predicho según la ecuación 4.29, donde X_{rob} es la posición del vehículo en el sub-mapa obtenida a partir de X_p .

$$h_i = R_{rob}^{-1}(Y_i - X_{rob}) \quad (4.29)$$

4.8.2. Cálculo del área de búsqueda

Una vez obtenida la predicción de la medida para cada una de las marcas, el siguiente paso es la obtención de la medida real z_i . Para ello será necesario buscar la proyección de cada marca en un determinado área de búsqueda. En este apartado se presentan las tareas a realizar de cara a determinar las regiones, en ambas imágenes de proyección, con mayor probabilidad de encontrar la mencionada medida real.

Tal y como se describe en el apartado 4.2, la predicción del vector de medida h_i viene definida por las coordenadas de cada marca en el sistema de referencia del vehículo.

Los pasos a seguir para obtener las coordenadas sobre las imágenes izquierda y derecha en píxeles (u_L, v_L) y (u_R, v_R) y los jacobianos $\frac{\partial U_L}{\partial h_i}$ y $\frac{\partial U_R}{\partial h_i}$ son los siguientes, tomando como ejemplo las coordenadas de la proyección en la cámara izquierda.

a) *Cámara izquierda:*

En primer lugar es posible obtener las coordenadas de proyección en unidades de píxel (u_{LS}, v_{LS}) directamente a partir de h_i , tal y como se describe en la introducción del apartado, según se muestra en 4.30.

$$u_{LS} = CC1_L - FC1_L \frac{h_{ix}}{h_{iz}}; \quad v_{LS} = CC2_L - FC2_L \frac{h_{iy}}{h_{iz}} \quad (4.30)$$

Para calcular los jacobianos, derivamos los términos correspondientes según 4.31, donde $U_{LS} = \begin{pmatrix} u_{LS} \\ v_{LS} \end{pmatrix}$ y $U_L = \begin{pmatrix} u_L \\ v_L \end{pmatrix}$.

$$\frac{\partial U_{LS}}{\partial h_i} = \begin{pmatrix} -\frac{FC1_L}{h_{iz}} & 0 & \frac{FC1_L}{h_{iz}^2} \cdot h_{ix} \\ 0 & -\frac{FC2_L}{h_{iz}} & \frac{FC2_L}{h_{iz}^2} \cdot h_{iy} \end{pmatrix} \quad (4.31)$$

El siguiente paso es calcular las correspondientes coordenadas de imagen (u_L, v_L) tras aplicar el modelo de distorsión directo:

1. En primer lugar se desnormalizan las coordenadas (u_{LS}, v_{LS}) para expresarlas en unidades métricas:

$$u_{LNC} = -\frac{u_{LS}}{FC1_L} + \frac{CC1_L}{FC1_L} \quad ; \quad v_{LNC} = -\frac{v_{LS}}{FC2_L} + \frac{CC2_L}{FC2_L} \quad (4.32)$$

2. Partiendo de las coordenadas desnormalizadas, el siguiente paso es calcular las nuevas coordenadas de imagen *distorsionadas*. Para ello, se aplica el modelo de distorsión directo, tal como se muestra en las ecuaciones 4.33 y 4.34, donde: $r^2 = u_{LNC}^2 + v_{LNC}^2$ y $f = 1 + K_{1L}r^2 + K_{2L}r^4$.

K_{1L}, K_{2L} son los coeficientes de distorsión radial y P_{1L}, P_{2L} son los coeficientes de distorsión tangencial de la cámara izquierda.

$$u_{LC} = f \cdot u_{LNC} + 2P_{1L}u_{LNC}v_{LNC} + P_{2L}(r^2 + 2u_{LNC}^2) \quad (4.33)$$

$$v_{LC} = f \cdot v_{LNC} + P_{1L}(r^2 + 2v_{LNC}^2) + 2P_{2L}u_{LNC}v_{LNC} \quad (4.34)$$

3. Por último, una vez obtenidas las coordenadas ya corregidas, ya sólo queda normalizarlas para obtener las coordenadas reales de imagen.

$$u_L = -u_{LC}FC1_L + CC1_L \quad ; \quad v_L = -v_{LC}FC2_L + CC2_L \quad (4.35)$$

Para calcular $\frac{\partial U_L}{\partial h_i}$ se necesita el jacobiano $\frac{\partial U_L}{\partial U_{LS}}$. Derivando los términos correspondientes se obtiene la ecuación 4.36.

$$\frac{\partial U_L}{\partial U_{LS}} = \begin{pmatrix} \frac{\partial u_L}{\partial u_{LS}} & \frac{\partial u_L}{\partial v_{LS}} \\ \frac{\partial v_L}{\partial u_{LS}} & \frac{\partial v_L}{\partial v_{LS}} \end{pmatrix} \quad (4.36)$$

Los cuatro elementos del jacobiano de la ecuación 4.36 se pueden obtener a partir de las ecuaciones 4.37 a 4.40.

$$\frac{\partial u_L}{\partial u_{LS}} = 2u_{LNC}^2 (K_{1L} + 2K_{2L}r^2) + r^2 + 2P_{1L}v_{LNC} + 6P_{2L}u_{LNC} \quad (4.37)$$

$$\frac{\partial u_L}{\partial v_{LS}} = \frac{FC1_L}{FC2_L} \cdot (2u_{LNC}v_{LNC} (K_{1L} + 2K_{2L}r^2) + 2P_{1L}u_{LNC} + 2P_{2L}v_{LNC}) \quad (4.38)$$

$$\frac{\partial v_L}{\partial v_{LS}} = \frac{FC2_L}{FC1_L} \cdot (2u_{LNC}v_{LNC} (K_{1L} + 2K_{2L}r^2) + 2P_{1L}u_{LNC} + 2P_{2L}v_{LNC}) \quad (4.39)$$

$$\frac{\partial v_L}{\partial u_{LS}} = 2v_{LNC}^2 (K_{1L} + 2K_{2L}r^2) + r^2 + 2P_{2L}u_{LNC} + 6P_{1L}v_{LNC} \quad (4.40)$$

Por lo tanto, finalmente se puede calcular $\frac{\partial U_L}{\partial h_i}$ resolviendo 4.41.

$$\frac{\partial U_L}{\partial h_i} = \frac{\partial U_L}{\partial U_{LS}} \cdot \frac{\partial U_{LS}}{\partial h_i} \quad (4.41)$$

b) *Cámara derecha:*

Para obtener las coordenadas de imagen de la cámara derecha (u_R, v_R) y el jacobiano $\frac{\partial U_R}{\partial h_i}$, dado que el sistema de referencia del vehículo coincide con el de la cámara izquierda, es necesario transformar h_i para expresarlo en el sistema de coordenadas de la cámara derecha. A este vector transformado se le denominará h_{iR} , tal como se explicó al principio del apartado. Su obtención se realiza según la ecuación 4.42, donde R_{int} es la matriz de rotación cámara derecha \rightarrow cámara izquierda y

$T_{int} = \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix}$ es la matriz de rotación cámara derecha \rightarrow cámara izquierda.

$$h_{iR} = R_{int}^{-1} \cdot (h_i - T_{int}) \quad (4.42)$$

Para calcular (u_R, v_R) se seguirán los mismos pasos que para la cámara izquierda pero sustituyendo los parámetros para la cámara derecha: $K_{1R}, K_{2R}, P_{1L}, P_{2R}$ y h_{iR} .

Por otro lado, para obtener $\frac{\partial U_R}{\partial h_i}$ en primer lugar se calculará $\frac{\partial U_R}{\partial h_{iR}}$ de la misma forma que en el caso de la cámara izquierda, sustituyendo también los correspondientes parámetros.

Una vez obtenido $\frac{\partial U_R}{\partial h_i}$, se necesitará también obtener $\frac{\partial h_{iR}}{\partial h_i}$. Calculando el jacobiano sobre la ecuación 4.42, se obtiene la ecuación 4.43.

$$\frac{\partial h_{iR}}{\partial h_i} = R_{int}^{-1} \quad (4.43)$$

Una vez obtenidas las coordenadas de imagen, resultado de la predicción del modelo, el siguiente paso a la hora de realizar la búsqueda de la marca en el proceso de medida es definir el *área de búsqueda* en torno a las coordenadas de imagen predichas.

Para calcular la incertidumbre en la predicción de las coordenadas de imagen en cada cámara, es necesario relacionarla con la incertidumbre en la medida de la posición de la marca en cuestión. Esta incertidumbre es la denominada *covarianza de innovación* S_i , la cual proviene básicamente de tres fuentes diferentes:

- Incertidumbre en la posición real del vehículo P_{XX}
- Incertidumbre en la posición real de la marca $P_{Y_i Y_i}$
- Incertidumbre en la medida de dicha posición (ruido de medida) R_i

Teniendo en cuenta las covarianzas cruzadas, su cálculo queda como se muestra en la ecuación 4.44 (ver trabajo [7]).

$$S_i = \frac{\partial h_i}{\partial X_v} P_{XX} \left(\frac{\partial h_i}{\partial X_v} \right)^T + \frac{\partial h_i}{\partial X_v} P_{XY_i} \left(\frac{\partial h_i}{\partial Y_i} \right)^T + \frac{\partial h_i}{\partial Y_i} P_{Y_i X} \left(\frac{\partial h_i}{\partial X_v} \right)^T + \frac{\partial h_i}{\partial Y_i} P_{Y_i Y_i} \left(\frac{\partial h_i}{\partial Y_i} \right)^T + R_i \quad (4.44)$$

La obtención de los jacobianos implicados en la fórmula se deduce fácilmente de los resultados obtenidos en el apartado 4.9.3.

El siguiente paso es transformar la incertidumbre en la medida, calculada anteriormente, en la incertidumbre en su proyección en ambas cámaras U_L y U_R (incluido el efecto de la distorsión radial y tangencial). Para ello únicamente hay que realizar la transformación que muestran las ecuaciones 4.45 y 4.46 usando los jacobianos previamente calculados.

$$P_{U_L} = \frac{\partial U_L}{\partial h_i} \cdot S_i \cdot \left(\frac{\partial U_L}{\partial h_i} \right)^T \quad (4.45)$$

$$P_{U_R} = \frac{\partial U_R}{\partial h_i} \cdot S_i \cdot \left(\frac{\partial U_R}{\partial h_i} \right)^T \quad (4.46)$$

Las dos covarianzas calculadas definen sendas densidades de probabilidad gaussianas de media U_L y U_R respectivamente, según se muestra en la Fig. 4.20.

Si se restringe el área de búsqueda a un número de desviaciones típicas (normalmente tres), se obtiene, de forma general, una elipse para cada proyección, cuya ecuación se puede obtener de la siguiente forma:

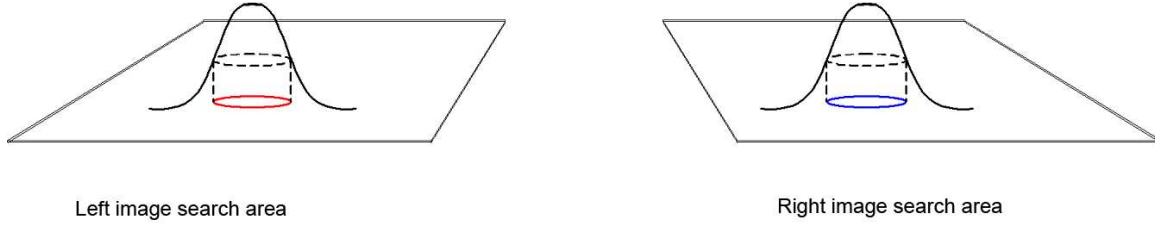


Figura 4.20: Áreas de búsqueda en las proyecciones de la marca, en función de su incertidumbre

1. Teniendo en cuenta la ecuación de la Normal de dos variables y de media cero, por ejemplo para la imagen izquierda queda:

$$\Pr(U) = \frac{1}{\sqrt{(2\pi)^2 |P_{U_L}|}} \cdot e^{-\frac{1}{2}(U^T \cdot P_{U_L}^{-1} \cdot U)} \quad (4.47)$$

2. Tomando la restricción de búsqueda a 3σ , se puede obtener la expresión 4.48.

$$U^T \cdot P_{U_L}^{-1} \cdot U < 3^2 \quad (4.48)$$

Es decir, los puntos que cumplen una distancia de *Mahalanobis* al origen igual a 3.

3. Desarrollando la ecuación anterior se obtiene la expresión 4.49 que delimita el área (u,v) donde realizar la búsqueda:

$$u^2 \left(P_{U_L}^{-1} \right)_{11} + 2uv \left(P_{U_L}^{-1} \right)_{12} + v^2 \left(P_{U_L}^{-1} \right)_{22} < 3^2 \quad (4.49)$$

A la hora de realizar la búsqueda será necesario desplazar las elipses calculadas hasta el valor de la media U_L ó U_R .

4.8.3. Determinación de la medida

Para determinar el vector de medida z_i es necesario obtener previamente las coordenadas de proyección reales de cada marca en el frame actual (u_{Lnew}, v_{Lnew}) y (u_{Rnew}, v_{Rnew}). Para ello es necesario identificar unívocamente cada marca obteniendo un vector de características. Posteriormente, en el proceso de medida se realizará una búsqueda de dicho vector de características en el área obtenida anteriormente. Esta búsqueda consiste generalmente en el cálculo de la distancia euclídea entre el vector de características que identifica la marca y uno perteneciente al área de búsqueda.

En la bibliografía existen diversos métodos de extracción de características a partir de una imagen, como por ejemplo PCA (*Principal Components Analysis*, “Análisis de Componentes principales”) [63] [64], Haar Wavelets [65], SVM (*Support Vector Machines*, “Máquinas de Soporte Vectorial”) [66], SIFT (*Scale-Invariant Feature Transform*, “Transformada de Características Invariante a la Escala”) [67] [17] [68] y correlación directa [7].

Dos de los más empleados recientemente son *SIFT* y la *correlación directa*, siendo éstos los métodos analizados para su aplicación en el sistema propuesto. A continuación, se describen resumidamente ambos métodos, para posteriormente realizar un estudio comparativo en el que

se comparará tanto la robustez ante los cambios de escala y perspectiva como los tiempos de cómputo empleados.

El método SIFT, para la generación de vectores de características a partir de una imagen, toma una imagen y la transforma en una colección de vectores de características locales. Cada uno de estos vectores es invariante a cualquier cambio de escala, rotación o translación de la imagen. Para ayudar en la extracción de estos vectores el algoritmo SIFT aplica una estrategia de filtrado en cuatro etapas:

1. *Detección de extremos en el espacio de escala.* En esta etapa del filtrado se identifican aquellas localizaciones y escalas que sean identificables desde diferentes puntos de vista. Esto se puede realizar eficientemente usando una función de *espacio de escala*. Esta función, se ha demostrado bajo suposiciones razonables que debe ser la función Gaussiana. La función de espacio de escala se define según 4.50.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (4.50)$$

Siendo $*$ el operador convolución, G es una Gaussiana de escala variable e I es la imagen de entrada. Por otra parte, x e y son las coordenadas del punto de la imagen a evaluar y σ la escala asignada al mismo.

En este punto se pueden usar varias técnicas para detectar localizaciones estables de *key-points* (puntos característicos) en el espacio de escala. Una de estas técnicas es la *Diferencia de Gaussianas*, la cual localiza extremos en el espacio de escala $D(x, y, \sigma)$ calculando la diferencia entre dos imágenes: una con una escala K veces superior a la otra. Por lo tanto D se obtiene como se muestra en 4.51.

$$D(x, y, \sigma) = L(x, y, K\sigma) - L(x, y, \sigma) \quad (4.51)$$

Para detectar máximos y mínimos locales de D cada punto se compara con sus ocho vecinos en la misma escala y sus nueve vecinos una escala por encima y por debajo. Si este valor es el mínimo o el máximo entre todos esos puntos, entonces este punto es un extremo, tal como se muestra en la Fig. 4.21.

2. *Localización de Keypoints.* En esta etapa se trata de eliminar puntos de la lista de keypoints buscando aquellos que tienen un contraste bajo o se localizan de forma incorrecta en un borde. Esto se consigue calculando el valor de la Laplaciana para cada uno de los keypoints encontrados en la etapa anterior. La localización de extremos viene dada por la ecuación 4.52.

$$z = -\frac{\delta^2 D^{-1}}{\delta x^2} \cdot \frac{\delta D}{\delta x} \quad (4.52)$$

Si el valor de la función en z está por debajo de un determinado umbral, entonces este punto es excluido. Con esto se eliminan los extremos con bajo contraste. Para eliminar los extremos basados en localizaciones pobres se debe tener en cuenta que en estos casos existe una gran curvatura principal a lo largo del eje pero una curvatura pequeña en la dirección perpendicular dentro de la función Diferencia de Gaussianas. Si esta diferencia es menor que la proporción entre el mayor y el menor autovector, calculado en la matriz Hessiana de 2x2 en la localización y escala del keypoint, entonces dicho keypoint es rechazado.

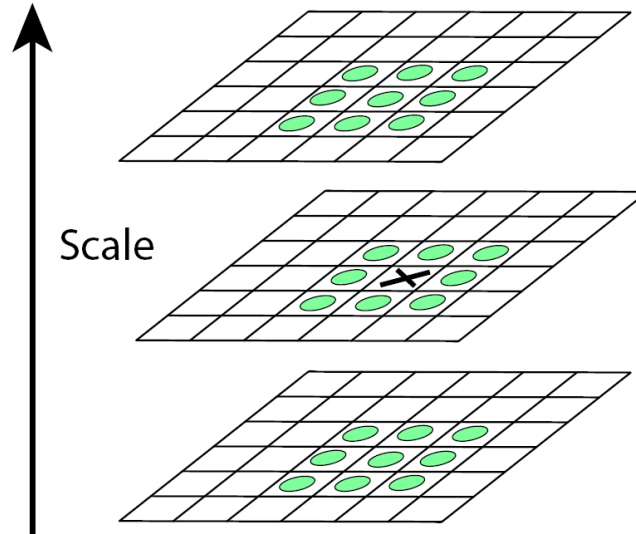


Figura 4.21: Detección de extremos en el espacio de escala

3. *Asignación de orientación.* En esta etapa se asigna una orientación consistente a los keypoints basada en propiedades locales de la imagen. El descriptor del keypoint, descrito más abajo, se puede representar de forma relativa a esta orientación, consiguiéndose invarianza a la rotación. La forma de obtener la orientación es la que sigue:

- Uso de la escala de los keypoints para seleccionar la imagen suavizada usando la Gaussiana.
- Cálculo del valor del gradiente m , según la ecuación 4.53.

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (4.53)$$

- Cálculo de la orientación θ , como se muestra en 4.54.

$$\theta(x, y) = \arctan \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \quad (4.54)$$

- Formación de un histograma de orientación a partir de los gradientes de orientación de los puntos elegidos.
 - Localización del máximo en el histograma. Se usa este máximo y cualquier otro máximo local dentro del 80 % de la altura de éste para crear un keypoint con esa orientación.
 - A algunos puntos se les asignará múltiples orientaciones.
 - Ajuste de una parábola a los tres valores del histograma más cercanos a cada máximo para interpolar las posiciones de éstos.
4. *Descriptor del keypoint.* Los datos del gradiente local, usados anteriormente, se usan también para crear los descriptores de los keypoints. La información de gradiente se rota hasta alinearse con la orientación del keypoint y se pondera por una Gaussiana de varianza 1.5 veces la escala del keypoint. Estos datos se usan para crear una serie de histogramas en una ventana centrada en el keypoint.

Los descriptores de los keypoints usan típicamente un conjunto de 16 histogramas, alineados en una rejilla de 4x4, conteniendo cada uno ocho posibles orientaciones. Esto da como resultado un vector de características compuesto de 128 elementos.

Para aplicarlo al sistema presentado, una opción sería realizar la búsqueda de keypoints a lo largo de cada una de las imágenes de proyección, de forma global. Sin embargo, esta opción se descarta rápidamente debido al alto coste computacional.

Para optimizar el área de búsqueda se toman como punto de partida las regiones de máxima probabilidad calculadas anteriormente. Dado que las marcas SIFT pueden tener asociadas diversas escalas, a la hora de definir una región de búsqueda, ésta debe ser lo suficientemente amplia como para contener la marca original que se pretende buscar. El método propuesto consiste en definir un área de búsqueda rectangular, tomando como base el rectángulo contenedor de la región elíptica de búsqueda y aumentándolo en un determinado margen adicional. Cuanto mayor sea el margen adicional, mejor será el resultado a la hora de encontrar la marca original, a cambio de un incremento en el tiempo de cómputo. Al final del apartado se muestra un estudio de estos aspectos.

En las ecuaciones 4.55 y 4.56 se muestra la obtención del rectángulo de búsqueda para la imagen de proyección izquierda en función de los elementos de la matriz Pu_L^{-1} y teniendo en cuenta la restricción de búsqueda de 3σ , donde w y h representan la altura y anchura de la región de búsqueda respectivamente. MA representa el margen adicional añadido a la región de búsqueda.

$$\frac{w}{2} = \frac{3}{\sqrt{(Pu_L^{-1})_{11} - \frac{(Pu_L^{-1})_{12}^2}{(Pu_L^{-1})_{22}}}} + MA \quad (4.55)$$

$$\frac{h}{2} = \frac{3}{\sqrt{(Pu_L^{-1})_{22} - \frac{(Pu_L^{-1})_{12}^2}{(Pu_L^{-1})_{11}}}} + MA \quad (4.56)$$

Una vez identificados los keypoints candidatos se procede a compararlos empleando la distancia euclídea entre sus correspondientes descriptores y el de la marca original.

En la Fig. 4.22 se muestra la imagen de proyección izquierda en la que se indica la región de búsqueda de una marca SIFT y los keypoints candidatos que se deben encontrar en el interior de dicha región. A continuación, en la Fig. 4.23 se muestran las marcas SIFT presentes en el frame, donde la flecha indica la orientación de la marca y su tamaño la escala.

La otra alternativa para la realización de la medida es el uso de la *correlación directa*. En este método, el vector de características está definido directamente por los valores de todos los píxeles del patch. Para comparar la similitud del original con las muestras de la región de búsqueda se realiza el proceso de correlación descrito a continuación.

El objetivo de este método es, por tanto, encontrar dentro de la región de búsqueda el *patch* con una mejor correlación con el patch extraído de la imagen original.

Para ello, se parte de la imagen de tamaño $B \times B$ y la región de búsqueda en la nueva imagen. Para cada uno de los puntos de la región de búsqueda, se realiza una correlación de tipo *suma normalizada de diferencias al cuadrado*. Esto se expresa en la ecuación 4.57.

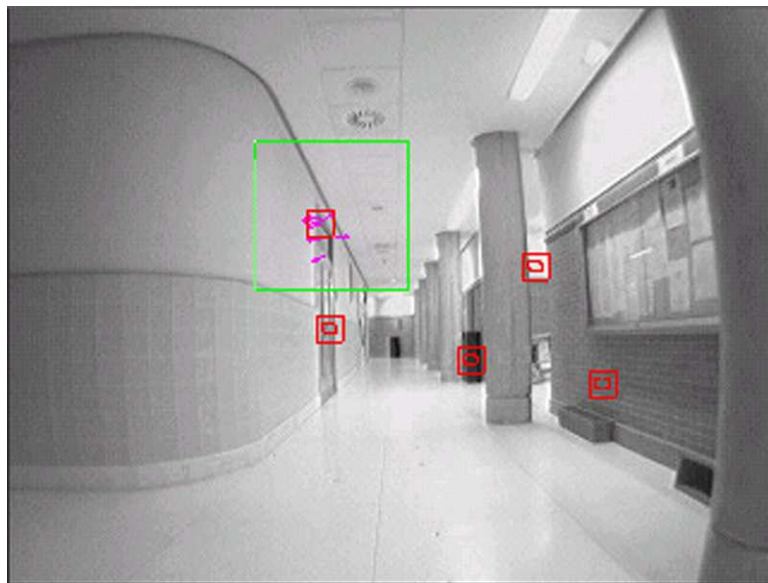


Figura 4.22: Keypoints candidatos dentro del área de búsqueda



Figura 4.23: Marcas SIFT en un frame

$$C = \frac{\sum_{i=i_0, j=j_0}^{i=i_{\max}, j=j_{\max}} \left[\frac{g_1(i+i_{offset}, j+j_{offset}) - \bar{g}_1}{\sigma_1} - \frac{g_0(i, j) - \bar{g}_0}{\sigma_0} \right]^2}{n_{pixels}} \quad (4.57)$$

Donde g_0 y g_1 son los valores de gris de los píxeles correspondientes en la posición (j, i) del extracto de la imagen original y el patch actualmente seleccionado en la región de búsqueda. \bar{g}_0 y \bar{g}_1 son los niveles medios de gris en las dos áreas mencionadas anteriormente, σ_0 y σ_1 son sus correspondientes desviaciones típicas y, por último, n_{pixels} es el número de píxeles totales en el patch. Cuanto mayor sea el valor de C , peor será la correlación.

Gráficamente, en la Fig. 4.24 se muestra el proceso de correlación píxel a píxel, donde (j_{offset}, i_{offset}) indica la posición del patch a evaluar respecto a la posición del patch original a buscar (j_0, i_0) . Teniendo en cuenta que el patch es siempre cuadrado: $B = j_{\max} - j_0 = i_{\max} - i_0$.

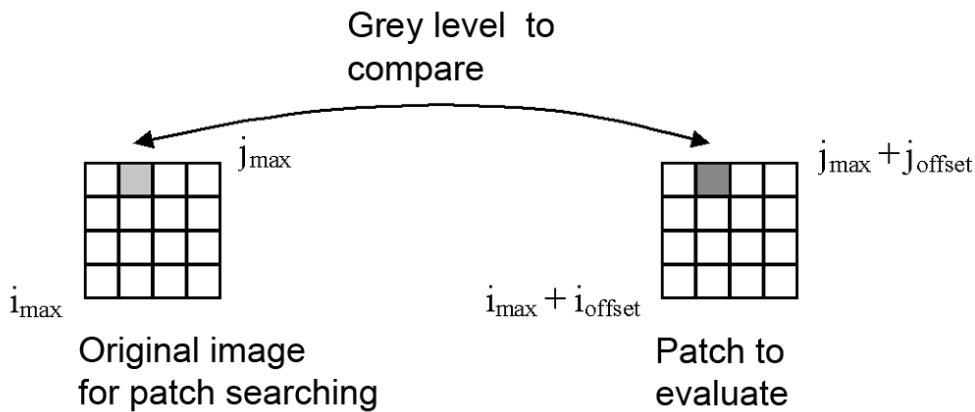


Figura 4.24: Proceso de correlación píxel a píxel

De cara a elegir el método más adecuado, en la implementación del sistema presentado, se ha procedido a realizar un estudio comparativo entre las dos alternativas presentadas anteriormente, esto es, el método *SIFT* y el método de *correlación directa*.

En primer lugar se ha realizado un estudio estadístico de la estabilidad y robustez de ambos métodos. Para ello se han obtenido el número de medidas correctas realizadas en cada marca antes de ser eliminada del proceso. En la Fig. 4.25 se muestran los valores de medidas correctas para el caso de usar correlación directa y para el caso de usar *SIFT*. En este último caso se han estudiado los resultados usando diferentes márgenes adicionales de búsqueda *MA* con forma cuadrada.

Como se puede apreciar, a medida que el margen de búsqueda aumenta, los éxitos en las medias también son más numerosos. El caso de correlación directa es significativamente bueno, sólo superado por el caso de mayor área de búsqueda *SIFT*.

Por otro lado, un aspecto a tener en cuenta en el estudio es el tiempo de cómputo empleado en cada método. En la Fig. 4.26 se muestran los tiempos de cómputo totales para cada uno de los casos mencionados anteriormente. Se puede apreciar cómo, a pesar de las restricciones impuestas en el área de búsqueda, el único método capaz de ejecutarse por debajo de la restricción de tiempo real establecido es la correlación directa, siendo éste el método elegido en el sistema presentado.

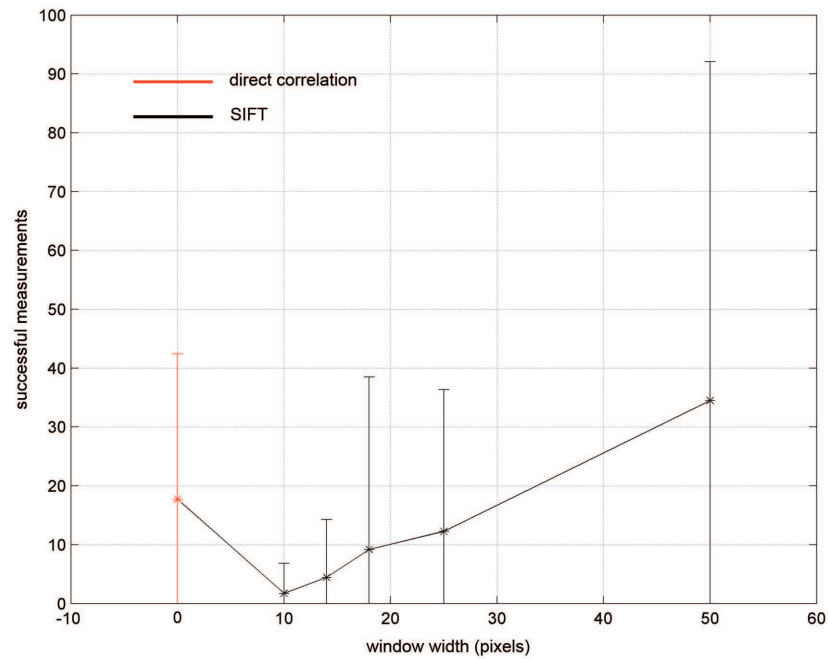


Figura 4.25: Número de correlaciones correctas por marca para los casos de correlación directa y SIFT (media y desviación típica). Este último caso se estudia para diferentes tamaños de área de búsqueda

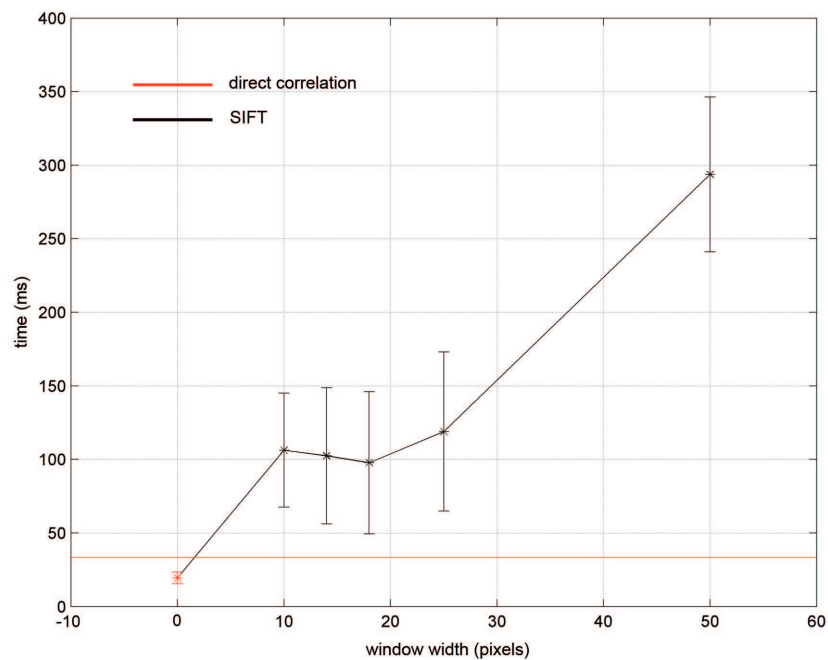


Figura 4.26: Tiempos de cómputo totales por frame para los casos de correlación directa y SIFT (media y desviación típica). Este último caso se estudia para diferentes tamaños de área de búsqueda

Por último, en la Fig. 4.27 se muestran los resultados de estimación obtenidos para un pequeño recorrido en interiores. A partir de estos resultados se observan dos puntos de desviación principales en cada implementación. La primera desviación se acusa usando el método SIFT en la primera curva del camino, mientras que usando la implementación propuesta el vehículo tiende a desviarse en la tercera curva. Representando el error acumulado medio $\varepsilon_n = (1/n_{total}) \sum_{i=0}^n |X_i - X_{ref_i}|$, en la Fig. 4.28 se aprecia que éste es mayor en la implementación SIFT. X_i representa la estimación realizada por el sistema de SLAM, mientras que X_{ref_i} representa la referencia.

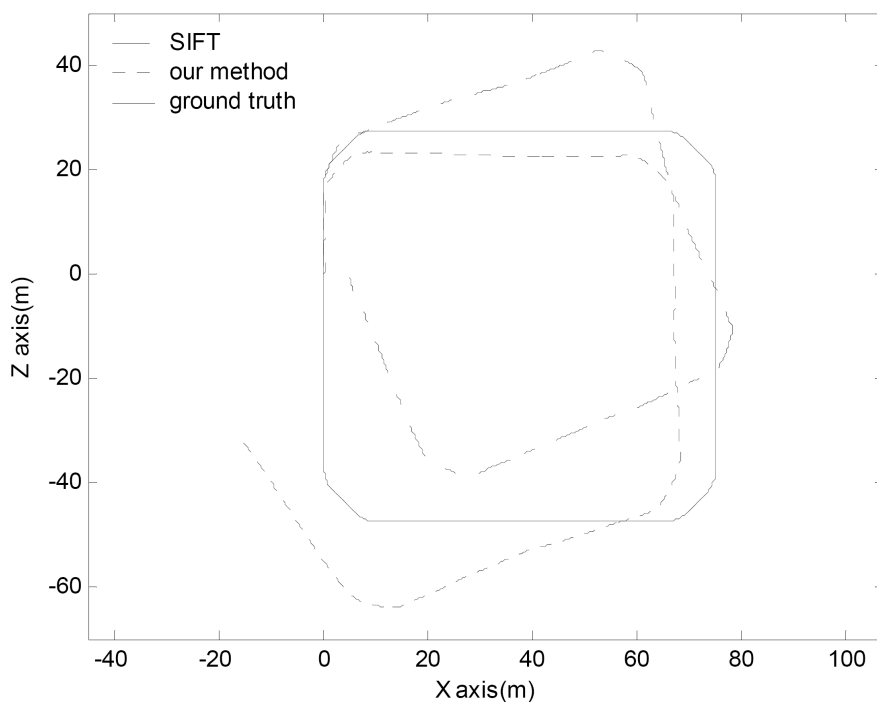


Figura 4.27: Estimación del camino de prueba recorrido por el vehículo usando SIFT y correlación directa. La referencia (ground truth) aparece como línea continua

4.8.4. Transformación del patch

A medida que el vehículo avanza en su camino los patches, correspondientes a las marcas capturadas en un determinado momento, cambian su apariencia relativa.

Como se ha explicado en el apartado anterior, en el proceso de búsqueda por correlación directa, en cada frame se realizan correlaciones de la imagen del patch actual respecto al patch almacenado en el momento de la inicialización de la marca. Dado el cambio en la apariencia del patch correspondiente a la posición actual, si la diferencia entre la pose actual y la original en el momento de la inicialización es suficientemente grande, la correlación será errónea. Para tener en cuenta este fenómeno se definió el concepto de visibilidad.

Para permitir una mejora en el proceso de correlaciones, se propone una técnica de transformación de la apariencia de los patches almacenados basada en la pose actual del vehículo. Dado que el movimiento del vehículo es fundamentalmente de avance, se supondrá que la estructura tridimensional del patch se corresponde con un plano paralelo al plano de visión de la cámara (ver Fig. 4.29). De esta forma, a medida que el vehículo avance acercándose a los patches, éstos

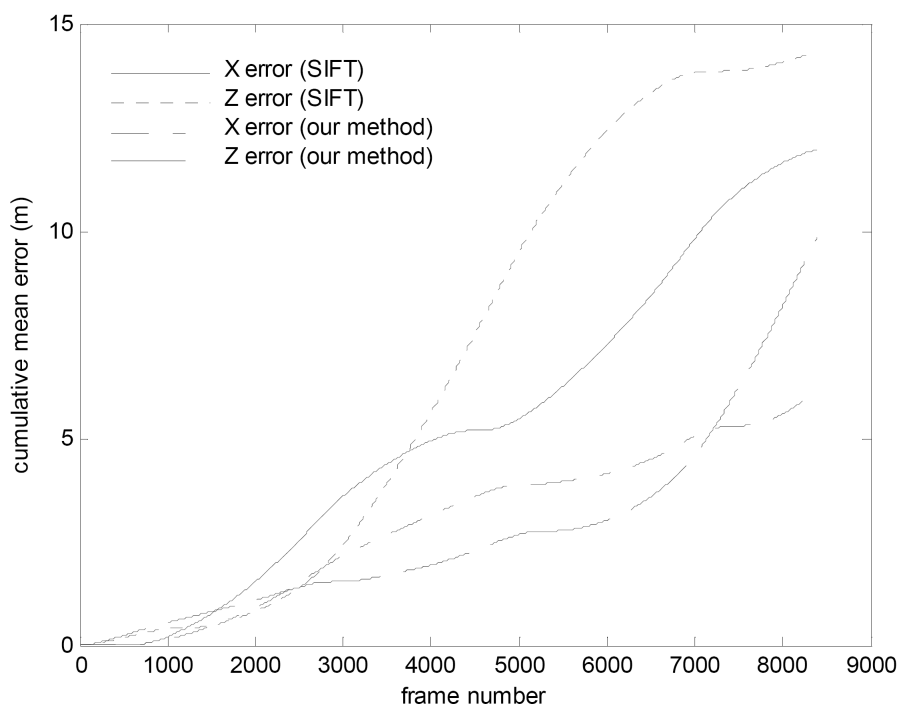


Figura 4.28: Error acumulado medio para el eje X y Z en función del número de frame n . Los resultados se muestran tanto para la implementación SIFT como para el método propuesto

se harán mayores.

Con todo lo explicado anteriormente, además del cálculo de las coordenadas de proyección del patch, para poder realizar la transformación de su apariencia será necesario estimar las proyecciones de cada uno de los píxeles del patch, teniendo en cuenta el *estado actual* del vehículo y las *posiciones 3D almacenadas* de cada uno de los puntos correspondientes tal y como se muestra en la Fig. 4.29 (ver apartado 4.9.2).

Una vez calculadas las nuevas proyecciones de cada uno de los píxeles del patch, antes de realizar las correlaciones, es necesario asignar un valor a todos los píxeles que quedaron sin correspondencia en el paso anterior. Existen diversos métodos de interpolación para estimar dichos valores, como por ejemplo la *interpolación bilineal*, *interpolación bicúbica*, etc. Dado que el método puede ser costoso en términos de tiempo de procesamiento, se ha optado por el uso de un método de *interpolación por vecindad*. Además, dicho método será aplicado de una forma eficiente. La aplicación del método es la siguiente:

1. En primer lugar, se deberán identificar todos los píxeles vacíos.
2. Posteriormente, para cada uno de dichos píxeles, se realizará una búsqueda progresiva del píxel *ocupado* más cercano, tal y como se muestra en la Fig. 4.30.
3. Por último, el píxel vacío tomará el valor del píxel ocupado encontrado.

Para comprobar las mejoras obtenidas con el método propuesto se han evaluado los tiempos de vida por cada marca. Es decir, desde que una marca es inicializada hasta que es eliminada, a causa del exceso de correlaciones incorrectas. Esto a su vez determinará los límites de visibilidad tanto en ángulo como en distancia. Cómo se puede apreciar en la Fig. 4.31 y Fig. 4.32, los tiempos

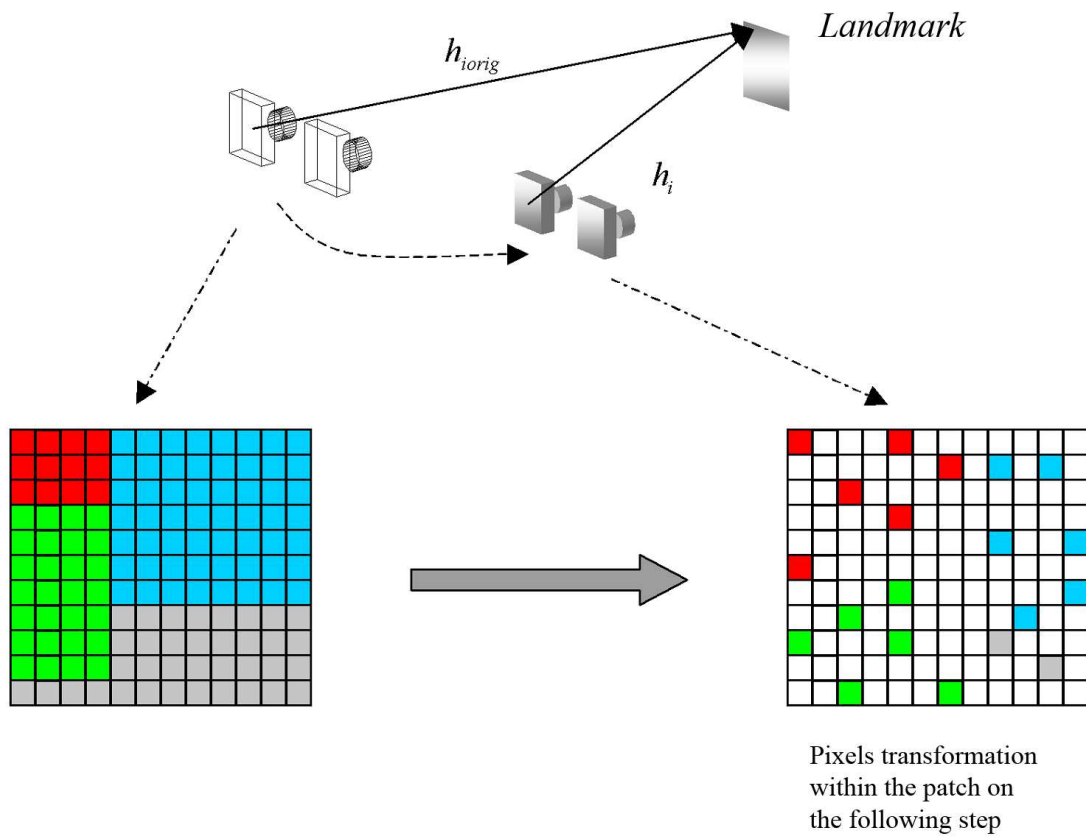


Figura 4.29: Proceso de transformación de la apariencia del patch

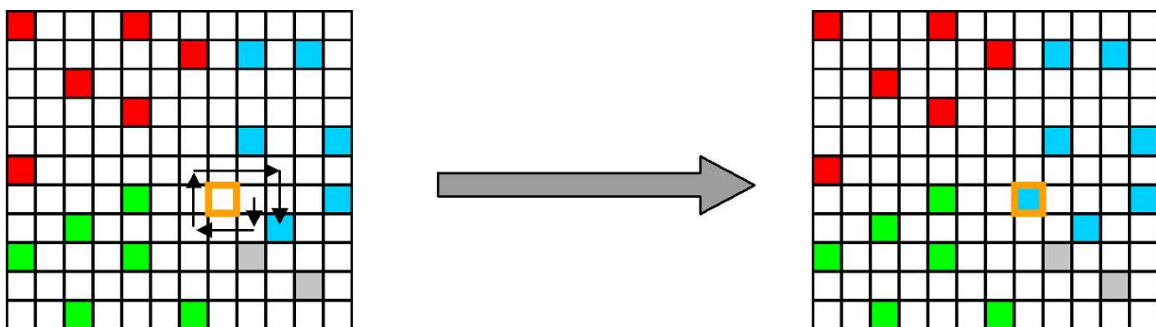


Figura 4.30: Proceso de búsqueda progresiva del píxel ocupado más cercano

de vida por marca son notablemente superiores usando el método de transformación de patch presentado. Es decir, la transformación de la apariencia del patch permite obtener correlaciones correctas a lo largo de cambios de puntos de vista durante más tiempo. Esto redundará en una utilización más eficiente de la información proporcionada por las marcas, obteniéndose una mayor precisión en la estimación del vector de estado del vehículo.

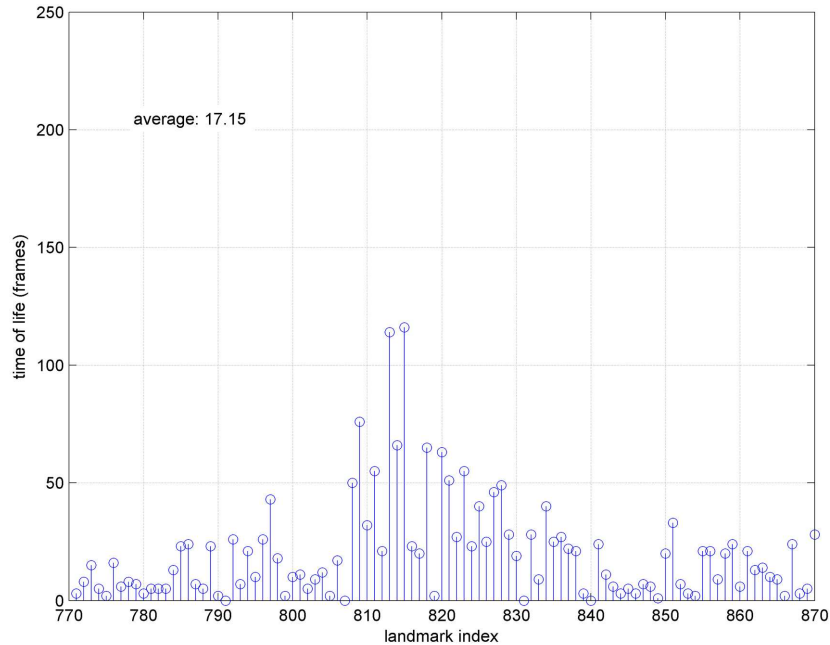


Figura 4.31: Tiempos de vida de las marcas sin aplicar el método de transformación de patch. Se muestra el número de correlaciones exitosas para cada marca.

Por otra parte, el hecho de realizar un proceso de interpolación conlleva un tiempo de cómputo adicional que se suma al resto de tareas del proceso global. En la Fig. 4.33 se comparan los tiempos de cómputo por frame, aplicando el método de patch adaptado y sin aplicarlo. En el capítulo de resultados se analizan con detalle los orígenes de los diferentes tiempos de cómputo. Como se puede apreciar en la gráfica, a pesar del ligero aumento en el tiempo de cómputo, éste sigue manteniéndose por debajo del límite de tiempo real establecido. Por lo tanto, la aplicación del método queda justificada.

4.8.5. Actualización

Una vez realizada la medida, se obtienen las nuevas coordenadas de proyección en cada cámara (u_{Lnew}, v_{Lnew}) , (u_{Rnew}, v_{Rnew}) . Antes de realizar la actualización es necesario transformar las coordenadas de proyección en el vector de medida real $z_i = f(u_{Lnew}, v_{Lnew}, u_{Rnew}, v_{Rnew})$, es decir, las nuevas coordenadas de la marca respecto del sistema de referencia del vehículo.

Este es un problema de geometría inversa, cuya resolución se explica en el apartado 4.9.2. En este caso el cálculo ha de realizarse en el sistema de referencia del vehículo, tal y como se describe en el apartado 4.8.2.

Una vez obtenido z_i , antes de realizar la actualización, es necesario construir los vectores y matrices implicados como sigue:

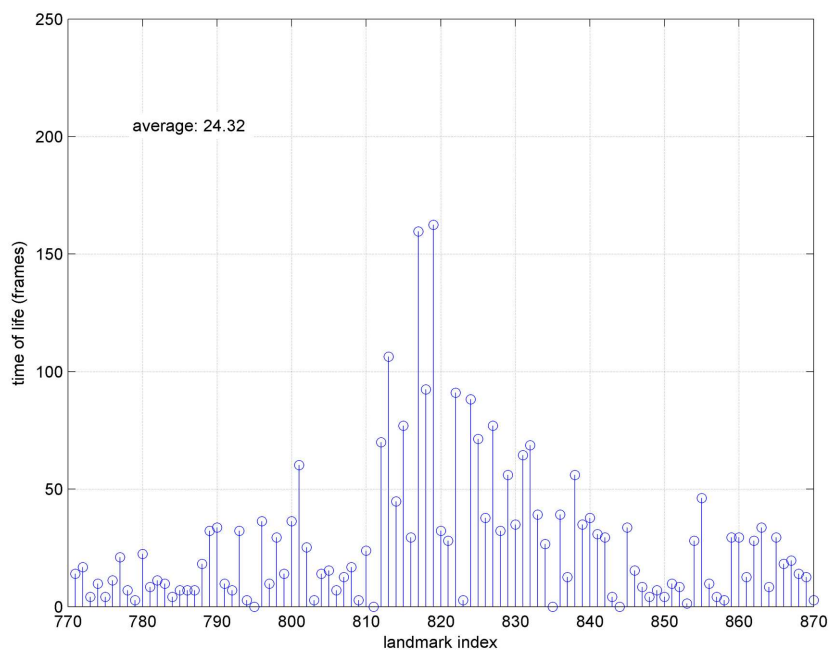


Figura 4.32: Tiempos de vida de las marcas aplicando el método de transformación de patch. Se muestra el número de correlaciones exitosas para cada marca.

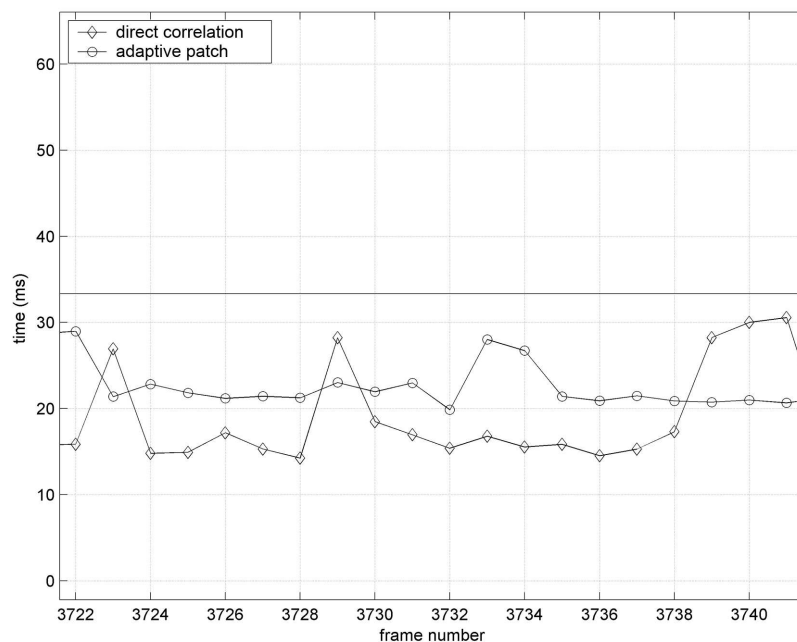


Figura 4.33: Tiempos de cómputo totales empleando correlación directa o la técnica del patch transformado

1. Si se define el vector de innovación $\eta_i = z_i - h_i$ como la diferencia entre el vector de medida real z_i y el predicho h_i , es posible construir η_{tot} correspondiente al conjunto de todas las marcas medidas como se muestra en 4.58, donde n_{sel} es el número de marcas exitosamente medidas.

$$\eta_{tot} = \begin{pmatrix} \eta_1 \\ \eta_2 \\ \dots \\ \eta_{n_{sel}} \end{pmatrix} \quad (4.58)$$

2. El segundo elemento necesario para la actualización del filtro es $\frac{\partial h}{\partial X_{tot}}$. Su construcción se realiza según 4.59.

$$\frac{\partial h}{\partial X_{tot}} = \begin{pmatrix} \frac{\partial h_1}{\partial X_v} & \frac{\partial h_1}{\partial Y_1} & 0 & 0 & 0 \\ \frac{\partial h_2}{\partial X_v} & 0 & \frac{\partial h_2}{\partial Y_2} & 0 & 0 \\ \vdots & 0 & 0 & \ddots & 0 \\ \frac{\partial h_{n_{sel}}}{\partial X_v} & 0 & 0 & 0 & \frac{\partial h_{n_{sel}}}{\partial Y_{n_{sel}}} \end{pmatrix} \quad (4.59)$$

3. Por último, es necesario calcular la covarianza de innovación total S , para lo cual es necesario antes calcular la matriz de ruido de medida total R_{tot} .

$$R_{tot} = \begin{pmatrix} R_1 & 0 & 0 & 0 \\ 0 & R_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & R_{n_{sel}} \end{pmatrix} \quad (4.60)$$

Para calcular la covarianza de innovación S simplemente hay que transformar la matriz de covarianza total y añadir el ruido de medida total, como se muestra en 4.61.

$$S = \left[\frac{\partial h}{\partial X_{tot}} \cdot P \cdot \left(\frac{\partial h}{\partial X_{tot}} \right)^T \right] + R_{tot} \quad (4.61)$$

Una vez construidas las matrices necesarias para la actualización, sólo es necesario aplicar la fórmula de actualización del EKF, tal y como se describe en [69], donde la ganancia de Kalman se obtiene como $W = P \cdot \left(\frac{\partial h}{\partial X_{tot}} \right)^T \cdot S^{-1}$.

$$\hat{X}_{new} = \hat{X}_{old} + W \cdot \eta_{tot} \quad (4.62)$$

$$P_{new} = P_{old} - W \cdot S \cdot W^T \quad (4.63)$$

4.9. Inicialización de nuevas marcas

Una parte importante en la implementación del filtro es la captura de nuevas marcas. Dado que el sistema no parte de ninguna marca conocida a priori, inicialmente el vector de estado total estará compuesto únicamente por el estado del vehículo. Inicialmente, por lo tanto, se

supone que el vehículo se encuentra en el origen de coordenadas del sistema del sub-mapa, con la orientación por defecto (línea visual paralela al eje X). En cuanto a la velocidad lineal y angular es necesario partir de un valor inicial no nulo para permitir la convergencia del filtro. Obviamente, la matriz de covarianza total P se supondrá inicialmente nula ya que se parte de una posición inicial preestablecida.

El primer paso tras comenzar la ejecución del filtro es obtener la primera marca a introducir en el filtro. Tal como se explicó anteriormente, dado que se parte de una posición del vehículo preestablecida, la primera marca capturada será inicializada como si fuera conocida a priori, es decir, tendrá una matriz de covarianza P_{YY} nula. A la hora de buscar el patch correspondiente a la mejor marca para realizar el seguimiento, la región de búsqueda a tener en cuenta será toda la imagen captada por la cámara izquierda, exceptuando un margen en los bordes exteriores del tamaño del propio patch, tal y como se muestra en la Fig. 4.34.

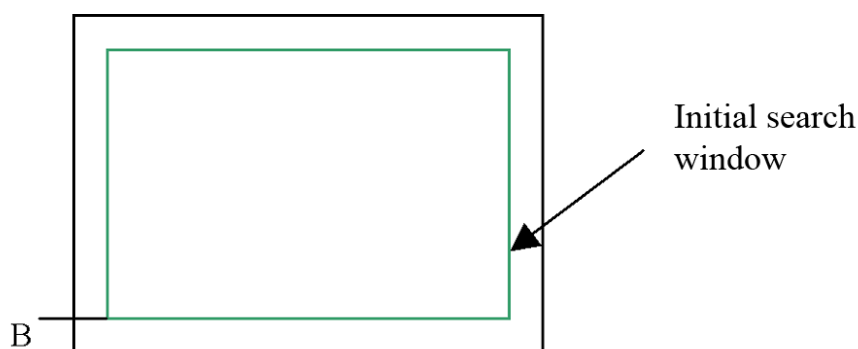


Figura 4.34: Ventana de búsqueda inicial

A la hora de introducir los elementos de la nueva marca en el filtro, el procedimiento será el explicado en el apartado 4.9.4.

Una vez capturada e inicializada la primera marca, en cada iteración del filtro se irán capturando nuevas marcas en función de los siguientes parámetros:

- *Número de marcas visibles:* El número mínimo de marcas visibles no debe ser inferior a un valor predeterminado, en este caso 5. Por lo tanto, siempre que haya menos de cinco marcas visibles, será necesario buscar una nueva marca a incorporarla en el filtro.
- *Número de marcas exitosamente medidas:* Tal y como se explica en el apartado 4.8.3, en cada iteración del filtro se intenta medir la posición de las marcas visibles por medio de la correlación. Si la correlación no es suficientemente buena, la medida se toma como fallida, con lo que dichas marcas no serán utilizadas a la hora de actualizar el filtro. Si el número de marcas exitosamente medidas es inferior a un cierto valor, en este caso 4, es posible perder la convergencia del filtro después de ciertas iteraciones. En esta situación, por lo tanto, también se procederá a buscar una nueva marca para inicializar.

Llegado el momento de buscar una nueva marca, el proceso es el siguiente:

1. Se genera una región de búsqueda rectangular (50 x 100 píxeles) situada aleatoriamente, con distribución uniforme, dentro de la imagen y excluyendo los márgenes anteriormente descritos, como se indica en la Fig. 4.35.

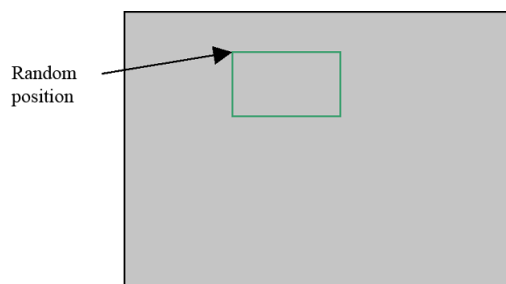


Figura 4.35: Región de búsqueda aleatoria

2. Posteriormente se comprueba si ya existe alguna marca visible y exitosamente medida dentro de la región de búsqueda. Si es así, se descarta la región y se vuelve a generar otra aleatoriamente. La razón de esta comprobación es evitar la acumulación de marcas en un área visual pequeña, ya que la incertidumbre en la posición del vehículo tiende a ser mayor en este caso.

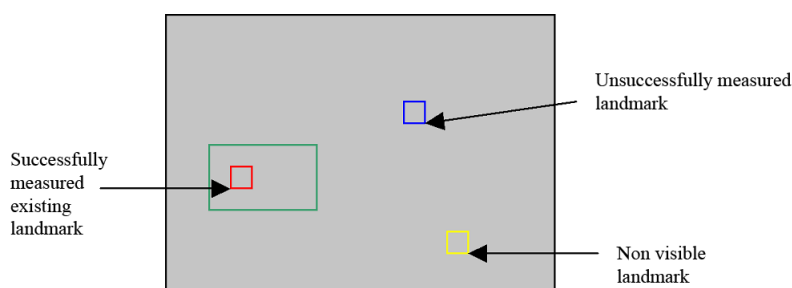


Figura 4.36: Estados posibles de las marcas

3. En caso contrario, se procede a la búsqueda de la marca. Si la idoneidad del patch no es suficientemente buena también se descarta dicha región y se vuelve a generar otra aleatoriamente.
4. En los dos casos anteriores se realizarán hasta un máximo de diez intentos para la obtención de una nueva marca. Si se sobrepasa dicho límite, se permite avanzar al filtro una iteración más para intentarlo de nuevo. En caso contrario se almacenan las coordenadas de imagen de la nueva marca capturada y se procede a obtener su vector de estado y covarianza.

Además de estas tareas, una vez capturada la nueva marca será necesario almacenar la posición X_{porig} del vehículo en ese momento con objeto de calcular la visibilidad de dicha marca, tal como se explica en el apartado 4.8.1.

También será necesario almacenar las imágenes asociadas a la marca (patch izquierdo y patch derecho) para poder realizar las futuras correlaciones (ver apartado 4.8.3). Es importante resaltar que los patches permanecen constantes a lo largo de toda la *vida* de la marca, es decir, no se actualizan con las nuevas vistas. El motivo de esta restricción es evitar el error acumulativo que conllevaría dicha actualización a la hora de localizar la marca.

4.9.1. Método de detección de características

Tal como se ha explicado anteriormente, el primer paso para inicializar una nueva marca es buscarla. Es decir, se busca la marca cuyas características sean las más adecuadas para realizar

su posterior seguimiento.

Si se parte de la ventana de búsqueda definida anteriormente, se tienen los límites (u_{start}, v_{start}) y (u_{finish}, v_{finish}) dentro de los que realizar dicha búsqueda. El proceso de búsqueda se basa en tomar, para cada coordenada (u, v) , un patch de tamaño $B \times B$, que será evaluado para obtener una medida de su idoneidad. El método empleado para obtenerlo se basa en la propuesta planteada en [70]. Este método utiliza el gradiente de intensidad para cada píxel, el cual da información de la *no uniformidad* en los niveles de gris a lo largo de la imagen. Hace uso tanto del gradiente vertical g_y como horizontal g_x , es decir, tiene en cuenta, tanto bordes verticales como horizontales, esto es, detecta cambios bruscos en la imagen.

En el método planteado en [70] se propone el uso de la siguiente matriz aplicada al conjunto de píxeles del patch, como se muestra en 4.64, donde $g_x = \frac{I(x+1,y)-I(x-1,y)}{2}$ y $g_y = \frac{I(x,y+1)-I(x,y-1)}{2}$.

$$Z = \sum_{y=v_{0patch}}^{v_{\max patch}} \sum_{x=u_{0patch}}^{u_{\max patch}} \begin{pmatrix} g_x^2 & g_x g_y \\ g_y g_x & g_y^2 \end{pmatrix} \quad (4.64)$$

Es decir, Z implica el cálculo de los gradientes en todos los píxeles del patch, la formación de las matrices individuales y su posterior suma.

Para la medida del interés del patch evaluado será necesario calcular los dos autovalores λ_1 y λ_2 de la matriz Z . Cada autovalor da información de la uniformidad de la imagen en una determinada dirección, de manera que cuanto mayor sea un autovalor menor será la uniformidad en su correspondiente dirección. Dado que para realizar un buen seguimiento del patch es necesario tener una alta discontinuidad en ambas direcciones, la regla a seguir será tener en cuenta el menor de los autovalores, de forma que cuanto mayor sea éste, mayor será el interés del patch evaluado.

Para calcular la Z total del patch en todos los puntos (u, v) de la región de búsqueda, el procedimiento, optimizado en tiempo de cómputo, será el siguiente:

1. En primer lugar se calculan las sumas de las Z individuales de los píxeles por columnas a lo ancho de todo el área de búsqueda, tomando como número de filas la altura del patch: B , según se indica en la Fig. 4.37.

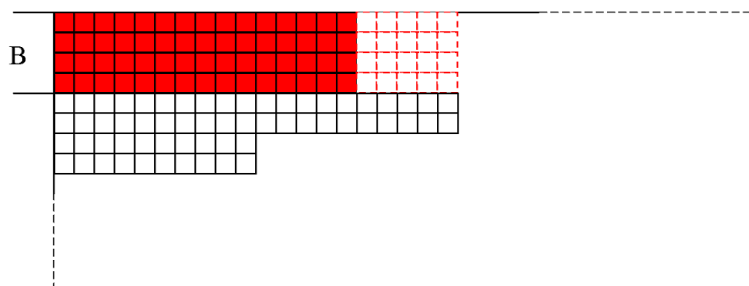


Figura 4.37: Procedimiento de detección de características

2. En segundo lugar, se calcula la Z total del primer patch a evaluar sumando los resultados anteriores para las B primeras columnas (ver Fig. 4.38).
3. A partir de este punto se realiza un proceso iterativo en el que para todo u de la región de búsqueda, tomando como referencia el último valor de Z calculado se resta el valor

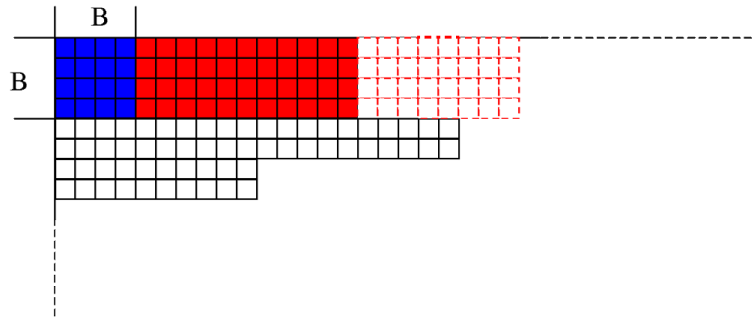


Figura 4.38: Procedimiento de detección de características

de la suma correspondiente a la columna anterior y se suma el valor correspondiente a la columna posterior. De esta forma queda calculado el valor de Z total para el siguiente patch a evaluar (ver Fig. 4.39).

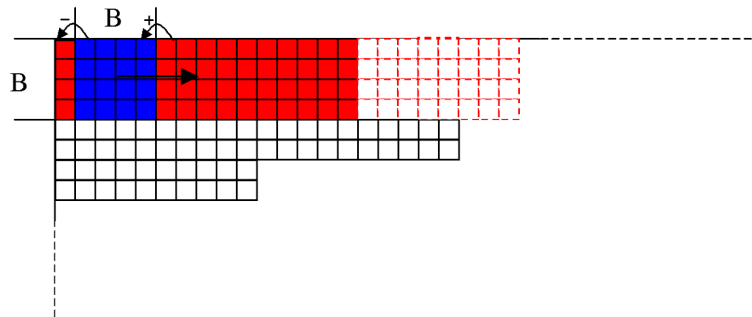


Figura 4.39: Procedimiento de detección de características

- Una vez obtenidos los valores de Z para todos los patches de la primera fila, el siguiente paso es calcular las sumas por columna para la segunda fila. Para ello, a las sumas calculadas en el paso 2 se les restan las Z individuales superiores y se les suman las Z individuales inferiores (ver Fig. 4.40).

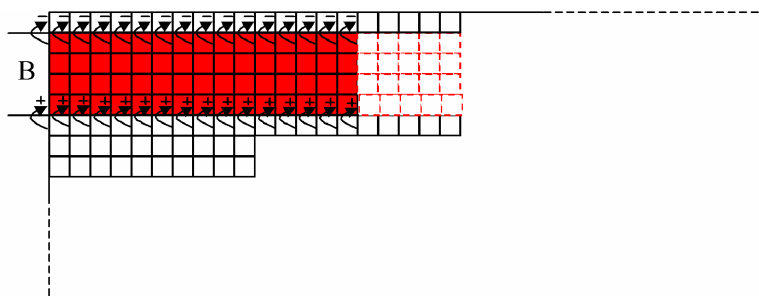


Figura 4.40: Procedimiento de detección de características

Con el cálculo de una nueva matriz Z se calculan sus autovalores, eligiéndose el mínimo. Si este valor es mayor que el anterior máximo global obtenido, éste se actualiza. Al final el patch correspondiente al máximo global será el elegido. Posteriormente se verificará, por medio de un umbral mínimo, para determinar si es aceptable o no para tomarlo como nueva marca.

4.9.2. Obtención del vector de estado de la marca

Para obtener, en primer lugar, las coordenadas del vector de estado de la marca Y_i se realizarán las siguientes tareas:

- a) *Búsqueda de la correspondencia epipolar.* Una vez obtenidas las coordenadas (u_L, v_L) de la marca en la cámara izquierda, hay que hallar las coordenadas (u_R, v_R) de la marca correspondiente en la cámara derecha. Para encontrar dichas coordenadas en la imagen derecha se debe buscar a lo largo de la denominada *recta epipolar*. Como se explicó anteriormente, a pesar de que la región de búsqueda vendría determinada por una región distinta a una recta, por razones de tiempo de cómputo se ha optado por esta aproximación, ya que el error cometido en distancias medias-altas es compensado por el margen de búsqueda (ver Fig. 4.41 derecha). Para su obtención se aplican los siguientes pasos:

1. El primero es calcular la ecuación de la mencionada recta epipolar como sigue:

$$ax + by + c = 0 \quad (4.65)$$

Para calcular los tres coeficientes se multiplicarán las coordenadas (u_L, v_L) de la cámara izquierda por la matriz fundamental F según la ecuación 4.66, tal y como se describe en [54].

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = F \begin{pmatrix} u_L \\ v_L \\ 1 \end{pmatrix} \quad F: \text{Matriz fundamental.} \quad (4.66)$$

Para calcular F hay que aplicar 4.67, donde C_L y C_R se obtienen según las expresiones 4.68 y 4.69.

$$F = (C_R^{-1})^T \cdot E \cdot C_L^{-1} \quad E: \text{Matriz esencial.} \quad (4.67)$$

$$C_L = \begin{pmatrix} FC1_L & 0 & CC1_L \\ 0 & FC2_L & CC2_L \\ 0 & 0 & 1 \end{pmatrix} \quad (4.68)$$

$$C_R = \begin{pmatrix} FC1_R & 0 & CC1_R \\ 0 & FC2_R & CC2_R \\ 0 & 0 & 1 \end{pmatrix} \quad (4.69)$$

- $FC1_L$ y $FC2_L$ se corresponden con las distancias focales de la cámara izquierda. $CC1_L$ y $CC2_L$ se corresponden con las coordenadas del *punto principal* de la cámara izquierda.
- $FC1_R$ y $FC2_R$ se corresponden con las distancias focales de la cámara derecha. $CC1_R$ y $CC2_R$ se corresponden con las coordenadas del *punto principal* de la cámara derecha.

La matriz esencial E relaciona los puntos en correspondencia de la imagen estéreo, asumiendo que las cámaras satisfacen el modelo pin-hole. Ésta se calcula según la ecuación 4.70.

$$E = \begin{pmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{pmatrix} \cdot R_{int} \quad (4.70)$$

- Una vez calculada la ecuación de la recta epipolar, el siguiente paso es, la búsqueda de correspondencia a lo largo de dicha recta. Es decir, la correlación de la imagen asociada a la marca tomada por la cámara izquierda (*patch* izquierdo), con imágenes de las mismas dimensiones situadas entorno a la recta epipolar en la imagen tomada por la cámara derecha, según se indica en la Fig. 4.41.

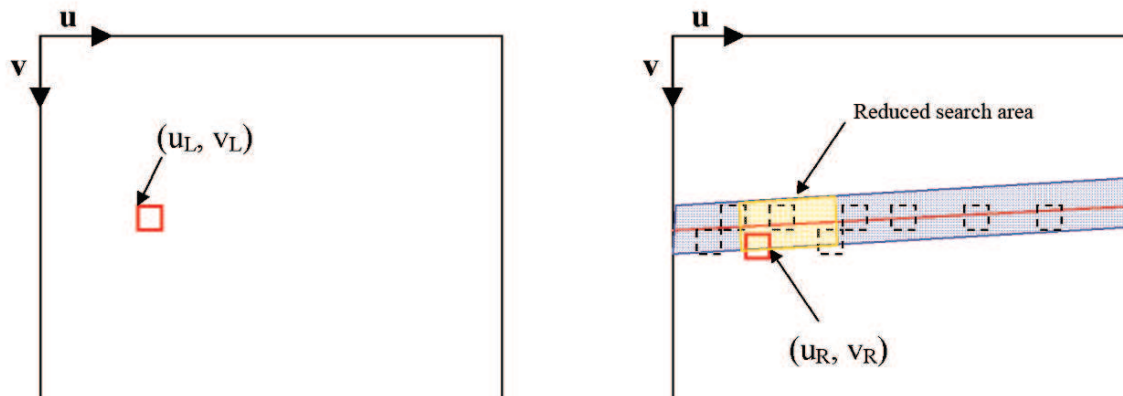


Figura 4.41: Búsqueda de correspondencia epipolar. La zona sombreada en amarillo representa el área reducida de búsqueda entorno a la proyección esperada en la imagen derecha

Para ello se toman muestras de *patches*, de forma consecutiva, para cada uno de los píxeles pertenecientes a la región definida por la recta. El método empleado para realizar la correlación es el mismo que fue descrito en el apartado 4.8.3.

Debido al margen de búsqueda por encima y por debajo de la recta epipolar, necesario para compensar el efecto de la distorsión, el tiempo de cómputo dedicado a realizar las correlaciones conllevaría un tiempo total superior al límite de tiempo real. En la Fig. 4.42 se puede apreciar que disparidades superiores a 130 píxeles sólo son necesarias para estimar la posición de las marcas en un rango de 0 - 5 m. Esto quiere decir que si se restringe el rango de distancias de estimación de las marcas para no inicializar las que se encuentren a menos de 5 m, se puede restringir la sección de la recta epipolar a sólo 130 píxeles.

Por otra parte, la gráfica de la Fig. 4.42 está calculada a partir de la ecuación aproximada 4.23, la cual no contempla la distorsión de las lentes. Tras incorporar el efecto de dicha distorsión, es posible reducir el margen de búsqueda hasta 35 píxeles para poder medir las posiciones de las marcas a partir de una distancia de aproximadamente 3 m. Teniendo en cuenta la ubicación de las cámaras, la distancia desde las mismas hasta el frontal del vehículo es de alrededor de 2 m. Por lo tanto, la probabilidad de observar nuevas marcas situadas a una distancia inferior a 3 m es extremadamente baja, siendo suficiente la restricción del área de búsqueda a 35 píxeles entorno a la correspondencia directa sobre la recta epipolar.

- Una vez hallado el punto de mejor correlación se comprueba si el valor de la misma supera un cierto umbral. Si es así, obtenemos las coordenadas (u_R, v_R) correspondientes para ser usadas en el siguiente paso, tal como se muestra en la Fig. 4.41.
- b) *Eliminación de la distorsión radial y tangencial.* Dado que a las imágenes capturadas por ambas cámaras no les ha sido eliminada la distorsión, para poder calcular la verdadera posición de la marca Y_i en el sistema del sub-mapa, se deben corregir las coordenadas (u_L, v_L) , (u_R, v_R) de forma que se obtengan sus equivalentes en un sistema no distorsionado. A estas nuevas coordenadas se las denominará (u_{LS}, v_{LS}) , (u_{RS}, v_{RS}) .

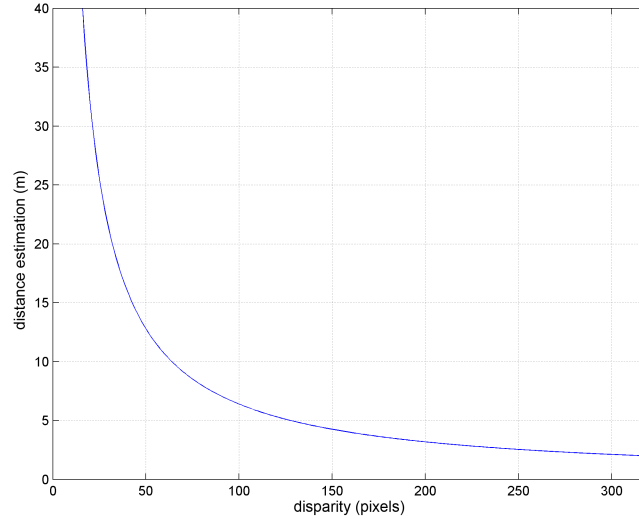


Figura 4.42: Distancia estimada de de una marca en función de la disparidad $d = u_L - u_R$

Tal y como se expone en [71], el proceso a seguir para corregir la distorsión de las coordenadas (u_L, v_L) en la cámara izquierda es el siguiente:

- a) En primer lugar se desnormalizan las coordenadas (u_L, v_L) para expresarlas en unidades métricas.

$$u_{LC} = -\frac{u_L}{FC1_L} + \frac{CC1_L}{FC1_L} \quad ; \quad v_{LC} = -\frac{v_L}{FC2_L} + \frac{CC2_L}{FC2_L} \quad (4.71)$$

- b) Partiendo de las coordenadas desnormalizadas, el siguiente paso es calcular las nuevas coordenadas corregidas. Para ello, se realiza el proceso iterativo que se detalla a continuación:

- 1) La suposición de partida es que $u_{LNC} = u_{LC}$ y $v_{LNC} = v_{LC}$, donde u_{LNC} y v_{LNC} son las coordenadas corregidas y aún sin normalizar.
- 2) A partir de ese punto se recalculan n veces las nuevas coordenadas corregidas despejándolas de la fórmula descrita en 4.33 y 4.34. Al quedar u_{LNC} y v_{LNC} en función de ellas mismas, se supone que éstas tienen el valor calculado en la iteración anterior tal como se muestra en las ecuaciones 4.72 y 4.73, donde: $r^2 = u_{LNC}^2 + v_{LNC}^2$ y $f = 1 + K_{1L}r^2 + K_{2L}r^4$.

$$u_{LNC} = \frac{u_{LC} - (2P_{1L}u_{LNC}v_{LNC} + P_{2L}(r^2 + 2u_{LNC}^2))}{f} \quad (4.72)$$

$$v_{LNC} = \frac{v_{LC} - (P_{1L}(r^2 + 2v_{LNC}^2) + 2P_{2L}u_{LNC}v_{LNC})}{f} \quad (4.73)$$

Donde, como se indicó anteriormente, K_{1L} , K_{2L} , P_{1L} y P_{2L} son los coeficientes de distorsión radial y tangencial respectivamente.

- c) Por último, una vez obtenidas las coordenadas ya corregidas, ya sólo queda normalizarlas para obtener las coordenadas equivalentes de píxel, según las ecuaciones expresadas en 4.74.

$$u_{LS} = -u_{LNC}FC1_L + CC1_L \quad ; \quad v_{LS} = -v_{LNC}FC2_L + CC2_L \quad (4.74)$$

Para el caso de la cámara derecha, el proceso se repite de idéntica forma al de la cámara izquierda pero sustituyendo los coeficientes de distorsión por los correspondientes a dicha cámara.

3. *Obtención de la posición Y_i de la marca en el sub-mapa.* Tal y como se describe en [54], se pueden relacionar las coordenadas (u,v) de la proyección de una marca en una de las cámaras con su posición $Y_i(x, y, z)$ en el sistema de coordenadas del sub-mapa. Tomando como ejemplo la proyección en la cámara izquierda, se calculan las coordenadas de imagen como se muestra en 4.75.

$$u_{LS} = \frac{m_{L11}x + m_{L12}y + m_{L13}z + m_{L14}}{m_{L31}x + m_{L32}y + m_{L33}z + m_{L34}}; \quad v_{LS} = \frac{m_{L21}x + m_{L22}y + m_{L23}z + m_{L24}}{m_{L31}x + m_{L32}y + m_{L33}z + m_{L34}} \quad (4.75)$$

De manera análoga, para la cámara derecha, se obtendrían las coordenadas según 4.76.

$$u_{RS} = \frac{m_{R11}x + m_{R12}y + m_{R13}z + m_{R14}}{m_{R31}x + m_{R32}y + m_{R33}z + m_{R34}}; \quad v_{RS} = \frac{m_{R21}x + m_{R22}y + m_{R23}z + m_{R24}}{m_{R31}x + m_{R32}y + m_{R33}z + m_{R34}} \quad (4.76)$$

Dado que el objetivo es obtener $Y_i(x, y, z)$, éstas serán las variables a despejar del sistema de ecuaciones anterior. Para ello se describen dichas ecuaciones según 4.77

$$\begin{cases} (u_{LS}m_{L31} - m_{L11})x + (u_{LS}m_{L32} - m_{L12})y + (u_{LS}m_{L33} - m_{L13})z = m_{L14} - u_{LS}m_{L34} \\ (v_{LS}m_{L31} - m_{L21})x + (v_{LS}m_{L32} - m_{L22})y + (v_{LS}m_{L33} - m_{L23})z = m_{L24} - v_{LS}m_{L34} \\ (u_{RS}m_{R31} - m_{R11})x + (u_{RS}m_{R32} - m_{R12})y + (u_{RS}m_{R33} - m_{R13})z = m_{R14} - u_{RS}m_{R34} \\ (v_{RS}m_{R31} - m_{R21})x + (v_{RS}m_{R32} - m_{R22})y + (v_{RS}m_{R33} - m_{R23})z = m_{R24} - v_{RS}m_{R34} \end{cases} \quad (4.77)$$

El sistema anterior es un sistema sobredeterminado de la forma $A \cdot Y_i = b$, donde A y b se construyen según las expresiones 4.78 y 4.79, respectivamente. Y_i , por su parte, se obtiene despejando como se muestra en 4.80.

$$A = \begin{pmatrix} u_{LS}m_{L31} - m_{L11} & u_{LS}m_{L32} - m_{L12} & u_{LS}m_{L33} - m_{L13} \\ v_{LS}m_{L31} - m_{L21} & v_{LS}m_{L32} - m_{L22} & v_{LS}m_{L33} - m_{L23} \\ u_{RS}m_{R31} - m_{R11} & u_{RS}m_{R32} - m_{R12} & u_{RS}m_{R33} - m_{R13} \\ v_{RS}m_{R31} - m_{R21} & v_{RS}m_{R32} - m_{R22} & v_{RS}m_{R33} - m_{R23} \end{pmatrix} \quad (4.78)$$

$$b = \begin{pmatrix} m_{L14} - u_{LS}m_{L34} \\ m_{L24} - v_{LS}m_{L34} \\ m_{R14} - u_{RS}m_{R34} \\ m_{R24} - v_{RS}m_{R34} \end{pmatrix} \quad (4.79)$$

$$Y_i = (A^T A)^{-1} A^T b \quad (4.80)$$

Para resolver este sistema se necesita conocer el valor de los coeficientes m_{Lij} y m_{Rij} , es decir, las matrices de proyección perspectiva M_L y M_R .

M_L se corresponde con la matriz de proyección en la cámara izquierda y viene definida por 4.81.

$$M_L = \begin{pmatrix} FC1_L & 0 & CC1_L & 0 \\ 0 & FC2_L & CC2_L & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} r_{inv11} & r_{inv12} & r_{inv13} & -x_{rob} \\ r_{inv21} & r_{inv22} & r_{inv23} & -y_{rob} \\ r_{inv31} & r_{inv32} & r_{inv33} & -z_{rob} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.81)$$

R_{inv} es la matriz de rotación inversa del vehículo respecto del sistema de referencia del sub-mapa, y se obtiene como muestra 4.82.

$$R_{inv} = R_{rob}^{-1} = \begin{pmatrix} r_{inv11} & r_{inv12} & r_{inv13} \\ r_{inv21} & r_{inv22} & r_{inv23} \\ r_{inv31} & r_{inv32} & r_{inv33} \end{pmatrix} \quad (4.82)$$

Dado que el sistema de referencia del vehículo es coincidente con el sistema de referencia de la cámara izquierda, la única transformación a aplicar es entre el sistema del sub-mapa y el propio del vehículo.

Sin embargo, a la hora de obtener la matriz M_R para la cámara derecha, es necesario aplicar una transformación adicional correspondiente a la relación entre la cámara derecha y la izquierda, según se muestra en 4.83.

$$M_R = \begin{pmatrix} FC1_R & 0 & CC1_R & 0 \\ 0 & FC2_R & CC2_R & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} r_{int(inv)11} & r_{int(inv)12} & r_{int(inv)13} & 0 \\ r_{int(inv)21} & r_{int(inv)22} & r_{int(inv)23} & 0 \\ r_{int(inv)31} & r_{int(inv)32} & r_{int(inv)33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -T_x \\ 0 & 1 & 0 & -T_y \\ 0 & 0 & 1 & -T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{inv11} & r_{inv12} & r_{inv13} & -x_{rob} \\ r_{inv21} & r_{inv22} & r_{inv23} & -y_{rob} \\ r_{inv31} & r_{inv32} & r_{inv33} & -z_{rob} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.83)$$

$R_{int(inv)}$ es la matriz de rotación inversa *cámara derecha* \rightarrow *cámara izquierda* y se obtiene según 4.84

$$R_{int(inv)} = R_{int}^{-1} = \begin{pmatrix} r_{int(inv)11} & r_{int(inv)12} & r_{int(inv)13} \\ r_{int(inv)21} & r_{int(inv)22} & r_{int(inv)23} \\ r_{int(inv)31} & r_{int(inv)32} & r_{int(inv)33} \end{pmatrix} \quad (4.84)$$

Por tanto, siempre que se quiera obtener el vector Y_i será necesario recalculer las matrices M_L y M_R para describirlas en función de la posición del vehículo en el sistema de referencia del sub-mapa.

4. *Obtención de la posición 3D estimada del patch.* En el momento de la inicialización de la marca, además de la posición de ésta en el sub-mapa, de cara a calcular la transformación equivalente de la apariencia, será necesario obtener la posición estimada de todos los píxeles pertenecientes al patch.

Para obtener las posiciones de todos los píxeles sería necesario realizar el cálculo de correspondencia epipolar en cada uno de ellos, tal y como se hace con la posición de la marca en sí. Dado el tamaño del patch (10 x 10 píxeles), el tiempo de procesamiento necesario para realizar las correlaciones sería excesivo. Como solución aproximada, se parte de la suposición de que el patch se encuentra orientado paralelo al plano de proyección de la cámara (ver Fig. 4.43).

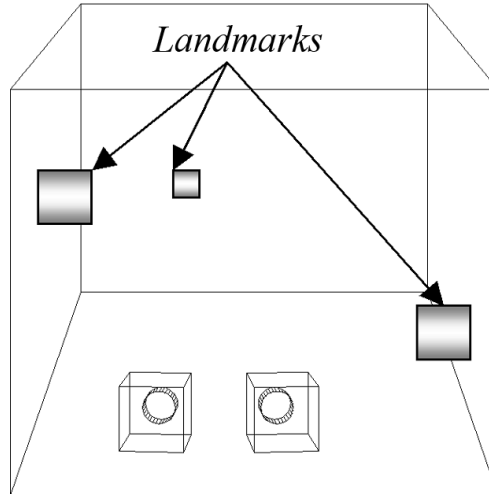


Figura 4.43: Suposición de la situación de los patches respecto a la cámara

Teniendo en cuenta la anterior suposición, se pueden estimar las proyecciones de cada uno de los puntos pertenecientes al patch usando el modelo de proyección. A partir de estos datos, se pueden obtener las posiciones 3D de todos los puntos siguiendo el procedimiento descrito en el apartado anterior.

4.9.3. Obtención de la covarianza del vector de estado de la marca

A la hora de capturar una nueva marca e incluirla en el filtro de Kalman se necesitan tres tipos de covarianzas parciales:

- En primer lugar P_{XY} , es decir, la covarianza del vector de estado de la nueva marca Y_i con el vector de estado del vehículo completo X_v . (Ver apartado 4.2).
- En segundo lugar las covarianzas cruzadas de la nueva marca Y_i con el resto de marcas ya almacenadas Y_j , es decir: $P_{Y_j Y_i}$ y $P_{Y_i Y_j}$.
- Por último, la autocovarianza P_{YY} .

Los dos primeros tipos de covarianzas serán estudiados en el siguiente apartado. El presente apartado se dedicará a la obtención de P_{YY} .

La autocovarianza de Y_i está compuesta por dos componentes:

- a) En primer lugar la debida a la incertidumbre en el estado del vehículo, es decir la autocovarianza P_{XX} . Para su cálculo se debe transformar dicha incertidumbre en la correspondiente a la determinación de la situación (vector de estado) de la nueva marca Y_i . Esto se puede ver en la ecuación 4.85.

$$\frac{\partial Y_i}{\partial X_v} P_{XX} \left(\frac{\partial Y_i}{\partial X_v} \right)^T \quad (4.85)$$

- b) En segundo lugar una componente de ruido aleatorio debida a la incertidumbre en la propia medida realizada por las cámaras, según la ecuación 4.86.

$$\frac{\partial Y_i}{\partial h_i} R_i \left(\frac{\partial Y_i}{\partial h_i} \right)^T \quad (4.86)$$

Por lo tanto, para calcular la autocovarianza P_{YY} se empleará la ecuación 4.87.

$$P_{YY} = \frac{\partial Y_i}{\partial X_v} P_{XX} \left(\frac{\partial Y_i}{\partial X_v} \right)^T + \frac{\partial Y_i}{\partial h_i} R_i \left(\frac{\partial Y_i}{\partial h_i} \right)^T \quad (4.87)$$

Es decir, a la hora de inicializar una nueva marca es necesario calcular los jacobianos del estado de la marca respecto al estado del vehículo $\frac{\partial Y_i}{\partial X_v}$ y $\frac{\partial Y_i}{\partial h_i}$, así como la matriz de ruido de medida R_i . A continuación se describen cada uno de ellos:

1. *Cálculo de $\frac{\partial Y_i}{\partial h_i}$* . Para obtener este jacobiano, se puede dividir el problema en dos partes:

- a) En primer lugar se puede obtener $\frac{\partial h_i}{\partial Y_i}$ sabiendo que: $h_i = R_{rob}^{-1}(Y_i - X_{rob})$ y calculando el jacobiano, según la ecuación 4.88.

$$\frac{\partial h_i}{\partial Y_i} = R_{rob}^{-1} \quad (4.88)$$

- b) Teniendo en cuenta que $\frac{\partial h_i}{\partial Y_i}$ es invertible, se obtiene el jacobiano $\frac{\partial Y_i}{\partial h_i}$ a partir de la ecuación 4.89.

$$\frac{\partial Y_i}{\partial h_i} = \left(\frac{\partial h_i}{\partial Y_i} \right)^{-1} \quad (4.89)$$

2. *Cálculo de $\frac{\partial Y_i}{\partial X_v}$* . En este caso, en primer lugar se obtendrá el jacobiano $\frac{\partial Y_i}{\partial X_p}$.

- a) El primer paso será calcular el jacobiano de la predicción respecto a la pose del vehículo $\frac{\partial h_i}{\partial X_p}$. Dado que el vector X_p está compuesto por el vector de posición X_{rob} y el vector de rotación q_{rob} (ver apartado 4.2), dicho jacobiano puede dividirse en dos partes, tal como se muestra en la ecuación 4.90.

$$\frac{\partial h_i}{\partial X_p} \begin{matrix} \nearrow \\ \searrow \end{matrix} \left(\begin{array}{c} \frac{\partial h_i}{\partial X_{rob}} \\ \frac{\partial h_i}{\partial q_{rob}} \end{array} \right) \quad (4.90)$$

Para el cálculo de $\frac{\partial h_i}{\partial X_{rob}}$ se puede usar la misma estrategia usada anteriormente ¹. Es decir, sabiendo que: $h_i = R_{rob}^{-1}(Y_i - X_{rob})$, se puede expresar como:

$$\frac{\partial h_i}{\partial X_{rob}} = -R_{rob}^{-1} \quad (4.91)$$

Para la obtención de $\frac{\partial h_i}{\partial q_{rob}}$ se necesita tener en cuenta la relación entre el vector de rotación q_{rob} y la matriz de rotación R_{rob} (ver apartado 4.2). Esto se refleja en la expresión 4.92.

¹El signo “-” de la ecuación 4.91 implica multiplicar todos los términos de la matriz por “-1”.

$$R_{rob} = \begin{pmatrix} q_0^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_0 q_z) & 2(q_x q_z + q_0 q_y) \\ 2(q_x q_y + q_0 q_z) & q_0^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_0 q_x) \\ 2(q_x q_z - q_0 q_y) & 2(q_y q_z + q_0 q_x) & q_0^2 - q_x^2 - q_y^2 + q_z^2 \end{pmatrix} \quad (4.92)$$

Despejando $h_i = R_{rob}^{-1}(Y_i - X_{rob})$, se obtiene:

$$\frac{\partial h_i}{\partial q_{rob}} = \frac{\partial R_{rob}^{-1}}{\partial q_{rob}^{-1}} (Y_i - X_{rob}) \frac{\partial q_{rob}^{-1}}{\partial q_{rob}} \quad (4.93)$$

Por último, es posible obtener $\frac{\partial Y_i}{\partial X_p}$ a partir de $\frac{\partial h_i}{\partial X_p}$ y $\frac{\partial Y_i}{\partial h_i}$, calculados anteriormente, según 4.94.

$$\frac{\partial Y_i}{\partial X_p} = \frac{\partial Y_i}{\partial h_i} \cdot \frac{\partial h_i}{\partial X_p} \quad (4.94)$$

b) Una vez calculado $\frac{\partial Y_i}{\partial X_p}$ se puede obtener $\frac{\partial Y_i}{\partial X_v}$ de la siguiente forma:

$$\frac{\partial Y_i}{\partial X_v} = \frac{\partial Y_i}{\partial X_p} \cdot \frac{\partial X_p}{\partial X_v} \quad (4.95)$$

La obtención de $\frac{\partial X_p}{\partial X_v}$ proporcionará dos posibles valores:

- **1:** para los términos de derivadas parciales respecto a sí mismos: $\frac{\partial x_{rob}}{\partial x_{rob}}$, $\frac{\partial y_{rob}}{\partial y_{rob}}$, $\frac{\partial z_{rob}}{\partial z_{rob}}$, $\frac{\partial q_0}{\partial q_0}$, $\frac{\partial q_x}{\partial q_x}$, $\frac{\partial q_y}{\partial q_y}$ y $\frac{\partial q_z}{\partial q_z}$
- **0:** para el resto de términos.

3. *Cálculo de R_i .* La matriz de ruido de medida expresa la incertidumbre a la hora de realizar dicha medida. Tal como se define en el apartado 4.2, el vector de medida está formado por las coordenadas de las diferentes marcas respecto al sistema de referencia del vehículo h_i . Debido a que la obtención del vector de observación se realiza de forma indirecta, no se puede predecir a priori dicha matriz de ruido.

a) Tal y como se explicó en el apartado 4.9.2, la observación parte de la medida de las coordenadas (u_L, v_L) , (u_R, v_R) correspondientes a cada una de las diferentes marcas detectadas por el sistema. A priori se puede suponer que la incertidumbre, a la hora de realizar dicha medida, se basa en la indeterminación en saber si la proyección de la marca se encuentra en un píxel o el adyacente (ver Fig. 4.44).

A partir de aquí se puede deducir que la incertidumbre en la medida es de ± 1 píxel. Además de esto se puede suponer que dicha indeterminación es independiente, tanto en la dirección u como en la v . En base a estas suposiciones se puede construir un vector formado por los dos pares de coordenadas de imagen de cada marca, según la ecuación 4.96.

$$T_i = \begin{pmatrix} u_L \\ v_L \\ u_R \\ v_R \end{pmatrix} \quad (4.96)$$

Tratando a las cuatro variables de dicho vector como aleatorias de tipo gaussiano, se puede definir un valor medio (representado por las mismas) y una desviación típica de valor 1 píxel ($\sigma = 1$). Por lo tanto, es posible definir una matriz de ruido independiente a partir de las cuatro variables aleatorias, como muestra 4.97.

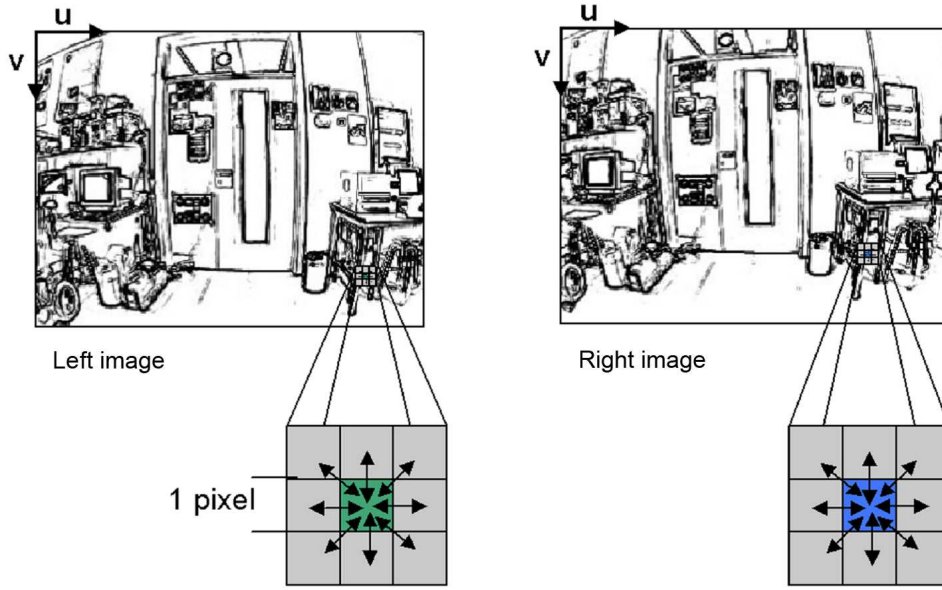


Figura 4.44: Representación del ruido de medida de 1 pixel

$$R_{Ti} = \begin{pmatrix} \sigma_{uL}^2 & 0 & 0 & 0 \\ 0 & \sigma_{vL}^2 & 0 & 0 \\ 0 & 0 & \sigma_{uR}^2 & 0 \\ 0 & 0 & 0 & \sigma_{vR}^2 \end{pmatrix} \quad (4.97)$$

- b) Para calcular la matriz R_i correspondiente al vector de observación h_i se deberá realizar la transformación mostrada en 4.98.

$$R_i = \frac{\partial h_i}{\partial T_i} \cdot R_{Ti} \cdot \left(\frac{\partial h_i}{\partial T_i} \right)^T \quad (4.98)$$

El problema, ahora, consiste en obtener el jacobiano $\frac{\partial h_i}{\partial T_i}$. Para ello se parte de la ecuación para obtener las coordenadas de cada marca dentro del sub-mapa descrita en el apartado 4.9.2, obteniendo la expresión 4.99.

$$A_{i0} \cdot h_i = b_{i0} \quad (4.99)$$

En este caso se desea expresar dicha ecuación en función del vector de observación h_i , es decir, las coordenadas de la marca en el sistema de referencia del vehículo. Por lo tanto es necesario utilizar las matrices A_{i0} y b_{i0} calculadas a partir de las matrices M_{L0} y M_{R0} , obtenidas expresando M_L y M_R en el sistema de referencia del vehículo. Tal como se describe en el apartado 4.9.2, para obtener las matrices A_{i0} y b_{i0} es necesario conocer las coordenadas de proyección (u_{LS}, v_{LS}) , (u_{RS}, v_{RS}) . El proceso a seguir para su cálculo es el seguido en el apartado 4.8.2, quedándose en el paso anterior a la aplicación del modelo de distorsion.

El siguiente paso es calcular los jacobianos de los vectores a ambos lados de la ecuación, según indica 4.100.

$$\frac{\partial (A_{i0} \cdot h_i)}{\partial T_{iS}} = \frac{\partial b_{i0}}{\partial T_{iS}} \quad (4.100)$$

T_{iS} es un vector columna como muestra la ecuación 4.101.

$$T_{iS} = \begin{pmatrix} u_{LS} \\ v_{LS} \\ u_{RS} \\ v_{RS} \end{pmatrix} \quad (4.101)$$

Desarrollando los jacobianos anteriores, se obtiene un sistema de doce ecuaciones con doce incógnitas. De estas ecuaciones se obtienen los doce elementos del jacobiano $\frac{\partial h_i}{\partial T_{iS}}$.

Reagrupando las ecuaciones se puede expresar el sistema según 4.102.

$$A_{i0} \cdot \frac{\partial h_i}{\partial T_{iS}} = C_i \quad (4.102)$$

La matriz diagonal C_i se forma como se muestra en 4.103.

$$C_i = \begin{pmatrix} (a) & 0 & 0 & 0 \\ 0 & (a) & 0 & 0 \\ 0 & 0 & (b) & 0 \\ 0 & 0 & 0 & (b) \end{pmatrix} \quad (4.103)$$

Los términos (a) y (b) se muestran desarrollados en 4.104.

$$\begin{cases} (a) = -m_{L31}h_{ix} - m_{L32}h_{iy} - m_{L33}h_{iz} - m_{L34} \\ (b) = -m_{R31}h_{ix} - m_{R32}h_{iy} - m_{R33}h_{iz} - m_{R34} \end{cases} \quad (4.104)$$

Por lo tanto, se obtiene el jacobiano $\frac{\partial h_i}{\partial T_{iS}}$ según la expresión 4.105.

$$\frac{\partial h_i}{\partial T_{iS}} = (A_{i0}^T A_{i0})^{-1} \cdot A_{i0}^T \cdot C_i \quad (4.105)$$

Para obtener $\frac{\partial h_i}{\partial T_i}$ se puede dividir $\frac{\partial h_i}{\partial T_{iS}}$ en dos partes, como se muestra en 4.106.

$$\frac{\partial h_i}{\partial T_{iS}} \longrightarrow \begin{pmatrix} \frac{\partial h_i}{\partial T_{iSL}} \\ \frac{\partial h_i}{\partial T_{iSR}} \end{pmatrix} \quad (4.106)$$

Los términos T_{iSL} y T_{iSR} se definen como:

$$\begin{cases} T_{iSL} = \begin{pmatrix} u_{LS} \\ v_{LS} \end{pmatrix} \\ T_{iSR} = \begin{pmatrix} u_{RS} \\ v_{RS} \end{pmatrix} \end{cases} \quad (4.107)$$

De esta forma, el jacobiano $\frac{\partial h_i}{\partial T_i}$ se puede obtener como se indica en la expresión 4.108.

$$\begin{cases} \frac{\partial h_i}{\partial T_{iL}} = \frac{\partial h_i}{\partial T_{iSL}} \cdot \frac{\partial U_{LS}}{\partial U_L} \\ \frac{\partial h_i}{\partial T_{iR}} = \frac{\partial h_i}{\partial T_{iSR}} \cdot \frac{\partial U_{RS}}{\partial U_R} \end{cases} \implies \frac{\partial h_i}{\partial T_i} = \begin{pmatrix} \frac{\partial h_i}{\partial T_{iL}} \\ \frac{\partial h_i}{\partial T_{iR}} \end{pmatrix} \quad (4.108)$$

Para obtener $\frac{\partial U_{LS}}{\partial U_L}$ y $\frac{\partial U_{RS}}{\partial U_R}$ habrá que invertir los jacobianos $\frac{\partial U_L}{\partial U_{LS}}$ y $\frac{\partial U_R}{\partial U_{RS}}$ obtenidos previamente en la ecuación 4.36 y su equivalente para la cámara derecha.

4.9.4. Adaptación del vector de estado completo y su covarianza

Una vez obtenidos los elementos necesarios para formar la nueva matriz de covarianza P es posible construir dicha matriz de la forma expresada en 4.109.

$$P = \begin{pmatrix} P_{XX} & P_{XY_1} & P_{XY_2} & \cdots & P_{XX} \cdot \left(\frac{\partial Y_i}{\partial X_v}\right)^T \\ P_{Y_1X} & P_{Y_1Y_1} & P_{Y_1Y_2} & \cdots & P_{Y_1X} \cdot \left(\frac{\partial Y_i}{\partial X_v}\right)^T \\ P_{Y_2X} & P_{Y_2Y_1} & P_{Y_2Y_2} & \cdots & P_{Y_2X} \cdot \left(\frac{\partial Y_i}{\partial X_v}\right)^T \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Y_i}{\partial X_v} \cdot P_{XX} & \frac{\partial Y_i}{\partial X_v} \cdot P_{XY_1} & \frac{\partial Y_i}{\partial X_v} \cdot P_{XY_2} & \cdots & \frac{\partial Y_i}{\partial X_v} P_{XX} \left(\frac{\partial Y_i}{\partial X_v}\right)^T + \frac{\partial Y_i}{\partial h_i} R_i \left(\frac{\partial Y_i}{\partial h_i}\right)^T \end{pmatrix} \quad (4.109)$$

Es decir, se trata de añadir una nueva columna por la derecha y una nueva fila por la parte inferior en base a los elementos obtenidos en el apartado anterior.

La adaptación del vector de estado es trivial, ya que simplemente será necesario añadir el vector de estado de la nueva marca como nueva fila al vector de estado completo (ver ecuación 4.110).

$$X = \begin{pmatrix} X_v \\ Y_1 \\ Y_2 \\ \vdots \\ Y_i \end{pmatrix} \quad (4.110)$$

4.9.5. Eliminación de marcas

Por último, a la hora de eliminar una marca del proceso del filtro, bastará con eliminar su fila y columna correspondiente en la matriz de covarianza total P . Como ejemplo, la ecuación 4.111 muestra el efecto de eliminar la marca número 1 en la matriz P .

$$P = \begin{pmatrix} P_{XX} & P_{XY_1} & P_{XY_2} & \cdots & P_{XY_n} \\ P_{Y_1X} & P_{Y_1Y_1} & P_{Y_1Y_2} & \cdots & P_{Y_1Y_n} \\ P_{Y_2X} & P_{Y_2Y_1} & P_{Y_2Y_2} & \cdots & P_{Y_2Y_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P_{Y_nX} & P_{Y_nY_1} & P_{Y_nY_2} & \cdots & P_{Y_nY_n} \end{pmatrix} \implies P = \begin{pmatrix} P_{XX} & P_{XY_2} & \cdots & P_{XY_n} \\ P_{Y_2X} & P_{Y_2Y_2} & \cdots & P_{Y_2Y_n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{Y_nX} & P_{Y_nY_2} & \cdots & P_{Y_nY_n} \end{pmatrix} \quad (4.111)$$

Respecto al vector de estado, será necesario eliminar la marca de su posición correspondiente, como muestra 4.112.

$$X = \begin{pmatrix} X_v \\ Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} \implies X = \begin{pmatrix} X_v \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} \quad (4.112)$$

4.10. Conclusiones y aportaciones

Tras la descripción y el análisis del método SLAM de bajo nivel implementado, se extraen las conclusiones que se muestran a continuación:

- Se ha presentado el vector de estado del filtro EKF, específico para el sistema desarrollado en esta Tesis, así como la estructura interna de cada sub-mapa. Asimismo, se han descrito, por un lado, el modelo de movimiento y, por otro, el modelo de medida, aplicados al filtro.
- Se ha realizado un análisis pormenorizado de la configuración óptima para la cámara estéreo. En primer lugar se ha estudiado la mejor resolución para las cámaras, teniendo en cuenta los criterios contrapuestos de precisión en la estimación y tiempo de cómputo necesario, llegando a la conclusión de la utilización de una resolución de 320x240. En cuanto a la separación entre las cámaras (*baseline*), se han estudiado aspectos como el error en la determinación de la distancia frente al campo visual, obteniendo un intervalo óptimo de 30 a 40 cm para entornos exteriores. Por último, se han analizado las implicaciones del ángulo de apertura de las lentes, llegando a la conclusión de que éste deberá ser lo más amplio posible, teniendo en cuenta la implicación de una posterior corrección de la distorsión.
- Se ha descrito el proceso de calibración de las cámaras, teniendo en cuenta la utilización de un par estéreo y cámaras de gran angular. Además, se ha descrito el modelo de distorsión *radial* y *tangencial* utilizado, demostrando sus ventajas frente a otras alternativas.
- Se ha descrito detalladamente el modelo de medida, teniendo en cuenta las particularidades del uso de una cámara estéreo. En cuanto al tipo de marcas a utilizar, se ha realizado un estudio de dos alternativas: marcas tipo SIFT y marcas basadas en correlación directa. La conclusión obtenida es que únicamente utilizando la correlación directa se garantiza el cumplimiento del límite de tiempo real. A cambio, utilizando marcas SIFT la estabilidad en las correlaciones es mayor, pero únicamente si se elige un área de búsqueda muy elevado, excediéndose ampliamente el límite de tiempo real. Por lo tanto, el tipo de marcas más adecuado resulta ser las basadas en correlación directa.
- Para mejorar el proceso de correlación de las marcas, se ha desarrollado un método de transformación del *patch* asociado a cada marca, basado en interpolación. Se demuestra una mejora en la eficiencia de utilización de las marcas.
- En cuanto a la inicialización de marcas, se ha desarrollado la matemática necesaria para la obtención de sus vectores de estado, así como las covarianzas de incertidumbre.

En cuanto a las principales aportaciones de este capítulo, éstas se resumen en:

- En primer lugar se ha desarrollado una metodología completa para la implementación del método de SLAM de bajo nivel basado en marcas visuales y en el filtro EKF, adaptada específicamente para el uso de una cámara estéreo de gran *baseline* y con gran angular.

- Se ha implementado un modelo de distorsión de lentes basado en dos parámetros de distorsión radial y otros dos de distorsión tangencial, los cuales proporcionan una mayor precisión en la corrección de dicha distorsión que otros métodos existentes en el estado del arte [61]
- Se ha adaptado el modelo de movimiento aleatorio básico para seis grados de libertad propuesto por A. Davison para el uso de un vehículo sobre un terreno fundamentalmente plano.
- Se ha propuesto una mejora en el método de búsqueda de correspondencia epipolar en la inicialización de marcas, acortando sensiblemente la longitud de la línea epipolar. Se demuestra la mejora del tiempo necesario para realizar dicha inicialización.
- Además, se ha desarrollado un método de adaptación de apariencia de los patches de las marcas, aplicable para el movimiento de vehículos en entornos exteriores fundamentalmente, que mejora la eficiencia en el uso de las mismas.

Capítulo 5

SLAM de alto nivel

5.1. Introducción

Tal como se explicó en el capítulo 3 de Visión General del Sistema, para abordar el problema de SLAM en grandes entornos, se propone dividir el mapa global en diferentes sub-mapas locales, definiendo sobre éstos un nivel superior de SLAM topológico que los relacione, manteniendo la consistencia del mapa global, tal como se muestra en la Fig. 5.1.

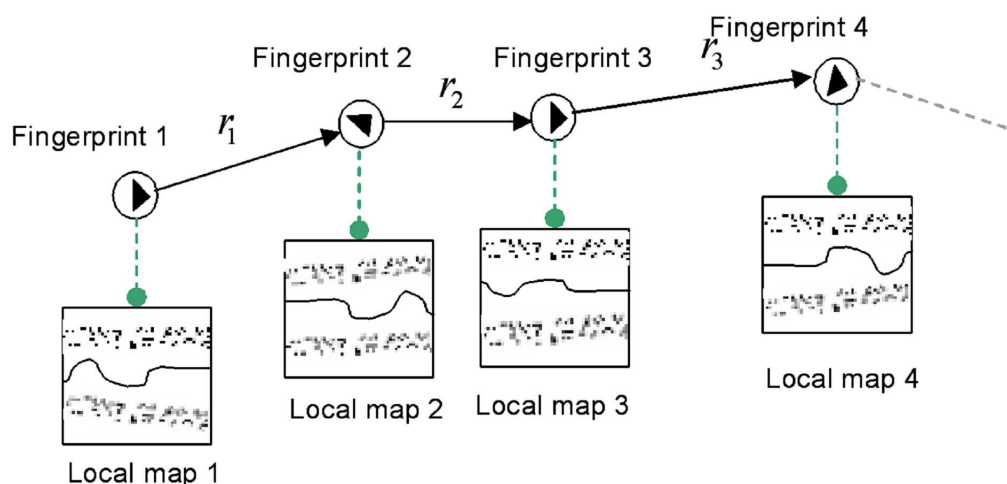


Figura 5.1: Representación gráfica del método híbrido de SLAM de dos niveles jerárquicos. Cada sub-mapa tiene una huella asociada a éste

El mapa global se dividirá, por tanto, en sub-mapas locales independientes e identificados por las denominadas *Huellas*. Estas huellas almacenan la pose del vehículo en el momento de la creación del sub-mapa y definen su sistema de referencia local, tal como se muestra en la Fig. 5.2

La generación de sub-mapas se realiza periódicamente en el espacio, de tal forma que después de recorrer un determinado tramo del recorrido, se crea un nuevo sub-mapa y se asocia una nueva huella. Para la generación de sub-mapas se deben tener en cuenta los siguientes aspectos:

- Si el vehículo se está moviendo entre dos huellas (*nodos*), se insertará una nueva *transición* para conectarlos entre sí, lo que representará un enlace entre las dos poses. Por otra parte, las transiciones almacenan las matrices de transformación e incertidumbres que describen la relación entre las huellas conectadas.
- Para optimizar la detección de cierre de lazos, cuando se detecta un cambio significativo en la dirección del vehículo, se crea una huella adicional denominada *Huella SIFT*. Este tipo de huella añade a la información de pose cierta información visual que identifique el lugar donde ha sido creada.
- Para detectar zonas previsitadas se realiza un proceso de comparación (*matching*) entre las huellas SIFT previamente capturadas y la huella actual, dentro de una región de incertidumbre.
- En caso de un *matching* positivo se detectará una situación de cierre de lazo y el mapa topológico se corregirá empleando el método MLR (*MultiLevel Relaxation*, “Relajación multinivel”) [41] aplicado a todo el conjunto de huellas. El MLR determina la estimación de máxima verosimilitud del conjunto de poses asociadas a las huellas.

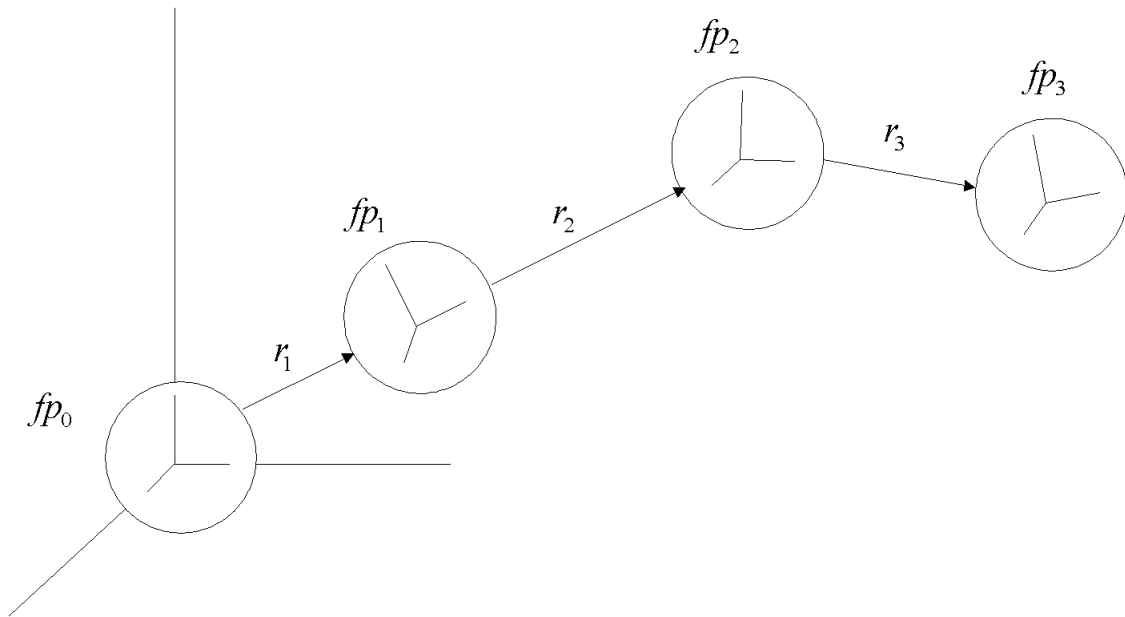


Figura 5.2: Estructura básica del mapa topológico. Cada huella (nodo) lleva asociado el sistema de referencia local de su propio submapa métrico. El sistema de referencia global es coincidente con el sistema local de la primera huella.

- Posteriormente, las marcas de cada sub-mapa se corrigen en función de la corrección aplicada a sus huellas asociadas.

5.2. Implementación

La implementación del sistema de SLAM propuesto añade un nivel topológico adicional, denominado High Level SLAM al Low Level SLAM, con el objeto de mantener la consistencia global del mapa con un tiempo de cómputo casi constante. Este objetivo se consigue usando el algoritmo MLR aplicado a las denominadas *Huellas*. Por lo tanto, el mapa global se dividirá en sub-mapas locales, identificados por las mencionadas huellas. Se definen dos tipos de huellas: *Huellas Ordinarias* y *Huellas SIFT*.

Las del primer tipo se denotan como $FP = \{fp_i | i \in 0..L\}$. Su propósito es almacenar la pose local del vehículo $X_{rob}^{fp_i}$ y su correspondiente covarianza local $P_{rob}^{fp_i}$ relativa a la huella anterior, es decir, el sistema de referencia del sub-mapa actual. Para definir el tamaño del sub-mapa se tienen en cuenta dos aspectos principales:

- Uno está relacionado con el problema de la no linealidad. Es conocido que la linealización del EKF se puede asumir sólo en entornos de tamaño limitado. Por lo tanto, para hacer frente a este problema se limitará el tamaño de los sub-mapas para mantener el error de linealización lo suficientemente bajo, tal como se explica en [36].

Además, se ha comprobado que manteniendo un tamaño constante, en términos de camino recorrido, se obtiene una mayor consistencia en los resultados de la reconstrucción de alto nivel del mapa global.

- El otro aspecto a tener en cuenta es tratar de mantener el tiempo de proceso por debajo de la restricción de tiempo real. Esto implica un límite en el número de marcas procesadas

en el filtro de bajo nivel. Experimentalmente se observa que procesando hasta 60 marcas por sub-mapa no se excede este límite.

Por lo tanto, se ha escogido como tamaño de submapa adecuado 10 m de camino recorrido, de forma que cada 10 m recorridos por el vehículo se tomará una nueva huella ordinaria.

El segundo tipo de huellas son un subconjunto de las primeras y se denotan como: $SF = \{sf_q \in FP | q \in 0..Q, Q < L\}$. La funcionalidad añadida de este tipo de huellas es la de almacenar la apariencia visual del entorno en el momento en el que han sido obtenidas. Esto se lleva a cabo definiendo un conjunto de *marcas SIFT* asociadas a la huella, las cuales identifican el lugar en ese momento $YF^q = \{Yf_m^q | m \in 0..M\}$. Este tipo de huellas se toman sólo en caso de que el vehículo haya sufrido un cambio significativo en su trayectoria, ya que en una aplicación como la abordada en esta Tesis, donde el entorno es semi-estructurado, es en estos momentos cuando se puede detectar una zona previsitada. Las condiciones para la detección de dicho cambio se dividen en dos etapas:

1. En primer lugar el vehículo debe sufrir un cambio en su orientación $\Delta\theta_1 \geq \gamma_{\text{máx}}$ en un determinado periodo de tiempo (Fig. 5.3).
2. En segundo lugar, para obtener un punto de vista más estable cada vez que el vehículo vuelva a visitar el mismo lugar, se espera para realizar la captura de la huella SIFT hasta que la variación en la orientación caiga por debajo de un determinado umbral $\Delta\theta_2 \leq \gamma_{\text{mín}}$. (Fig. 5.3)

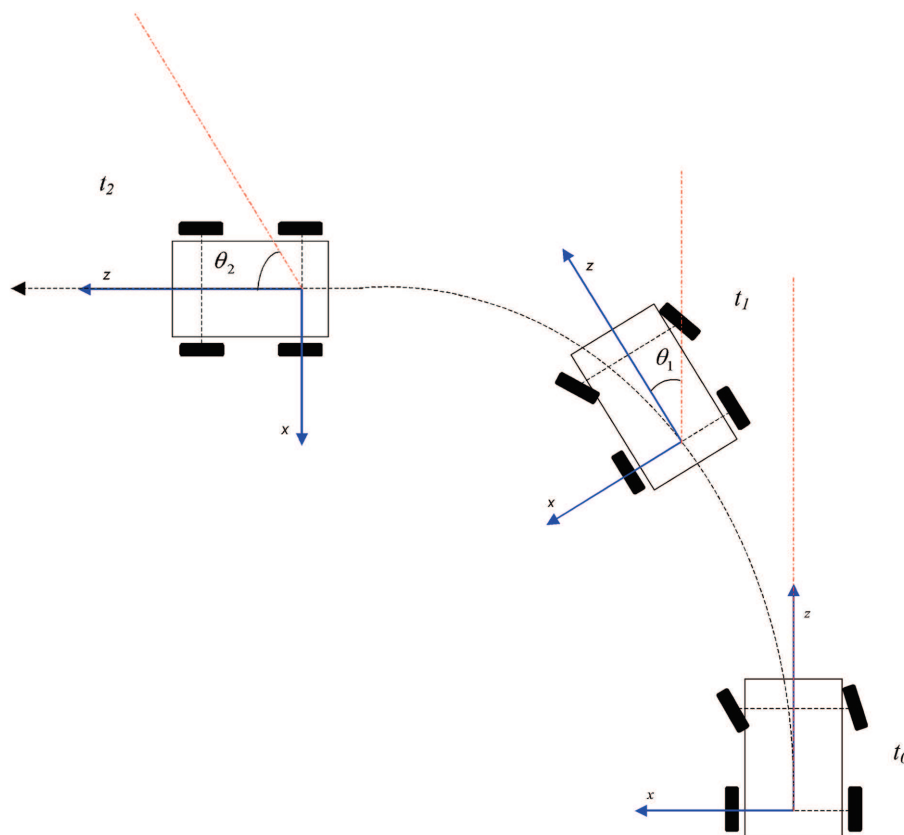


Figura 5.3: Representación del primer umbral de giro θ_1 , seguido del segundo θ_2

El ángulo de orientación se puede obtener fácilmente a partir del correspondiente *quaternion*. Tanto los límites absolutos como el periodo de tiempo, se han establecido experimentalmente tras la realización de pruebas en diferentes entornos urbanos, evitando la captura de huellas SIFT debidas a ligeros cambios aleatorios en el movimiento del vehículo a lo largo del recorrido, capturándolas únicamente en los denominados *puntos singulares* del mapa. Se entenderá como puntos singulares aquellos en los que se produzcan cambios en la trayectoria del vehículo, como por ejemplo: intersecciones de calles, rotondas, curvas pronunciadas, esquinas, etc.

Cada vez que se crea una nueva huella SIFT se realiza un *matching* con las huellas SIFT previamente adquiridas dentro de una región de incertidumbre. Esta región se obtendrá a partir de la covarianza global del vehículo P_{rob}^G , dado que ésta contiene la información de incertidumbre global del mismo. Si el proceso de *matching* resulta positivo esto quiere decir que el vehículo se encuentra en un lugar previsitado, identificándose por tanto una situación de *cierre de lazo*. En ese momento se aplica el algoritmo MLR para determinar la estimación de máxima verosimilitud de las poses asociadas a todas las huellas. Finalmente las correcciones en las huellas son transmitidas a sus correspondientes sub-mapas asociados.

En la Fig. 5.4 se muestra un diagrama que representa las etapas del proceso del SLAM de alto nivel.

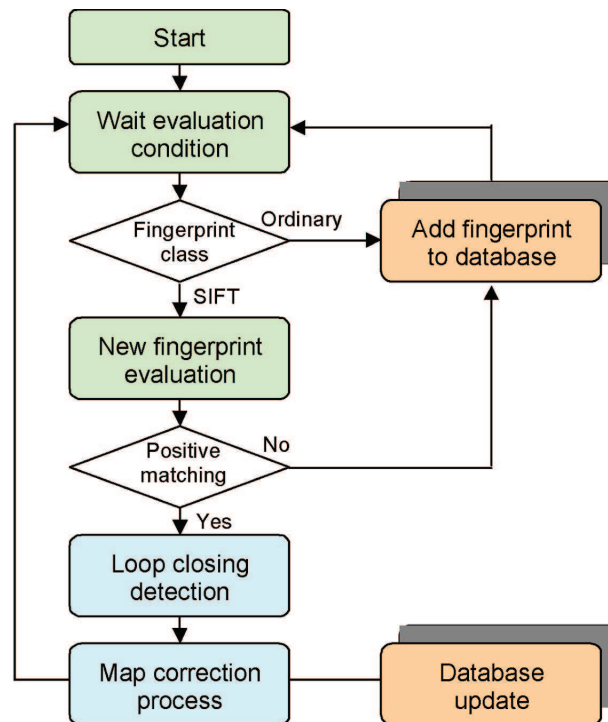


Figura 5.4: Gestión de alto nivel del mapa

5.3. Sub-mapas locales

Cada vez que se crea una nueva huella, tanto ordinaria como SIFT, se crea un nuevo submapa local. La pose local relativa del vehículo X_{rob}^{fpi} y su covarianza asociada P_{rob}^{fpi} se almacenan en la huella en ese momento. Debido a la necesidad de conocer en todo momento la incertidumbre global actual, es necesario mantener P_{rob}^G actualizada.

Para calcular P_{rob}^G se aplicará la fórmula de *coupling summation*, “sumas de acoplamientos” presentada en [32] de forma recursiva. El proceso es el siguiente:

1. Para obtener P_{rob}^G se deberá resolver la ecuación mostrada en 5.1, tal como se muestra gráficamente en la Fig. 5.5. Donde $X_{rob}^{fp_l}$ representa la pose local relativa a la huella actual, mientras que X_{rob}^0 y $X_{fp_l}^0$ representan las poses absolutas del vehículo y de la huella actual, respectivamente.

$$P_{rob}^G = \frac{\partial X_{rob}^0}{\partial X_{fp_l}^0} \cdot P_{fp_l}^0 \cdot \left(\frac{\partial X_{rob}^0}{\partial X_{fp_l}^0} \right)^T + \frac{\partial X_{rob}^0}{\partial X_{rob}^{fp_l}} \cdot P_{rob}^{fp_l} \cdot \left(\frac{\partial X_{rob}^0}{\partial X_{rob}^{fp_l}} \right)^T \quad (5.1)$$

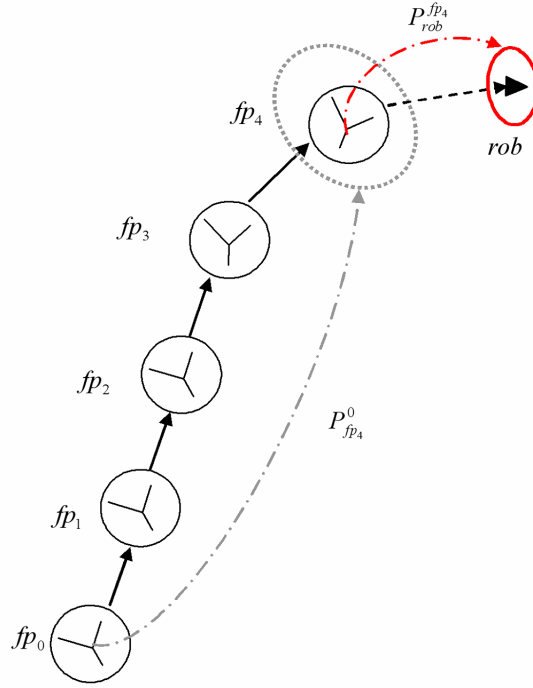


Figura 5.5: Primer paso en la obtención de P_{rob}^G , en función de la covarianza local relativa del vehículo $P_{rob}^{fp_l}$ (en rojo) y de la covarianza global absoluta de la huella actual $P_{fp_l}^0$ (en gris).

2. En segundo lugar, para obtener la covarianza global de la huella actual $P_{fp_l}^0$, se aplicará la ecuación 5.1 de nuevo, pero esta vez aplicada a la huella anterior, tal como se muestra en 5.2. Gráficamente se muestra en la Fig. 5.6

$$P_{fp_l}^0 = \frac{\partial X_{fp_l}^0}{\partial X_{fp_{l-1}}^0} \cdot P_{fp_{l-1}}^0 \cdot \left(\frac{\partial X_{fp_l}^0}{\partial X_{fp_{l-1}}^0} \right)^T + \frac{\partial X_{fp_l}^0}{\partial X_{fp_l}^{fp_{l-1}}} \cdot P_{fp_{l-1}}^{fp_l} \cdot \left(\frac{\partial X_{fp_l}^0}{\partial X_{fp_l}^{fp_{l-1}}} \right)^T \quad (5.2)$$

3. Se aplicará el método anterior de forma iterativa hasta llegar a la primera huella, donde $r \in \{0 \dots l-1\}$:

$$P_{fp_{l-r}}^0 = \frac{\partial X_{fp_{l-r}}^0}{\partial X_{fp_{l-1-r}}^0} \cdot P_{fp_{l-1-r}}^0 \cdot \left(\frac{\partial X_{fp_{l-r}}^0}{\partial X_{fp_{l-1-r}}^0} \right)^T + \frac{\partial X_{fp_{l-r}}^0}{\partial X_{fp_{l-r}}^{fp_{l-1-r}}} \cdot P_{fp_{l-1-r}}^{fp_{l-r}} \cdot \left(\frac{\partial X_{fp_{l-r}}^0}{\partial X_{fp_{l-r}}^{fp_{l-1-r}}} \right)^T \quad (5.3)$$

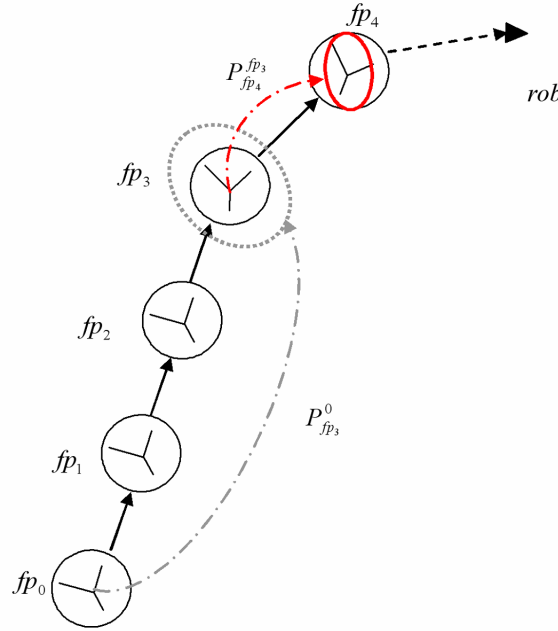


Figura 5.6: Segundo paso en la obtención de P_{rob}^G . $P_{fp_l}^0$ se obtiene en función de la covarianza local relativa de la huella actual respecto a la anterior $P_{fp_l}^{fp_{l-1}}$ (en rojo) y de la covarianza global absoluta de la huella anterior $P_{fp_{l-1}}^0$ (en gris)

4. Por último, para el cálculo de la covarianza global de la primera huella se puede resolver directamente como: $P_{fp_l}^0 = P_{fp_l}^{fp_0}$

En el momento de la creación del sub-mapa, la incertidumbre local del vehículo, condicionada al nuevo sub-mapa y en su propio sistema de referencia $P_{rob}^{fp_{l+1}}$ se establece a 0 inicialmente. Por lo tanto se asume una posición del vehículo localmente cierta respecto al nuevo sub-mapa creado. Por otro lado, las marcas visibles actuales ya habrán sido observadas en el sub-mapa anterior fp_l . Sin embargo, dado que el interés fundamental del sistema se centra en la estimación del recorrido del vehículo, una forma sencilla e intuitiva de tratar la independencia entre sub-mapas es eliminar las marcas del sub-mapa anterior e incorporadas al nuevo sub-mapa fp_{l+1} . De esta forma, el nuevo sub-mapa será iniciado con un determinado número de marcas ya inicializadas, las cuales tendrán nuevas coordenadas $Y_i^{fp_{l+1}}$ expresadas en el nuevo sub-mapa. El proceso se muestra gráficamente en la Fig. 5.7. Por lo tanto, el vector de estado total del sub-mapa comienza tal como se expresa en la ecuación 5.4.

$$X^{fp_{l+1}} = \left(X_{rob}^{fp_{l+1}} \quad Y_1^{fp_{l+1}} \quad Y_2^{fp_{l+1}} \quad \dots \right)^T \quad (5.4)$$

Para calcular $Y_i^{fp_{l+1}}$ a partir de su expresión en el mapa anterior $Y_i^{fp_l}$, se aplicará la fórmula de *common root coupling*, “acoplamiento con referencia común” propuesta en [32]. Ésta permite el cambio de sistema de referencia de fp_l a fp_{l+1} usando el sistema de referencia común del propio vehículo en el nuevo sub-mapa $X_{rob}^{fp_{l+1}}$. En este caso se define $X_{rob}^{fp_{l+1}} = 0$ dado que éste es el sistema de referencia de fp_{l+1} en ese momento.

Para obtener las covarianzas de las marcas expresadas en el sistema de referencia de fp_{l+1} se vuelve a hacer uso de la fórmula de *common root coupling*, en la forma que se muestra en la

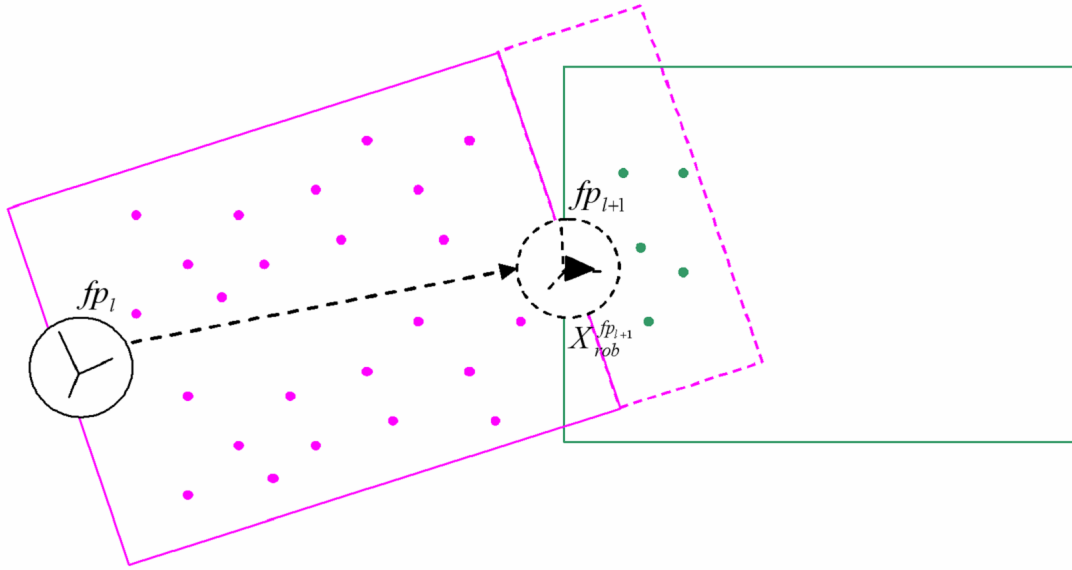


Figura 5.7: Proceso de creación de un nuevo sub-mapa. Las marcas pertenecientes al sub-mapa anterior visibles en el nuevo (magenta) se eliminan del primero y se incorporan como marcas iniciales en el segundo (verde).

ecuación 5.5.

$$P_{Y_i Y_i}^{fp_{l+1}} = \frac{\partial Y_i^{fp_{l+1}}}{\partial Y_i^{rob}} \cdot P_{Y_i Y_i}^{rob} \cdot \left(\frac{\partial Y_i^{fp_{l+1}}}{\partial Y_i^{rob}} \right)^T + \frac{\partial Y_i^{fp_{l+1}}}{\partial X_{fp_{l+1}}^{rob}} \cdot P_{fp_{l+1}}^{rob} \cdot \left(\frac{\partial Y_i^{fp_{l+1}}}{\partial X_{fp_{l+1}}^{rob}} \right)^T \quad (5.5)$$

Si se asume que en el primer instante $X_{rob}^{fp_{l+1}} = 0$, el segundo término de la ecuación 5.5 desaparece y $P_{Y_i Y_i}^{fp_{l+1}}$ depende sólo de $P_{Y_i Y_i}^{rob}$, que representa la incertidumbre en las posiciones de las marcas en el sistema de referencia del vehículo.

Asimismo, si se asume que las nuevas marcas están incorreladas entre ellas, se puede componer la covarianza total del sub-mapa en el momento de la creación del mismo como la que se muestra en la ecuación 5.6.

$$P^{fp_{l+1}} = \begin{pmatrix} P_{rob}^{fp_{l+1}} & 0 & 0 & \dots \\ 0 & P_{Y_1 Y_1}^{fp_{l+1}} & 0 & \dots \\ 0 & 0 & P_{Y_2 Y_2}^{fp_{l+1}} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad (5.6)$$

En el método propuesto en [36] las marcas se comparten entre sub-mapas, en caso de que su número exceda un determinado umbral. En ese momento se crea un enlace entre los dos sub-mapas, expresado como una transformación de *similitud*. Esta transformación vendrá definida como una rotación R y una translación T , añadiéndole además una relación de escala s . De esta forma, al medir las marcas compartidas se permite, no sólo la optimización del sub-mapa local sino también el global, incluso sin cerrar grandes lazos.

Debido al gran tamaño de los entornos exteriores, que son el objeto de la presente Tesis, los lugares comunes pertenecientes a diferentes sub-mapas no suelen ser visibles al mismo tiempo

y, por lo tanto, no se añadirían enlaces entre múltiples nodos habitualmente. Además, la asociación de datos en las marcas de bajo nivel para este tipo de entornos grandes se complica considerablemente.

Por este motivo, el método propuesto en la presente Tesis soluciona el problema definiendo las denominadas huellas SIFT, que identifican lugares singulares, permitiendo reidentificarlas cerrando un lazo y optimizando el mapa global. Esto se explicará en los próximos apartados.

5.4. Huellas SIFT

Como se explicó anteriormente, el sistema identifica lugares específicos usando las denominadas Huellas SIFT. Estas huellas, además de la pose del vehículo, están compuestas por una serie de marcas SIFT distribuidas por toda la imagen de referencia y caracterizan la apariencia visual de la imagen, obtenida por la cámara izquierda en dicha pose. Se ha evaluado la posibilidad de usar ambas imágenes de la cámara estéreo para caracterizar la huella SIFT. Sin embargo, tras comprobar los resultados del método desarrollado, se comprueba que éstas son altamente distinguibles empleando una única imagen.

Tal como se explicó en el apartado 4.8.3, las marcas SIFT son invariantes al escalado de imagen y rotación, así como parcialmente invariantes a cambios en iluminación y punto de vista 3D de la cámara.

Por otra parte, estas marcas son altamente distinguibles, lo que permite que cada marca sea correctamente emparejada con una alta probabilidad. Esto es posible gracias al descriptor $\vec{\delta}$ de longitud 128 asociado a cada una de las marcas, el cual identificará de forma única a todas ellas. El proceso consiste fundamentalmente en:

1. Seleccionar la imagen de referencia, en este caso la izquierda, y buscar los puntos candidatos para la extracción de características.
2. Extraer los descriptores $\vec{\delta}$ de cada una de las marcas, así como las coordenadas de proyección de la marca y la posición 3D del vehículo. Éstos datos se almacenan en cada huella SIFT de la forma indicada en la expresión 5.7

$$Y f_m^q = \left(u_L \quad v_L \quad x_{rob} \quad y_{rob} \quad z_{rob} \quad \vec{\delta} \right) \quad (5.7)$$

Dado que, tal como se explica en el apartado 5.2, las huellas SIFT sólo se generan cuando el vehículo ha realizado un cambio significativo en su trayectoria, éstas quedarán ubicadas en las *esquinas* del recorrido realizado, como se muestra a modo de ejemplo en la Fig. 5.8. Estos serán, por tanto, los puntos singulares a identificar por el proceso de detección de cierre de lazos.

5.5. Detección de cierre de lazo

Uno de los principales problemas del SLAM en grandes entornos es el problema del *cierre de lazo*. Esta situación se da cuando el vehículo en cuestión vuelve a pasar por un lugar en el que ya se ha encontrado previamente, tras haber recorrido otras partes del entorno. Si el sistema es capaz de reconocer esta situación, se dispondrá de una información adicional y más fiable de la pose del vehículo, pudiendo corregirla e incluso optimizar la estimación del mapa. Por lo tanto, el primer problema a resolver es la capacidad de reconocimiento de lugares previsitados.

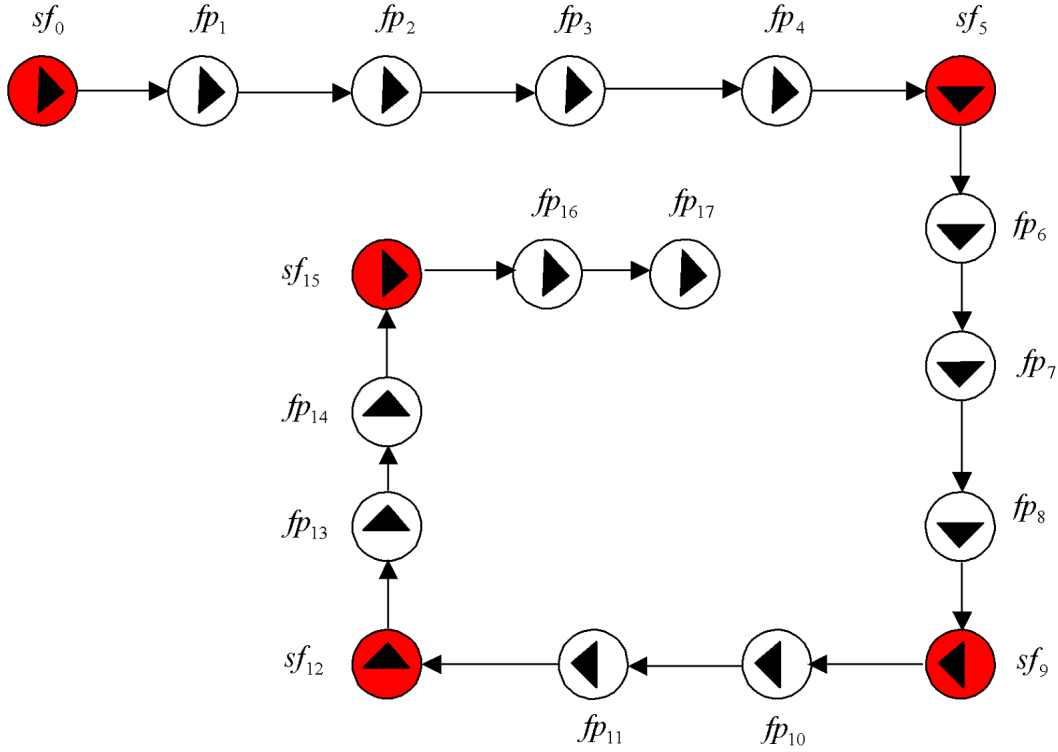


Figura 5.8: Ejemplo de grafo de alto nivel. Las huellas SIFT se marcan en rojo, generándose éstas en las *esquinas* del recorrido.

Durante el proceso continuo de generación del mapa global y avance del vehículo, cada vez que se genera una nueva huella SIFT, ésta se compara con todas las huellas SIFT existentes en el mapa dentro de la región de incertidumbre global definida por P_{rob}^G , tal como se muestra en la Fig. 5.9.

Esta comparación se lleva a cabo para cada par de huellas SIFT (sf_A, sf_B), teniendo en cuenta, tanto el número de marcas SIFT reconocidas como sus posiciones relativas dentro de las imágenes a comparar. El proceso es el siguiente:

1. Cálculo de la distancia Euclídea entre los descriptores $\vec{\delta}_i^A, \vec{\delta}_j^B$ de todas las marcas SIFT detectadas en ambas huellas (sf_A, sf_B), tal como se muestra en la ecuación 5.8.

$$\left\{ \left\| \vec{\delta}_1^A - \vec{\delta}_1^B \right\|, \dots, \left\| \vec{\delta}_1^A - \vec{\delta}_{mB}^B \right\|, \left\| \vec{\delta}_2^A - \vec{\delta}_1^B \right\|, \dots, \left\| \vec{\delta}_{mA}^A - \vec{\delta}_{mB}^B \right\| \right\} \quad (5.8)$$

De éstas se seleccionan sólo las que se encuentren suficientemente cerca. El valor umbral se obtiene empíricamente tras ensayar diferentes alternativas.

2. Posteriormente se calculan las líneas que unen cada par de marcas correctamente emparejadas. De éstas se calculan sus correspondientes longitudes $Ln_{i,j}^{A-B}$ y pendientes $Sp_{i,j}^{A-B}$, tal como se muestra en la Fig. 5.10(b).
3. Para estimar cuáles de las marcas SIFT han sido correctamente emparejadas tras aplicar las restricciones geométricas del paso anterior, se han estudiado dos alternativas.
 - El primer método consiste en un enfoque simple basado en la varianza estadística del conjunto de pendientes $Sp_{i,j}^{A-B}$ y longitudes $Ln_{i,j}^{A-B}$. La probabilidad total de

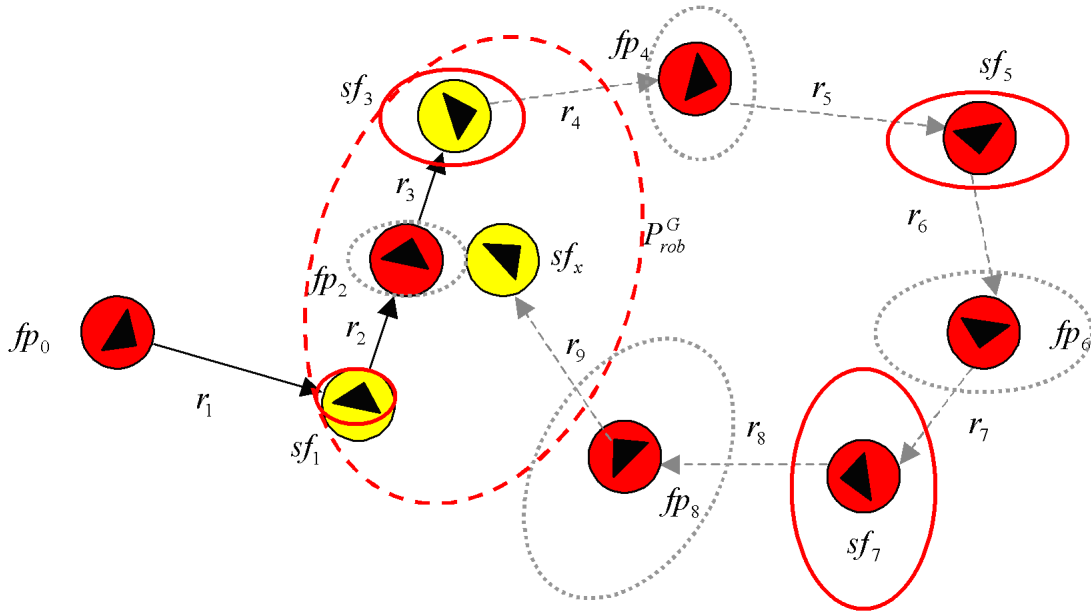


Figura 5.9: Proceso de búsqueda de lugares previsitados. Los nodos circulares representan las diferentes huellas a lo largo del recorrido del vehículo. En el instante mostrado, una nueva huella SIFT sf_x está siendo evaluada. Dentro de la región de incertidumbre global P_{rob}^G se encuentran dos huellas SIFT previamente obtenidas, que se han destacado en amarillo, candidatas a ser reconocidas.

matching entre las dos huellas se calcula como se indica en la ecuación 5.9, donde m'_1 , m'_2 y m'_3 son coeficientes obtenidos empíricamente.

$$P_{fp_match} = m'_1 \sigma_{Sp}^{-1} + m'_2 \sigma_{Ln}^{-1} + m'_3 \cdot num_matches \quad (5.9)$$

De esta forma, a mayor variabilidad en el conjunto de líneas de asociación, menor probabilidad de *matching* entre huellas.

- La segunda alternativa estudia la aplicación del método RANSAC. El propósito de este método consiste en la obtención de un modelo a partir de un conjunto de datos. De estos datos se supone que un grupo de ellos cumplen con el modelo (*inliers*) y el resto no lo cumplen (*outliers*). Si el modelo contiene una proporción suficiente de *inliers* se asume como correcto.

En este caso el modelo a seguir se define como el vector $(avg(Ln_{i,j}^{A-B}), avg(Sp_{i,j}^{A-B}))$, que contiene las medias de las longitudes y pendientes de las rectas de unión. El método RANSAC se aplica al conjunto de todas las líneas, calculando la distancia euclídea entre todos los pares longitud/pendiente y la media. Las marcas cuyas rectas de unión se acercan lo suficiente al modelo se consideran *inliers*, el resto serán consideradas *outliers*.

La probabilidad total de *matching* entre las dos huellas se calcula como una función ponderada de dos parámetros:

- Probabilidad asociada al número de marcas correctamente emparejadas, como se indica en 5.10.

$$P(num_matches) = \frac{num_matches}{m_3} \quad (5.10)$$

- Relación entre *inliers* y número de emparejamientos positivos, como muestra 5.11.

$$P_{fp_match} = m_1 \cdot P(num_matches) + m_2 \left(\frac{n_I}{num_matches} \right) \quad (5.11)$$

Los coeficientes m_1 , m_2 , m_3 fueron obtenidos experimentalmente. En algún caso, $P(num_matches)$ podría resultar mayor que 1, por lo tanto se ha introducido un límite superior para evitar esta situación. Los valores empíricamente obtenidos en los experimentos fueron: $m_1 = 2/3$, $m_2 = 1/3$ y $m_3 = 40$

A la hora de comparar ambos métodos, la primera conclusión que se extrae es la gran mejora del método RANSAC frente al método simple. Esta mejora es debida a la baja influencia de las *minorías* en la probabilidad final. Es decir, en el método RANSAC un resultado con un bajo número de *outliers* tendría asociada una probabilidad de *matching* alta. Por el contrario, el mismo resultado basado en las varianzas de la muestra, arrojaría una probabilidad de *matching* mucho menor, tal como se muestra en la Fig. 5.10. Esto es debido, fundamentalmente, a la alta contribución de los *outliers* en el incremento de la varianza. Por lo tanto, la principal ventaja del método RANSAC es su baja tasa de falsos negativos.

Se ha realizado un experimento en el que se realizaron ensayos sobre 80 pares de imágenes de prueba. El número de falsos negativos usando el método RANSAC fue de 16, mientras que empleando el método basado en varianzas fue de 28. Por lo tanto el método elegido fue el RANSAC.

5.6. Corrección del mapa

Una vez detectada una situación de cierre de lazo, el estado del vehículo deberá ser actualizado de forma que concuerde con el correspondiente al antiguo lugar reconocido. Además, todo el mapa global, incluyendo las huellas y las marcas de bajo nivel, deberá ser modificado de forma que el resultado final sea consistente. En la presente Tesis se han evaluado dos métodos encargados de la corrección del mapa global.

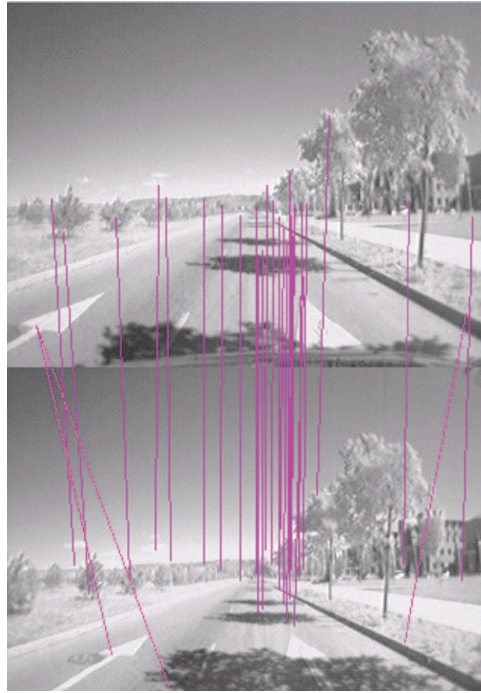
En primer lugar se ha desarrollado un método aproximado basado en la corrección lineal de la pose de las huellas así como de la posición de las marcas. Esta corrección se basa en la traza de las covarianzas de incertidumbres globales de las huellas a lo largo del camino recorrido por el vehículo.

Posteriormente se ha implementado la solución desarrollada en el método MLR (*MultiLevel Relaxation*) [41]. Éste se basa en la optimización por mínimos cuadrados no lineales, utilizando las covarianzas locales relativas de las huellas.

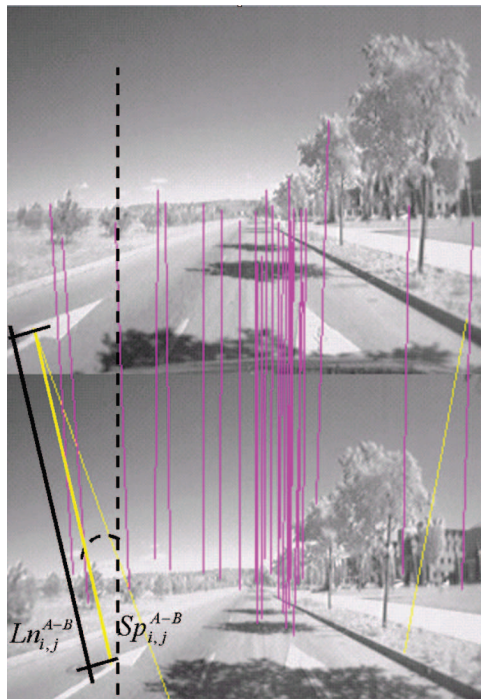
Al final del apartado se realiza una comparativa de resultados entre los dos métodos.

5.6.1. Método lineal

La idea fundamental del método desarrollado se basa en que la incertidumbre global del vehículo P_{rob}^G crecerá de forma continuada a medida que éste recorre su camino, antes de reconocer un lugar previamente visitado. Es decir, el error en la localización se incrementará hasta que el vehículo reconozca un lugar previsitado que le ayude a reducir su propia incertidumbre.



(a) Probabilidad de *matching* usando varianzas $P_{fp_match} = 0,63$



(b) Obtención de la longitud y pendiente de la recta asociada a una de las marcas etiquetadas como *outlier* (amarillo). Los *inliers* se etiquetan en color magenta. Probabilidad de *matching* usando RANSAC $P_{fp_match} = 0,87$

Figura 5.10: Líneas de unión entre marcas SIFT inicialmente emparejadas. Se muestran las probabilidades de *matching* sobre las mismas huellas usando RANSAC comparado con el uso del método basado en varianzas

Respecto a la construcción del mapa, se puede deducir que las marcas más antiguas tendrán una incertidumbre asociada menor y, por tanto deberán ser corregidas en menor medida. Por el contrario, las marcas más recientes deberán ser corregidas en una mayor cuantía. Este grado de corrección vendrá determinado en función de la incertidumbre histórica acumulada P_{rob}^G a lo largo de todo el lazo, tal como se muestra en la Fig. 5.11.

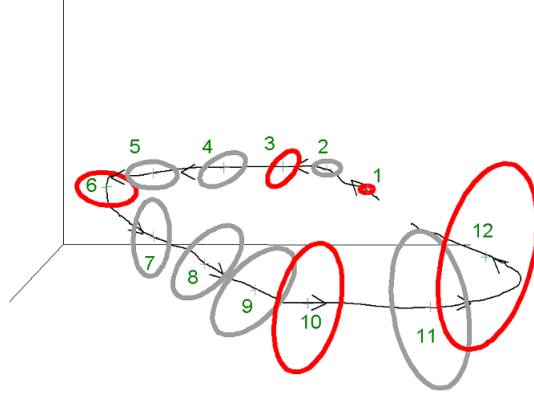


Figura 5.11: Representación del incremento progresivo en la incertidumbre global del vehículo

En las ecuaciones 5.12, 5.13 y 5.14 se muestran los tres pasos consecutivos aplicados a la posición de una sola marca para obtener su nuevo valor estimado tras la corrección. $T(P_{fp}^0(i))$ representa la traza de la covarianza global absoluta de la huella asociada a la marca i . \vec{D}_R es la expresión de tres componentes de la matriz de rotación absoluta del vehículo R_{rob}^0 . El operador Rot representa la transformación del vector de tres componentes en la matriz de rotación asociada. Los atributos $init$ y fin se refieren a la situación antes y después de la corrección del mapa. Al operar este método en coordenadas absolutas, las posiciones de las diferentes marcas Y_i están expresadas también en coordenadas absolutas.

$$Y_i' = (R_{init}^0)^{-1} \cdot (Y_i^{init} - (X_{rob}^0)_{init}) \quad (5.12)$$

$$Y_i'' = Y_i' \cdot Rot \left[\frac{T(P_{fp}^0(i))}{T(P_{fpfin}^0)} \cdot (\vec{D}_{Rfin})^{-1} + \left(1 - \frac{T(P_{fp}^0(i))}{T(P_{fpfin}^0)} \right) \cdot (\vec{D}_{Rinit})^{-1} \right] \quad (5.13)$$

$$Y_i^{fin} = Y_i'' - \left[\frac{T(P_{fp}^0(i))}{T(P_{fpfin}^0)} \cdot ((X_{rob}^0)_{fin})^{-1} + \left(1 - \frac{T(P_{fp}^0(i))}{T(P_{fpfin}^0)} \right) \cdot ((X_{rob}^0)_{init})^{-1} \right] \quad (5.14)$$

El proceso consiste básicamente en dos cambios de sistema de referencia, con la particularidad de que el segundo está ponderado:

1. En primer lugar se modifica la posición de las marcas, de forma que el sistema de referencia del vehículo, antes de la corrección de su pose, se traslada y se rota hasta que coincida con el sistema de referencia global.
2. En una segunda etapa, se vuelven a reubicar las marcas, trasladando y rotando el sistema de referencia anterior hasta volver a hacerlo coincidir con el del vehículo después de la corrección de su pose. Sin embargo, en este caso, la pose a tener en cuenta será una

interpolación lineal entre la pose original y la corregida, ponderada por la traza de la matriz de covarianza del vehículo asociada a cada marca.

En la Fig. 5.12 se muestra gráficamente el proceso.

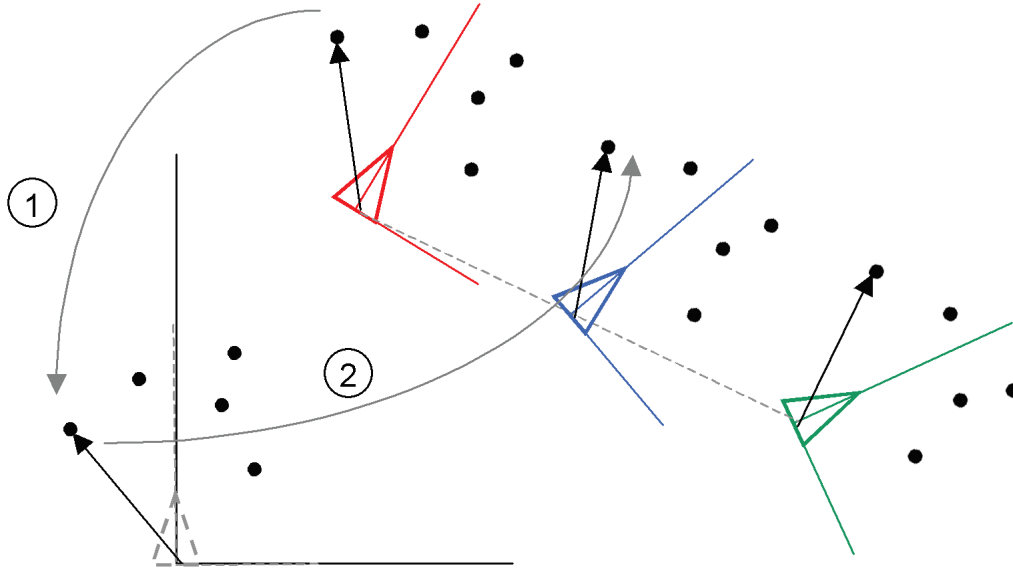


Figura 5.12: Proceso de corrección en la posición de las marcas. En rojo se muestra la pose inicial del vehículo antes de la corrección. En verde se muestra la pose una vez corregida. En azul se muestra la interpolación ponderada por la huella asociada a la marca en cuestión

Una vez corregido el mapa la incertidumbre actual del vehículo P_{rob}^G se actualiza a la incertidumbre correspondiente a la huella revisitada, tal como se muestra en la Fig. 5.13.

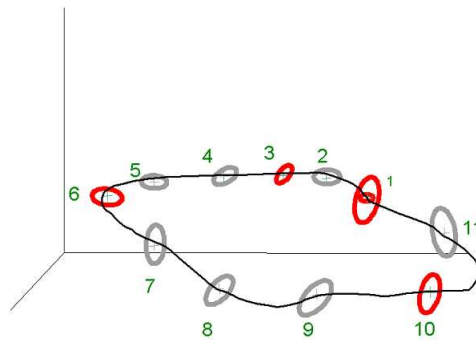


Figura 5.13: Representación del mapa global una vez corregido

5.6.2. Método MLR

Debido fundamentalmente a la complejidad potencial de los posibles lazos en el recorrido del vehículo para aplicaciones reales en entornos exteriores por un lado, y a la inexactitud del método propuesto anteriormente debido a que se trata de un método de corrección lineal aproximado, se ha evaluado la optimización del mapa empleando el método MLR.

El propósito de este algoritmo es asignar un conjunto globalmente consistente de coordenadas cartesianas a las huellas del grafo, basado en medidas locales e inconsistentes, tratando de maximizar la verosimilitud total de todas las medidas. Las razones fundamentales de su utilización son su implementación altamente eficiente, en términos de coste computacional y la extremadamente alta complejidad que permite entre los lugares nuevos y los previsitados. Este algoritmo proporciona la habilidad de cerrar múltiples lazos incluso si éstos están anidados de forma jerárquica.

Las entradas del MLR son las poses y covarianzas relativas de las huellas. Como salidas el MLR proporciona el conjunto más probable de poses de las huellas, es decir, el conjunto de huellas ya corregido.

El algoritmo MLR propuesto por Udo Frese et al. [41] sólo funciona con información 2D, por lo tanto, es necesario obtener la correspondiente pose en 2D de las huellas $X_{2D}^{fp_i}$, así como sus covarianzas $P_{2D}^{fp_i}$ a partir de $X_{fp_i}^{fp_{i-1}}$ y $P_{fp_i}^{fp_{i-1}}$ respectivamente. Aunque este hecho pudiera considerarse una limitación, en la práctica, para una aplicación como la abordada en esta Tesis no lo es tanto, ya que la mayoría de los entornos urbanos son esencialmente planos y los sistemas de navegación actuales manejan información 2D.

En primer lugar, la pose 2D se define como $X_{2D}^{fp_i} = (x_{2D} \ y_{2D} \ \theta_{2D})^T$, es decir, las dos coordenadas bidimensionales y el ángulo de orientación, resultado de la proyección sobre el plano de movimiento, de forma similar al proceso llevado a cabo por los navegadores comerciales. De este modo, podemos relacionar las coordenadas 2D y las 3D como se muestra en la ecuación 5.15.

$$X_{2D}^{fp_i} = \left(x_{fp_i}^{fp_{i-1}} \quad z_{fp_i}^{fp_{i-1}} \quad 2 \arccos \left(q_{0fp_i}^{fp_{i-1}} \right) \right)^T \quad (5.15)$$

Por otra parte, es posible calcular la covarianza 2D empleando los jacobianos correspondientes tal como se muestra en 5.16.

$$P_{2D}^{fp_i} = \frac{\partial X_{2D}^{fp_i}}{\partial X_{fp_i}^{fp_{i-1}}} \cdot P_{fp_i}^{fp_{i-1}} \cdot \left(\frac{\partial X_{2D}^{fp_i}}{\partial X_{fp_i}^{fp_{i-1}}} \right)^T \quad (5.16)$$

Dichos jacobianos se calculan como se indica en la ecuación 5.17

$$\frac{\partial X_{2D}^{fp_i}}{\partial X_{fp_i}^{fp_{i-1}}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2/\sqrt{1-q_{0fp_i}^2} & 0 & 2/\sqrt{1-q_{yfp_i}^2} & 0 \end{pmatrix} \quad (5.17)$$

El algoritmo MLR se basa en un método previo más sencillo denominado SLR (*Single Level Relaxation*, “Relajación en un único nivel”). Con objeto de poder llegar a entender su funcionamiento, a continuación se describen brevemente los pasos del método:

1. En primer lugar calcula una función de error cuadrático de las huellas según 5.18, donde $X_M = (Xc_{fp_1}^0 \ Xc_{fp_2}^0 \ \dots \ Xc_{fp_L}^0)^T$ representa el vector total que contiene el conjunto de todas las poses 2D corregidas. En este caso las poses se expresan en coordenadas globales.

$$\Psi^2(X_M) = (X_M)^T A_M X_M - 2(X_M)^T b_M \quad (5.18)$$

2. Posteriormente, se trata de buscar un valor de X_M que minimice Ψ^2 , para lo cual se resuelve $A_M X_M = b_M$. La manera eficiente de resolver esta ecuación es la clave de la denominada técnica de *relajación*.

La función de error Ψ^2 se define explícitamente como se indica en la ecuación 5.19, donde cada una de las huellas se calcula como se indica en 5.20.

$$\Psi^2(X_M) = \sum_{l=0}^L \left(\eta_{2D}^{fp_l} \right)^T \left(P_{2D}^{fp_l} \right)^{-1} \eta_{2D}^{fp_l} \quad (5.19)$$

$$\eta_{2D}^{fp_l} = f_M \left(X_{fp_l}^0, X_{fp_{l-1}}^0 \right) - X_{2D}^{fp_l} \quad (5.20)$$

La función f_M transforma $X_{fp_l}^0$ para expresarla en el sistema de coordenadas de $X_{fp_{l-1}}^0$. Por lo tanto, $\eta_{2D}^{fp_l}$ representa la diferencia entre la pose corregida y la estimada.

Linealizando f_M como se muestra en [41], la ecuación 5.19 se puede expresar en la forma de 5.18. La idea fundamental del proceso de relajación es explotar la dispersión de la ecuación $A_M X_M = b_M$ de forma que se pueda resolver cada fila de matrices correspondiente a cada una de las poses de las huellas $X_{fp_l}^0$ del vector total X_M de una vez, considerando las demás como constantes. Si se repite el procedimiento para el resto de las huellas de una forma iterativa la ecuación se resuelve de una forma eficiente.

La mejora aportada por el MLR se basa en la idea de simplificar el cálculo de $A_M X_M = b_M$ aún más reduciendo (discretizando) a la mitad cada vez el número de poses iterativamente. Se definen diversos niveles jerárquicos, uno por cada paso en la discretización. En el nivel de mayor discretización, la ecuación residual se resuelve directamente usando el método de factorización de *Cholesky* [72]. Finalmente la solución se interpola repetidamente a través de cada uno de los niveles de discretización hasta llegar al nivel original, de forma que se obtenga el resultado de la ecuación original. Este ciclo de discretización *hacia arriba* e interpolación *hacia abajo* se denomina *V-cycle*. En la Fig. 5.14 se muestra un gráfico ilustrativo del ciclo en V.

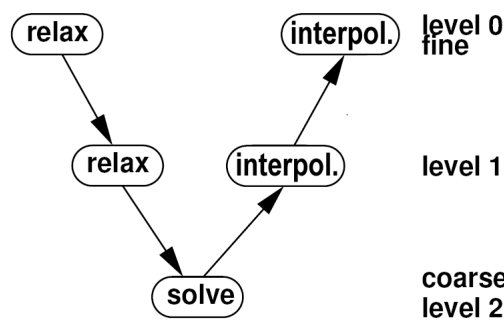


Figura 5.14: Gráfico del proceso de resolución de la ecuación por relajación. En sentido descendente se reflejan las iteraciones de discretización y en sentido ascendente se muestran las interpolaciones.

Una vez que el vector 2D corregido ha sido calculado, se obtienen las correspondientes huellas 3D corregidas. En el paso para obtener las poses 2D a partir de las 3D, como se indica en 5.15, se pierde la información de la coordenada $y_{fp_l}^{fp_{l-1}}$, es decir, la altura. Por lo tanto al volver a obtener

las coordenadas 3D a partir de las 2D es necesario establecer dicho valor. Para ello se asume un terreno esencialmente plano, dado que el sistema presentado está montado en un vehículo comercial destinado a áreas urbanas. Por lo tanto este valor se tomará como 0.

De esta forma, el vector de pose corregida para cada una de las huellas se forma como se indica en 5.21.

$$X_{fp_i}^0 = \left(x c_{fp_i}^0 \quad 0 \quad y c_{fp_i}^0 \quad \cos\left(\frac{\theta c_{fp_i}^0}{2}\right) \quad 0 \quad \sin\left(\frac{\theta c_{fp_i}^0}{2}\right) \quad 0 \right)^T \quad (5.21)$$

En la Fig. 5.15 se muestra un ejemplo de corrección del mapa de alto nivel usando el método MLR.

Como se explicó anteriormente, el método MLR no es capaz de proporcionar las covarianzas globales corregidas de las huellas. La razón es que el método se basa únicamente en las covarianzas relativas entre poses. Como se ha demostrado, el método presentado no necesita conocer las covarianzas globales para realizar la optimización del mapa. Sin embargo, para poder tener una estimación aproximada de las huellas SIFT revisitadas, es necesario mantener actualizada la incertidumbre global del vehículo.

Después de cerrar un lazo, se da la situación de que una huella está relacionada con más de una huella adicional, tal como le ocurre, por ejemplo a sf_3 en la Fig. 5.16. Para calcular la incertidumbre global del vehículo P_{rob}^G , se aplicará la fórmula recursiva de acoplamiento mostrada en 5.1. Para alcanzar la posición rob se pueden acoplar las incertidumbres locales de las huellas, empezando por fp_0 y siguiendo por el camino de los nodos en rojo, o bien recorriendo también el camino de los nodos blancos. Dado que el camino rojo es más corto, escoger esta opción conllevará una menor P_{rob}^G que escogiendo la segunda opción.

Con la acción de cerrar el lazo indicado, se ha reinicializado implícitamente la incertidumbre global en ese momento a la que tenía asociada la huella sf_3 , consiguiendo reducirla. Así pues, como regla, para calcular la P_{rob}^G actual se aplicará la fórmula recursiva al camino más corto posible desde la primera huella hasta la posición actual.

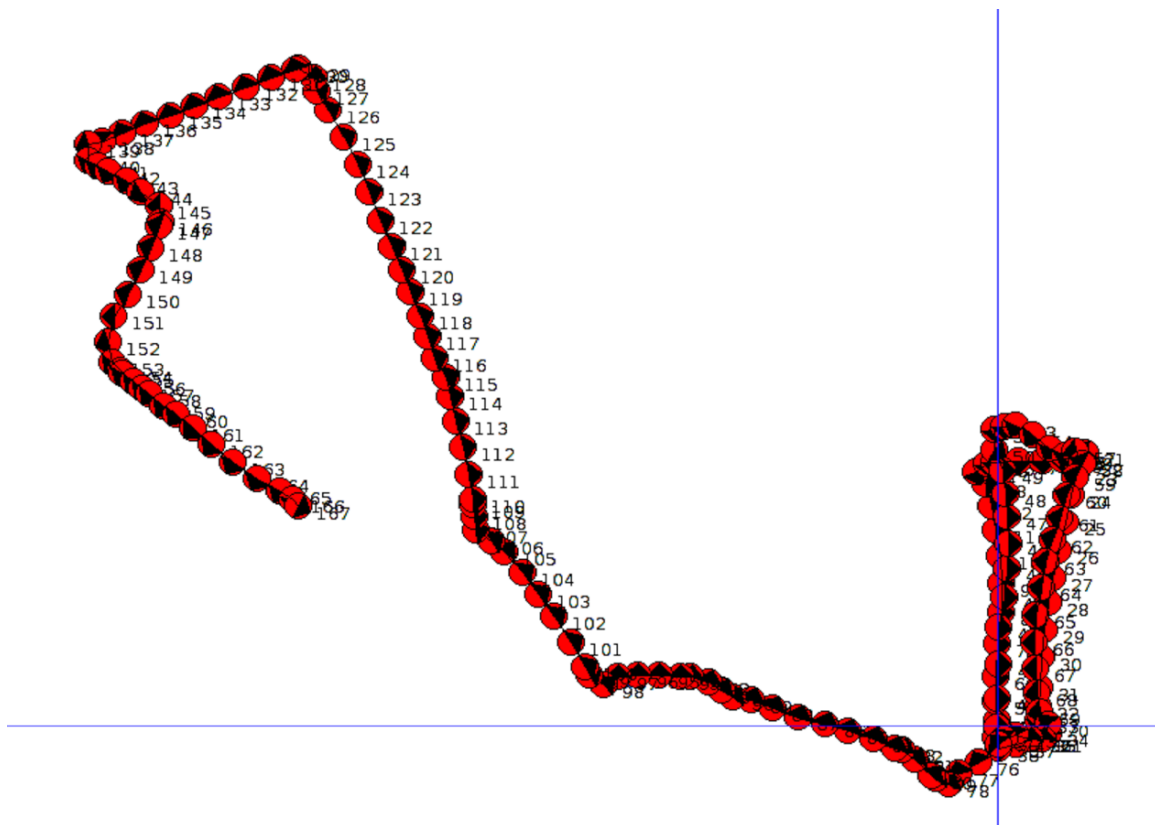
Conocer la incertidumbre global actual es importante de cara a incrementar la eficiencia del proceso de búsqueda de huellas, dado que el número de huellas SIFT a comparar será menor.

El último paso es transferir la corrección realizada en el *high level SLAM* hacia el *low level SLAM*. Esto se realiza aplicando la misma transformación de la pose de cada huella a todas las marcas dentro del sub-mapa. Con esta transformación se mantienen inalteradas las posiciones relativas de las marcas respecto a los sistemas de referencia de sus correspondientes sub-mapas locales. Por lo tanto las covarianzas locales de las marcas permanecen inalteradas en el sistema de referencia de cada sub-mapa aunque sí que se modifican a nivel global. A modo de ejemplo, en la Fig. 5.17 se muestran las incertidumbres globales de las marcas capturadas en una parte de los recorridos utilizados en los ensayos. En ella se representan las covarianzas globales de las marcas habiendo empleado simplemente el EKF (low level SLAM) en los mapas locales (Fig. 5.17(a)) y después de aplicar la optimización MLR (Fig. 5.17(b)).

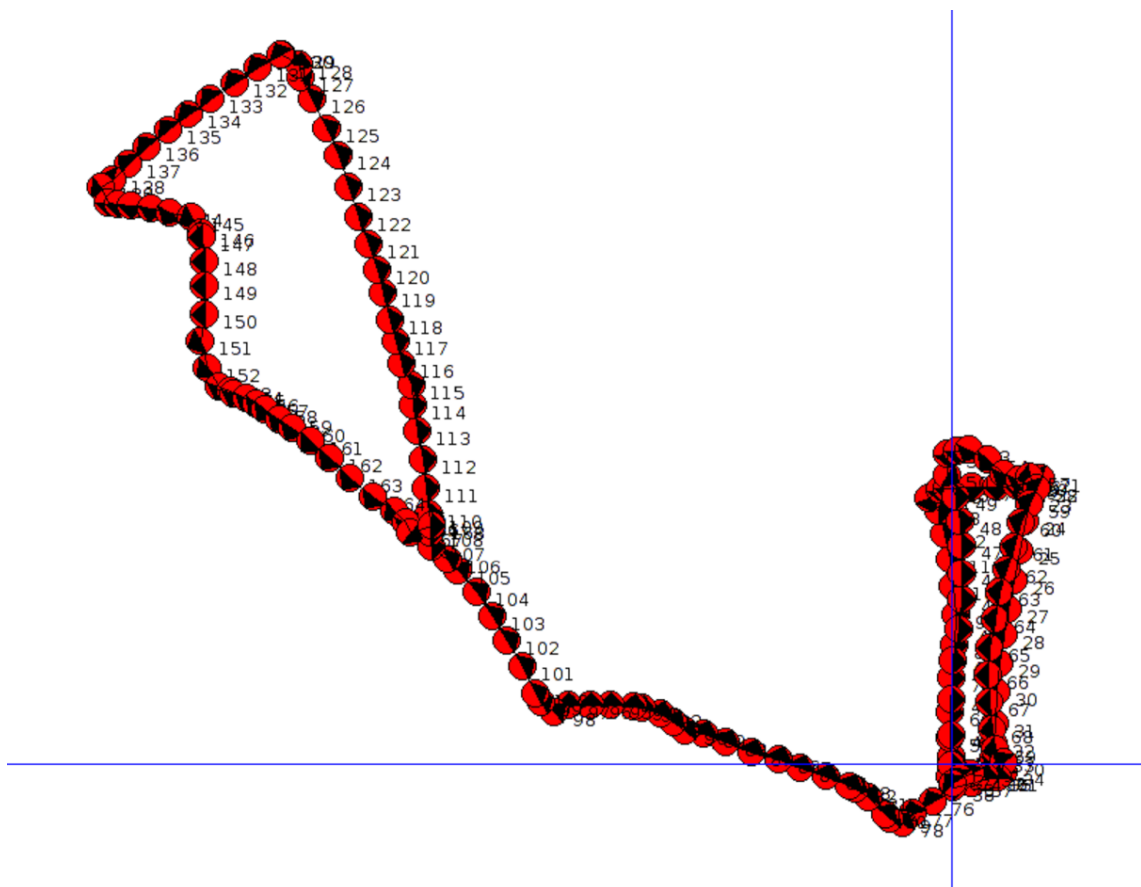
5.6.3. Comparativa

Una vez que ambos métodos han sido descritos, a la hora de evaluar las ventajas e inconvenientes de cada uno, se tendrán en cuenta las siguientes consideraciones:

- En primer lugar, como se explicó anteriormente, el método de optimización lineal se basa en



(a) Estado del mapa antes de realizar la optimización usando el MLR



(b) Estado del mapa después de realizar la optimización usando el MLR

Figura 5.15: Representación del mapa topológico de alto nivel. Las huellas están representadas con círculos rojos en los que el triángulo indica la orientación del vehículo.

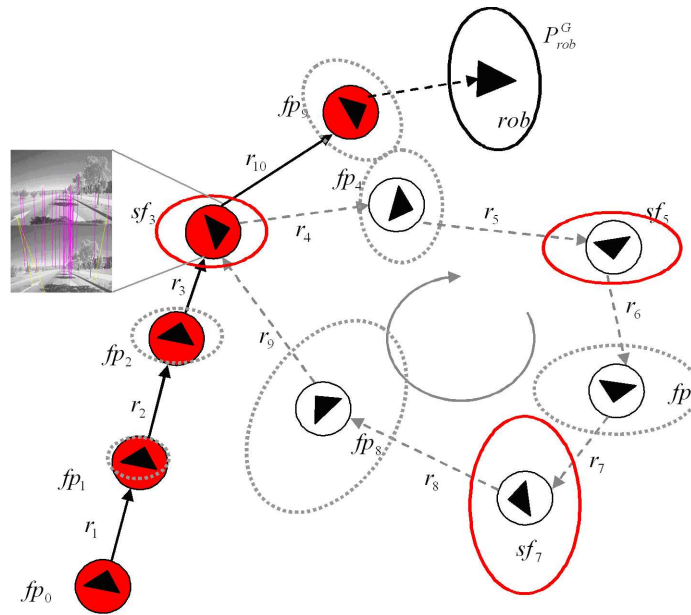
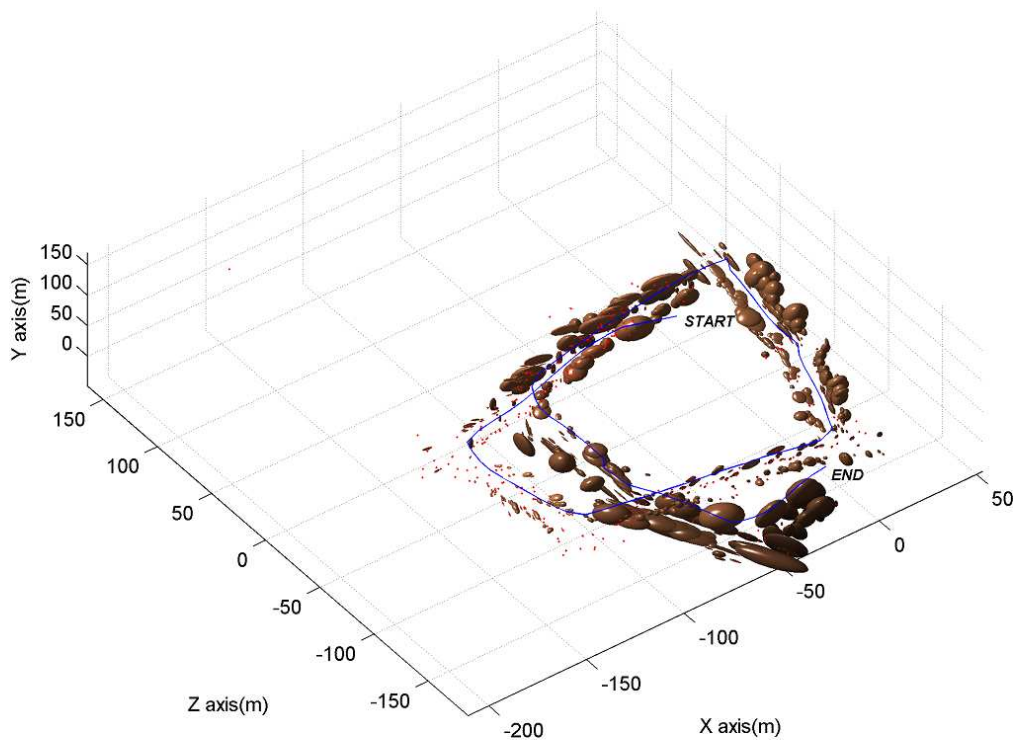


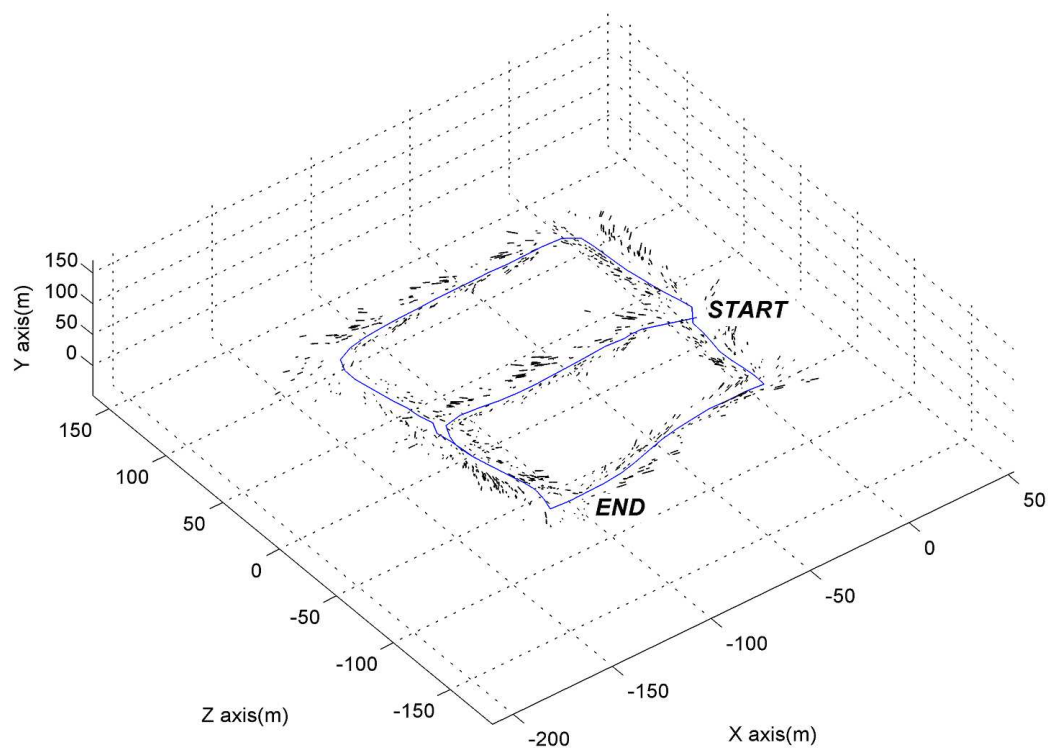
Figura 5.16: Representación de las incertidumbres globales del vehículo P_{rob}^G , incrementándose a lo largo del camino del vehículo en cada una de las poses de las huellas. Las incertidumbres globales marcadas en color rojo están asociadas a las huellas SIFT. Los números denominan a cada huella. El grafo, además muestra un ejemplo de selección del camino más corto para el cálculo de la incertidumbre global después de una situación de cierre de lazo.

una aproximación. Esta consiste fundamentalmente en una ponderación usando la traza de la matriz de covarianza global. Es decir, no se ponderan individualmente cada componente del vector de estado. Por otra parte, se linealiza la corrección en su conjunto, como se puede observar en 5.12, 5.13 y 5.14.

- Por otro lado, a pesar de las limitaciones expuestas anteriormente, el sistema lineal permite optimizar las tres componentes de la trayectoria del vehículo, así como del mapa. Es decir, realiza la corrección en 3D, preservando la información de la *altura*.
- En cuanto a los tiempos de cómputo, debido a la sencillez del método lineal, éste proporciona los resultados en un tiempo reducido. Sin embargo, con la implementación de la mejora del MLR sobre la base del SLR, finalmente los tiempos de procesamiento de ambos métodos son totalmente comparables.
- El MLR por su parte, contempla el problema en su conjunto, basándose en mínimos cuadrados no lineales. La matriz de covarianza del vehículo se tiene en cuenta en su totalidad en la implementación de la ecuación a minimizar. Además, la complejidad de relaciones que el método es capaz de soportar es alta, tal como se muestra en [41].
- Como contrapartida, el método MLR, en su implementación actual, sólo es capaz de procesar información 2D. Dada la aplicación fundamental del sistema, orientada a la navegación de vehículos en entornos urbanos, la información de altura del vehículo no es necesaria. Sin embargo, debido a la adaptación de los vectores 3D a 2D, se realiza una proyección sobre el plano de referencia en la que incluso el mapa 2D obtenido ve reducida su precisión.



(a) Resultado de la estimación únicamente del *low level SLAM*



(b) Resultado tras aplicar la optimización del *high level SLAM*

Figura 5.17: Detalle de una sección de una ruta de prueba en la que se aprecian las incertidumbres globales de las marcas

Teniendo en cuenta las consideraciones anteriores, el primer parámetro a estudiar será la precisión en la reconstrucción del mapa por parte de ambos métodos. Para ello se realizaron pruebas sobre un recorrido urbano de 1773.27 m, en el que sólo se cierra un lazo. Como se puede apreciar en la Fig. 5.18, si se compara la estimación de ambos métodos respecto a la referencia, se observa un área en el que el error es mayor en ambos métodos. Este aumento del error es consecuencia de una estimación errónea, por parte del sensor visual, en un giro del vehículo al comienzo del mencionado tramo. La deriva acumulada a partir de ese punto provoca el error que se observa. Una vez cerrado el lazo y optimizado el mapa global se aprecia que la corrección obtenida usando el método MLR es sensiblemente mejor que usando el método lineal.

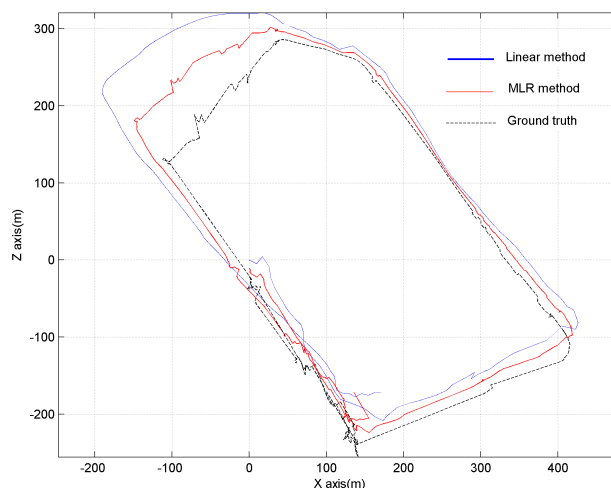


Figura 5.18: Estimaciones del mapa realizadas usando tanto el MLR como el método lineal, así como la estimación de referencia (*ground truth*). Por claridad sólo se muestra la estimación de la trayectoria del vehículo.

Esto es igualmente observable si se representan los errores absolutos respecto a la referencia en ambos ejes, como se muestra en la Fig. 5.19 y la Fig. 5.20.

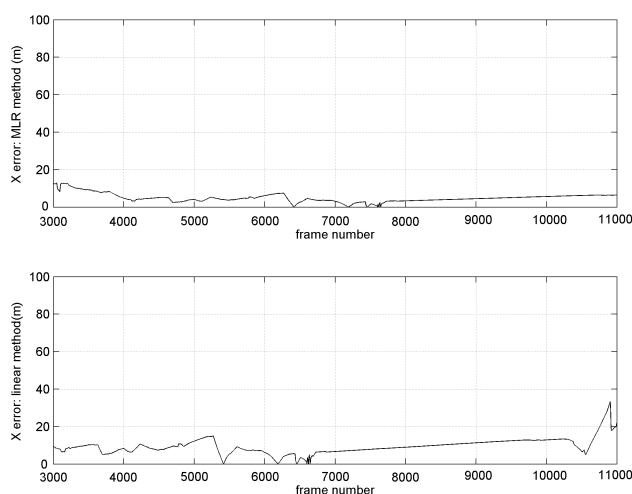


Figura 5.19: Errores de estimación en el eje X usando el MLR y el método lineal.

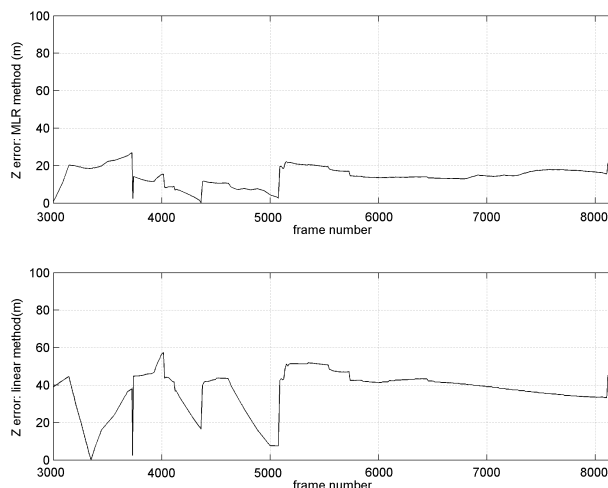


Figura 5.20: Errores de estimación en el eje Z usando el MLR y el método lineal.

Por otra parte, teniendo en cuenta la última de las consideraciones explicadas anteriormente, dado que para aplicar el método MLR es necesario realizar una proyección sobre el plano base, dependiendo de lo plano que sea el terreno a considerar ésta proyección mostrará un error mayor o menor respecto al mapa real.

Para evaluar este efecto en ambos métodos se han realizado pruebas sobre diferentes tramos con diferentes pendientes medias asociadas. En la gráfica de la Fig. 5.21 se muestra una comparativa de los errores medios por cada eje en función de las pendientes medias de los diferentes tramos estudiados. Como referencia se muestra también el error en la estimación de la altura.

Atendiendo a los resultados, se aprecia que el error medio cometido por el MLR en ambos ejes, a pesar de incrementarse ligeramente a medida que aumenta la pendiente media del recorrido, se mantiene siempre por debajo del obtenido con el método lineal. A pesar de no disponer de datos de entornos con mayores pendientes, se presume que por encima de un determinado umbral el error del MLR pueda superar al del método lineal. Ciudades como San Francisco (EEUU), son un ejemplo de entornos urbanos con unos desniveles medios muy elevados en los que el sistema lineal, desarrollado en esta Tesis, obtendría mejores resultados que el MLR. Sin embargo, teniendo en cuenta la topografía de la mayoría de los entornos urbanos, estas situaciones se suponen mucho menos frecuentes.

Como es de esperar, el error en la altura, para el método MLR es directamente el valor absoluto de dicha altura en cada caso. Éste es siempre mayor que en el caso de usar el método lineal para la estimación.

Atendiendo a los resultados, y teniendo en cuenta las aplicaciones objetivo de la Tesis, se demuestra que el método más apropiado es el MLR.

5.7. Fusión con GPS

Con objeto de mejorar la estimación global en recorridos muy largos, se decidió añadir un GPS de bajo coste al ya explicado sistema visual estéreo.

En este apartado se describe la forma en la que se introduce la información proporcionada

por el GPS de bajo coste en el sistema. Se llevará a cabo en los dos niveles jerárquicos:

- En el *low level SLAM*, actualizando las estimaciones del estado y covarianza locales.
- En el *high level SLAM* actualizando las huellas y corrigiendo el mapa global una vez recuperada la señal GPS, tras una situación de pérdida prolongada de la misma.

Debido a la implementación 2D del algoritmo MLR, se pierde la estimación de altura en el recorrido del vehículo. Por lo tanto, no tiene sentido el uso de la información de altitud proporcionada por el GPS. Además, en aplicaciones de asistencia a la navegación, sólo es necesaria la información 2D.

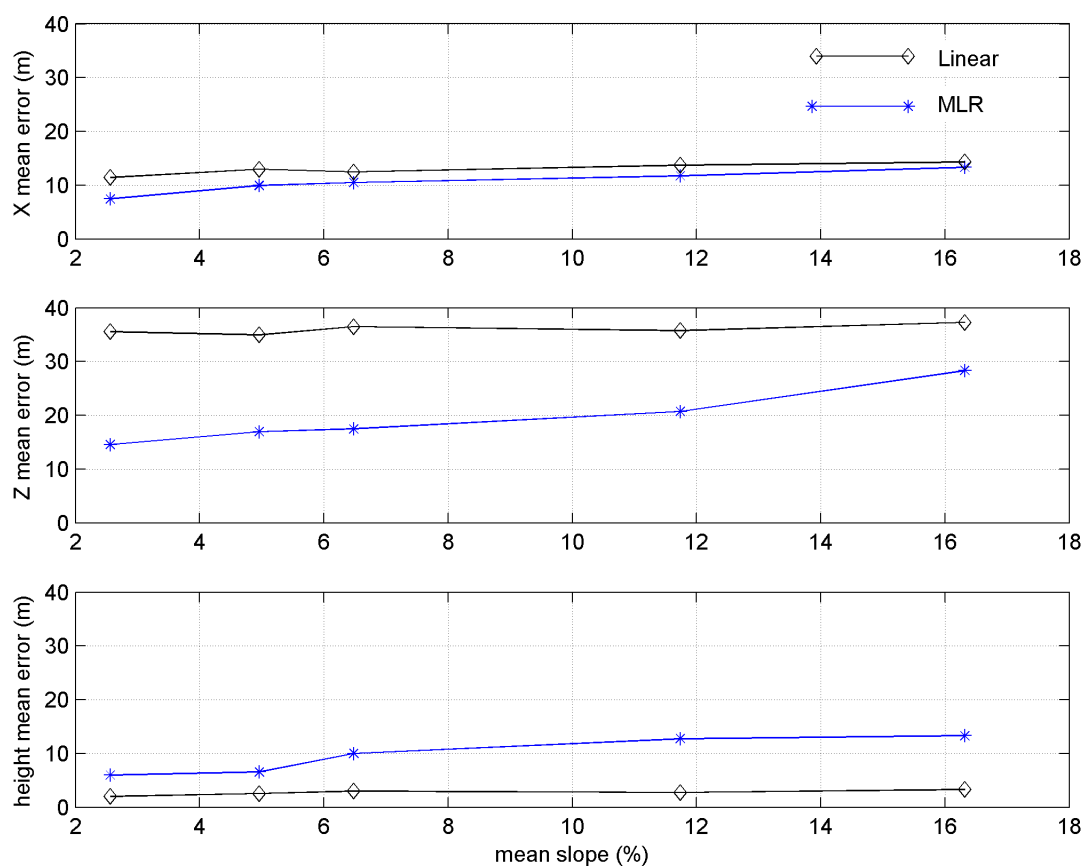


Figura 5.21: Errores medios de estimación en el eje X, en el Z y en la altura en función de las pendientes medias de los diferentes recorridos analizados .

5.7.1. Estimación de la incertidumbre del GPS

Para realizar la fusión de la información proporcionada por ambos sensores es necesario cuantificar el nivel de confianza en las medidas del GPS. Las fuentes de error en los datos del GPS son muy diversas, como por ejemplo: ruido en el receptor, desviaciones en el reloj del satélite, distorsiones de la señal al atravesar la atmósfera, descritas en el *modelo ionosférico*, reflexiones en edificios, etc.

La mayoría de ellas son difíciles o imposibles de cuantificar. Para recoger todos estos errores se define en [73] el valor estadístico denominado UERE (*User Equivalent Range Error*, “error de rango equivalente para el usuario”). En la tabla 5.1 se muestra un estudio estadístico de las contribuciones al error UERE, realizado en [73]. En este mismo estudio se extrae un valor final del UERE al 95 % de 7.22 m.

Estimate of Standard Positioning System User Range Error Single Frequency Receiver (8 June 2000)	
Error Source	User Range Error Contribution (\pm meters)
Navigation Message Curve Fit	0.20
Tropospheric Model	0.25
C/A Code Phase Bias	0.27
Orbit	0.57
Receiver Noise	0.80
Satellite Clock	1.43
Ionospheric Model (global average)	7.00 ¹
URE (95 %)	± 7.22 m

¹ Ionospheric model ranged from 1.30 m (best) to 7.00 (worst)

Tabla 5.1: Contribuciones al UERE.

Por otro lado, la estimación final de la incertidumbre del GPS dependerá también del número de satélites visibles, así como de su distribución espacial. Esto se cuantifica por medio del denominado DOP (*Dilution Of Probability*, “dilución de la probabilidad”), valor que proporciona en tiempo real el propio GPS. Este valor (DOP) será una proporción de la desviación estándar del error de posición σ_R . Por otro lado, σ_R a su vez, se define como el valor de UERE tomado al nivel de 2σ (95 % UERE) (ver Fig. 5.22).

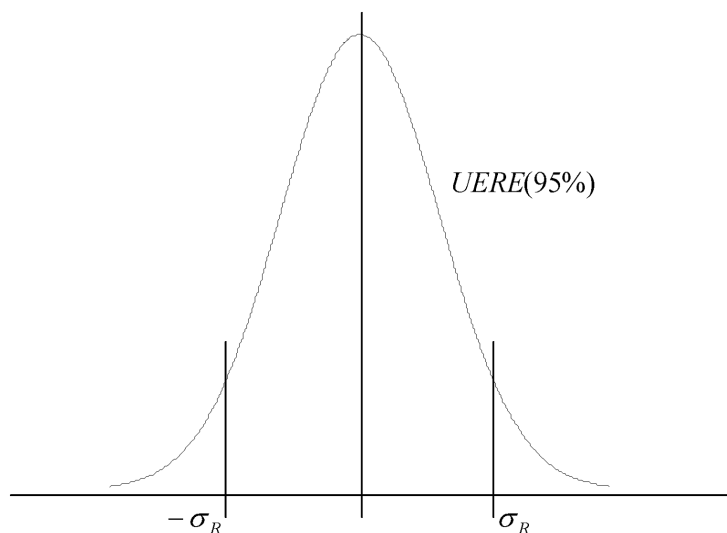


Figura 5.22: Relación entre el error medio σ_R y el valor de UERE.

Teniendo en cuenta que en este caso sólo es relevante el error horizontal, se hará uso del denominado HDOP (Horizontal Dilution Of Probability). Para obtener las desviaciones estándar

en X e Y (plano horizontal) se utiliza la ecuación 5.22 definida en [74].

$$HDOP = \frac{\sqrt{\sigma_x^2 + \sigma_y^2}}{(95\%URE)} \quad (5.22)$$

Teniendo en cuenta que no es posible conocer cada una de las desviaciones típicas individuales σ_x y σ_y , se asumirán ambas iguales. Por lo tanto, la región de incertidumbre final definida por la ecuación 5.23 será plana y circular.

$$P_{GPS} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix} \quad (5.23)$$

5.7.2. Fusión en el SLAM de bajo nivel

Cada vez que está disponible una nueva lectura de GPS (ecuación 5.24), lo cual en circunstancias normales ocurre con un periodo de 1 segundo, se procederá a fusionar dicha lectura con la estimación visual.

$$X_{GPS} = \begin{pmatrix} x_{GPS} & y_{GPS} \end{pmatrix}^T \quad (5.24)$$

La primera observación a realizar, antes de proceder a la fusión, es que el vector de estado del vehículo X_{rob}^{fpi} y del propio sub-mapa X^{fpi} son locales, es decir, están referenciadas a la huella actual fpi . Por otra parte, la información del GPS X_{GPS} es global, es decir, referenciada al sistema global. El resultado final de la fusión quedará expresado como el vector de estado global del vehículo tras la fusión X_{robF}^0 y su covarianza P_{robF}^G .

Por lo tanto, para fusionar las estimaciones de posición del sensor visual y el GPS será necesario, en primer lugar, obtener el vector de estado global del vehículo basado en visión X_{rob}^0 , así como su covarianza P_{rob}^G . Para obtener ésta última, se hará uso de la fórmula de *coupling summation* previamente utilizada en 5.1. En este caso, la expresión 5.25 se basará en la covarianza global total obtenida en la última lectura del GPS $P_{robF(t-1)}^G$, así como la covarianza relativa actual respecto a la de la última medida $P_{rob}^{robF(t-1)}$, tal como se muestra en la Fig. 5.23.

$$P_{rob}^G = \frac{\partial X_{rob}^0}{\partial X_{robF(t-1)}^0} \cdot P_{robF(t-1)}^G \cdot \left(\frac{\partial X_{rob}^0}{\partial X_{robF(t-1)}^0} \right)^T + \frac{\partial X_{rob}^0}{\partial X_{rob}^{robF(t-1)}} \cdot P_{rob}^{robF(t-1)} \cdot \left(\frac{\partial X_{rob}^0}{\partial X_{rob}^{robF(t-1)}} \right)^T \quad (5.25)$$

Para obtener $P_{rob}^{robF(t-1)}$ simplemente se deberá tener en cuenta la forma de actualizar el filtro de bajo nivel en caso de disponer de medidas GPS. Ésta consiste en que, cada vez que se obtenga una medida GPS, se reinicia el filtro respecto al sistema de referencia creado en ese momento $X_{robF(t-1)}^0$. Es decir, los cálculos realizados en el bajo nivel serán referenciados a $X_{robF(t-1)}^0$. Por lo tanto, $X_{rob}^{robF(t-1)}$ y $P_{rob}^{robF(t-1)}$ serán los valores actuales estimados por el propio filtro de bajo nivel (ver Fig. 5.23).

Dado que el GPS no proporciona información de orientación, inicialmente se calcula sólo la posición y, posteriormente, se estima la orientación tal como se explicará más adelante.

Para fusionar ambas posiciones, en primer lugar se expresará la estimación visual obtenida como un vector de dos componentes, según se muestra en la ecuación 5.26.

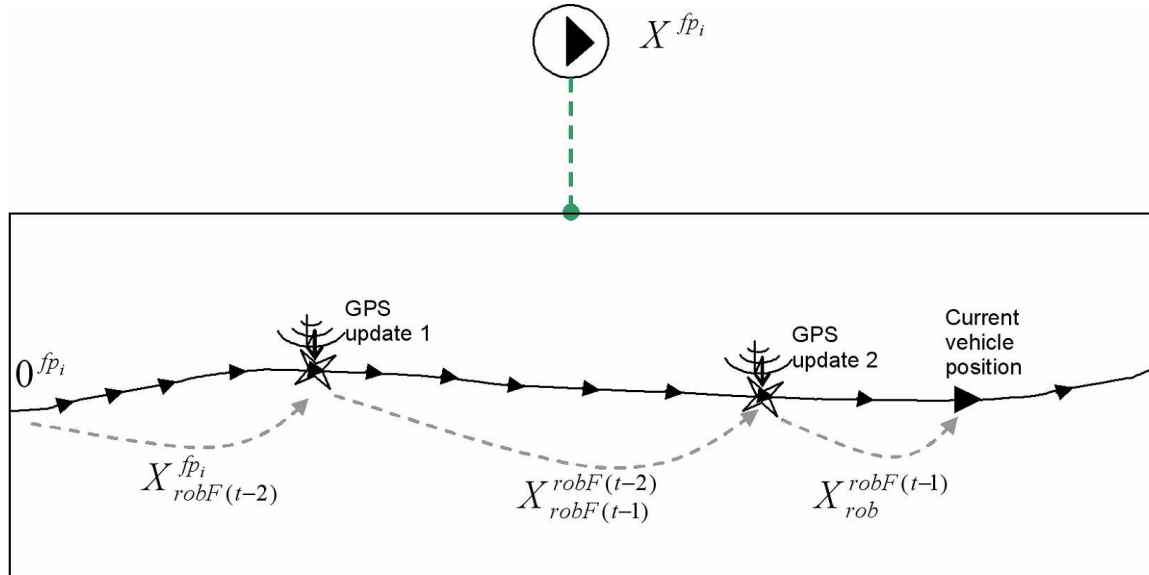


Figura 5.23: Fusión de información visual y GPS. Cada vez que se toma una nueva medida de GPS se resetea el filtro de bajo nivel y las medidas son, a partir de ese momento, referidas a la actualización de GPS anterior.

$$X_{Prob} = \begin{pmatrix} x_{rob}^0 & y_{rob}^0 \end{pmatrix}^T \quad (5.26)$$

Por lo tanto, para calcular la estimación final de la posición se combinan ambas estimaciones haciendo uso de sus respectivas covarianzas de incertidumbre 2D, tal como se muestra en 5.27. Esta fórmula se obtiene aplicando un método estadístico bidimensional basado en la regla de Bayes y el filtro de Kalman. La estimación resultante reduce la distribución de incertidumbre, dado que ésta se calcula como el producto de las dos originales, tal como se muestra en la Fig. 5.24. En [75] se presenta un método eficiente, en términos de tiempos de cómputo, para fusionar dichas distribuciones a partir de sus correspondientes matrices de covarianza.

$$X^{fusion} = X_{Prob} + P_{Prob}^G (P_{Prob}^G + P_{GPS})^{-1} (X_{GPS} - P_{Prob}) \quad (5.27)$$

P_{Prob}^G es la covarianza de la posición 2D calculada en 5.27. P_{Prob}^G , por lo tanto, se define como se muestra en 5.28.

$$P_{Prob}^G = \begin{pmatrix} P_{xx}^G & P_{xz}^G \\ P_{zx}^G & P_{zz}^G \end{pmatrix} \quad (5.28)$$

De forma similar a 5.27, la covarianza final de la estimación de posición, una vez fusionadas ambas fuentes, se calcula por medio de la ecuación 5.29.

$$P^{fusion} = P_{Prob}^G - P_{Prob}^G (P_{Prob}^G + P_{GPS})^{-1} P_{Prob}^G \quad (5.29)$$

Como se explicó anteriormente, el GPS no proporciona información de orientación. Para realizar una estimación se ha optado por la siguiente solución:

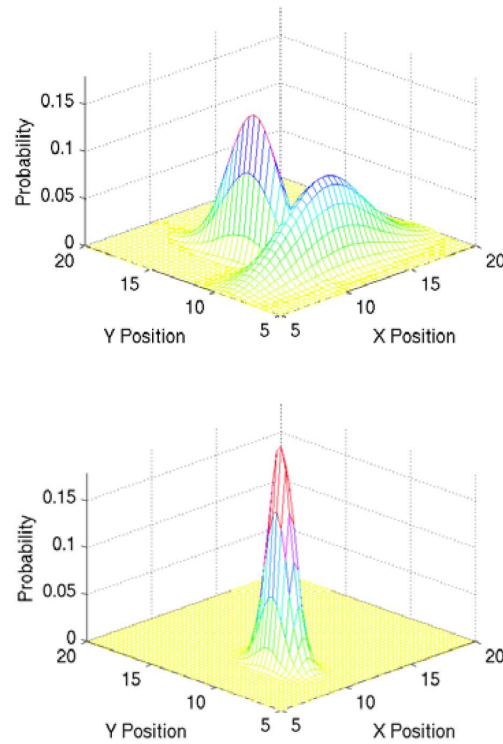


Figura 5.24: Ejemplo de fusión de distribuciones de incertidumbre. La imagen superior muestra las distribuciones originales. La imagen inferior muestra el resultado de la fusión.

1. Cada vez que se obtenga una nueva medida de GPS se calculará la posición actual del vehículo, relativa a la anterior medida de GPS.
2. Posteriormente, se calculará al ángulo absoluto del vector que une las dos posiciones anteriormente mencionadas, obteniéndose el correspondiente vector de rotación estimado q_{rob} , tal como se muestra en la Fig. 5.25.
3. Además, también se actualizará la velocidad lineal v_{rob} de acuerdo a la nueva estimación de la orientación.

Así pues, el vector de estado global del vehículo y su covarianza se actualizan como se muestra en las ecuaciones 5.30 y 5.31.

$$X_{robF}^0 = \left(X_x^{fusion} \quad 0 \quad X_y^{fusion} \quad \dots \right)^T \quad (5.30)$$

$$P_{robF}^G = \begin{pmatrix} P_{xx}^{fusion} & 0 & P_{xy}^{fusion} & \dots \\ 0 & 0 & 0 & \dots \\ P_{yx}^{fusion} & 0 & P_{yy}^{fusion} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad (5.31)$$

Con el objeto de obtener la mejor estimación en las huellas, éstas se generan sincronizadamente con las lecturas de GPS. De esta forma, en el momento que se dan las condiciones para

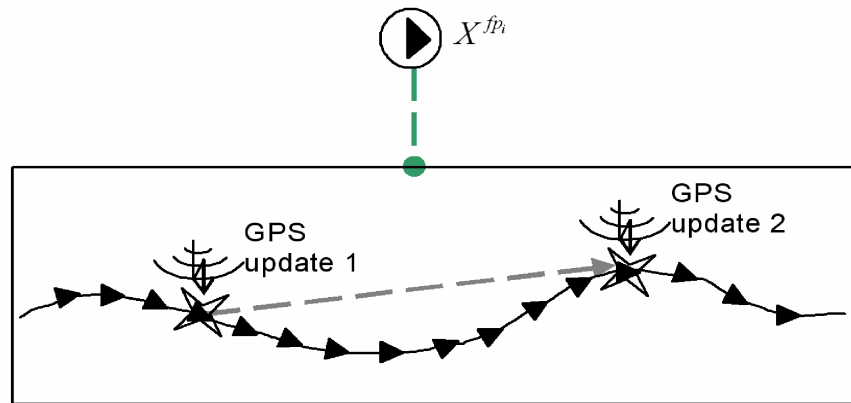


Figura 5.25: Detalle de un sub-mapa en el que se muestra la estimación de la orientación del vehículo a partir de la fusión con los datos del GPS. La estimación del mapa local se representa a través de las estimaciones de pose locales (triángulos negros). La flecha gris indica la orientación estimada del vehículo en el segundo punto de medida de GPS

la generación de una nueva huella, el sistema espera hasta que esté disponible una nueva lectura GPS.

5.7.3. Fusión en el SLAM de alto nivel

Uno de los problemas más comunes al usar un GPS en áreas urbanas muy pobladas es la pérdida total de señal. Normalmente, esto es debido a la *visibilidad* de un número reducido de satélites. Ésta puede ser causada por diferentes circunstancias como paso por túneles, puentes, zonas de árboles o incluso edificios altos.

En este método se considerará una pérdida *larga* de GPS cuando no existan medidas disponibles durante más de dos huellas consecutivas. En ese caso, la corrección del vector de estado implica una corrección del mapa global. Esta corrección se verá más acentuada en la parte del recorrido donde la señal de GPS no estaba disponible, tal como se muestra en la Fig. 5.26.

La forma en la que se introducirán nuevas huellas en el sistema, cuando el GPS no está disponible es ligeramente diferente a cuando sí lo está. En el primer caso, el sistema depende únicamente de la información visual, que en definitiva proporciona sólo información relativa local. Por lo tanto, las poses de las huellas serán introducidas de forma relativa a la huella anterior $X_{fp_i}^{fp_{i-1}}$. Sin embargo, cuando el GPS sí está disponible, la incertidumbre de medida del GPS será directamente referida al sistema de referencia global, de forma que no habrá error acumulativo. Por lo tanto, las huellas serán definidas relativas al sistema de referencia global $X_{fp_i}^0$ (ver Fig. 5.27 y Fig. 5.28).

A la hora de corregir el mapa, como consecuencia de los diferentes procedimientos anteriormente mencionados, las poses de las huellas relativas serán modificadas en un mayor grado que las de las huellas absolutas. Esto tiene sentido, ya que en esas huellas la incertidumbre acumulada se incrementa de manera continuada con el tiempo, a diferencia de las huellas absolutas, las cuales poseen un error absoluto *limitado*.

Por lo tanto, al perder la señal de GPS por un tiempo prolongado y volverla a recuperar de nuevo, de forma general la nueva pose estimada tendrá una incertidumbre menor que la acumulada hasta ese momento. De esta forma, se puede aprovechar este hecho para reducir la

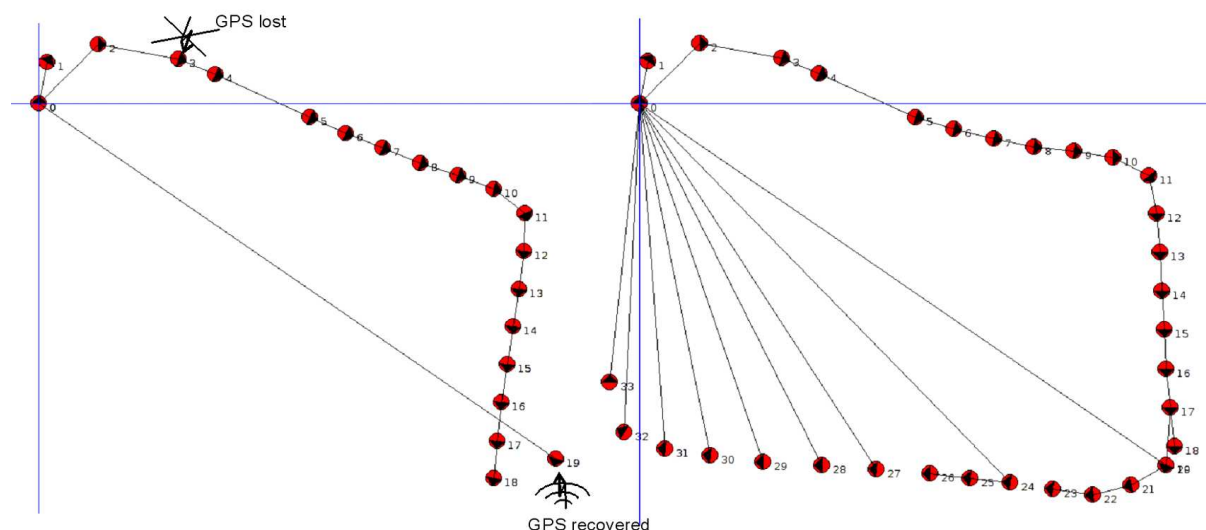


Figura 5.26: Diagrama de huellas MLR antes (izquierda) y después (derecha) de la recuperación de la señal de GPS. Las huellas en las que el GPS estaba disponible están expresadas en coordenadas globales (19..33). Las huellas sin GPS disponible están expresadas relativas a la huella anterior.

incertidumbre del tramo del recorrido estimado únicamente con información visual, tal como se muestra en la Fig. 5.27 y Fig. 5.28.

Así pues, para realizar la corrección del mapa, tan pronto como se recupere la señal de GPS, se creará una nueva huella relativa al sistema de referencia global. Posteriormente se añadirá una relación entre la última huella y la actual (ver Fig. 5.26). El mapa resultante será optimizado aplicando el método MLR. El efecto de recuperar la señal de GPS será, por lo tanto, equivalente a la detección de un cierre de lazo, corrigiéndose el tramo obtenido sin GPS de forma que el mapa final sea globalmente consistente. Para poder estimar la orientación del vehículo, dado que no existe una medida de GPS inmediatamente anterior, será necesario esperar dos medidas de GPS consecutivas y calcular la nueva orientación a partir de ellas.

5.8. Conclusiones y aportaciones

En el presente capítulo se han descrito las diferentes tareas y métodos englobados dentro del SLAM de alto nivel. Como conclusiones principales se destacan las siguientes:

- Se ha desarrollado un nivel topológico adicional sobre el nivel métrico explicado en el capítulo anterior. Éste está constituido por las denominadas *huellas* relacionadas entre sí, asociadas a cada sub-mapa y que definen los sistemas de referencia de los mismos.
- Se ha implementado un método de relación entre sub-mapas locales, basado en la independencia de los mismos. Se han desarrollado además métodos probabilísticos de cambio de sistemas de referencia, tanto de las poses del vehículo y las marcas, como de sus incertidumbres asociadas, para adecuar las transiciones entre los sub-mapas.
- Además, se ha desarrollado un método de detección de lugares previamente visitados, de cara a identificar situaciones de cierre de lazo mediante el sensor de visión. Este método se basa en el uso de unas huellas especiales denominadas *huellas SIFT*. Para obtener la correlación entre huellas, dentro de una región de incertidumbre del vehículo, se han estudiado

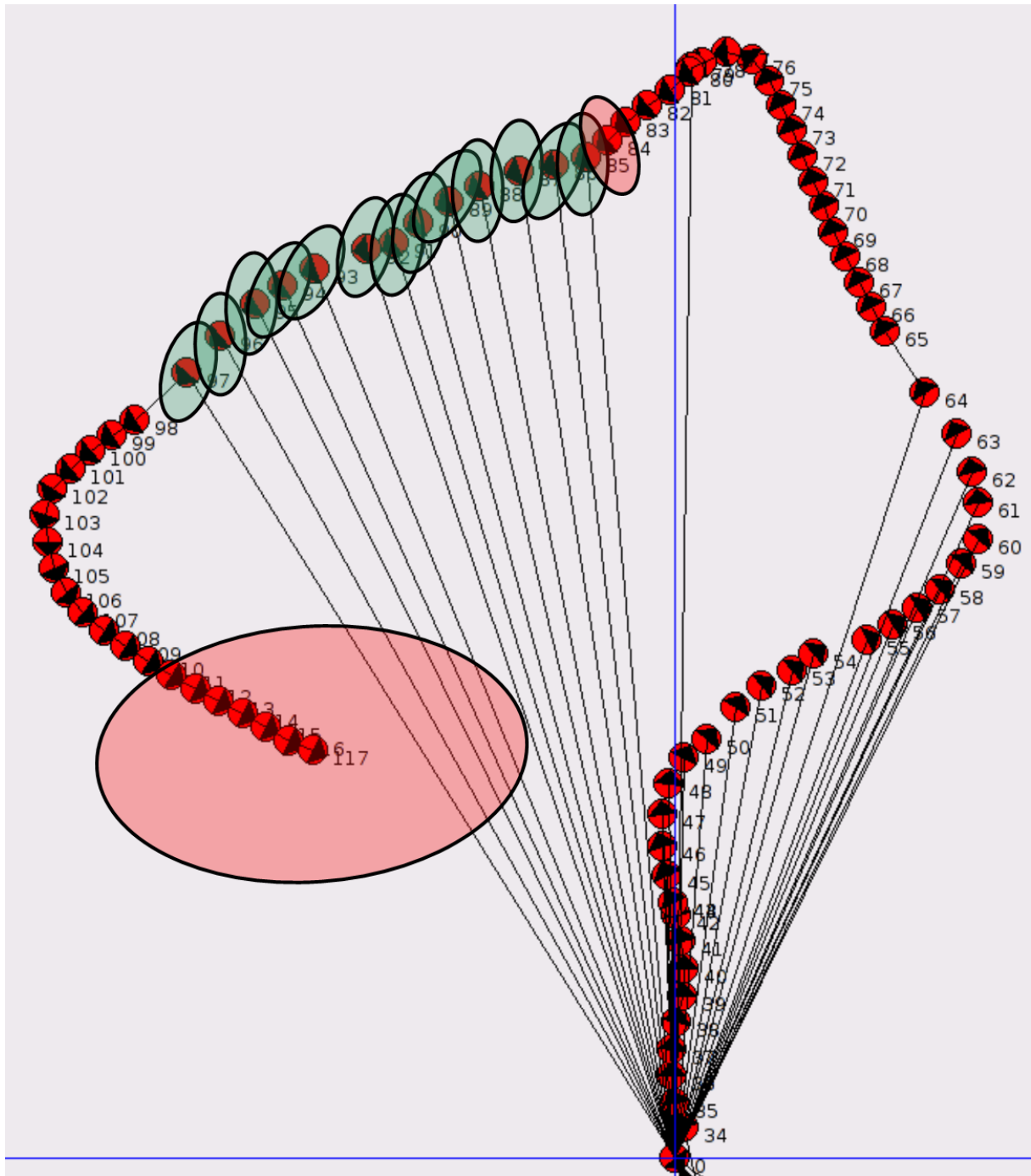


Figura 5.27: Diagrama MLR de huellas antes de la recuperación de la señal de GPS. Se muestran las incertidumbres globales de las huellas de un tramo con GPS disponible, aproximadamente constantes con el tiempo (color verde). La última huella introducida termina un tramo sin GPS disponible, apreciándose un incremento notable en el tamaño de la región de incertidumbre global (color rojo).

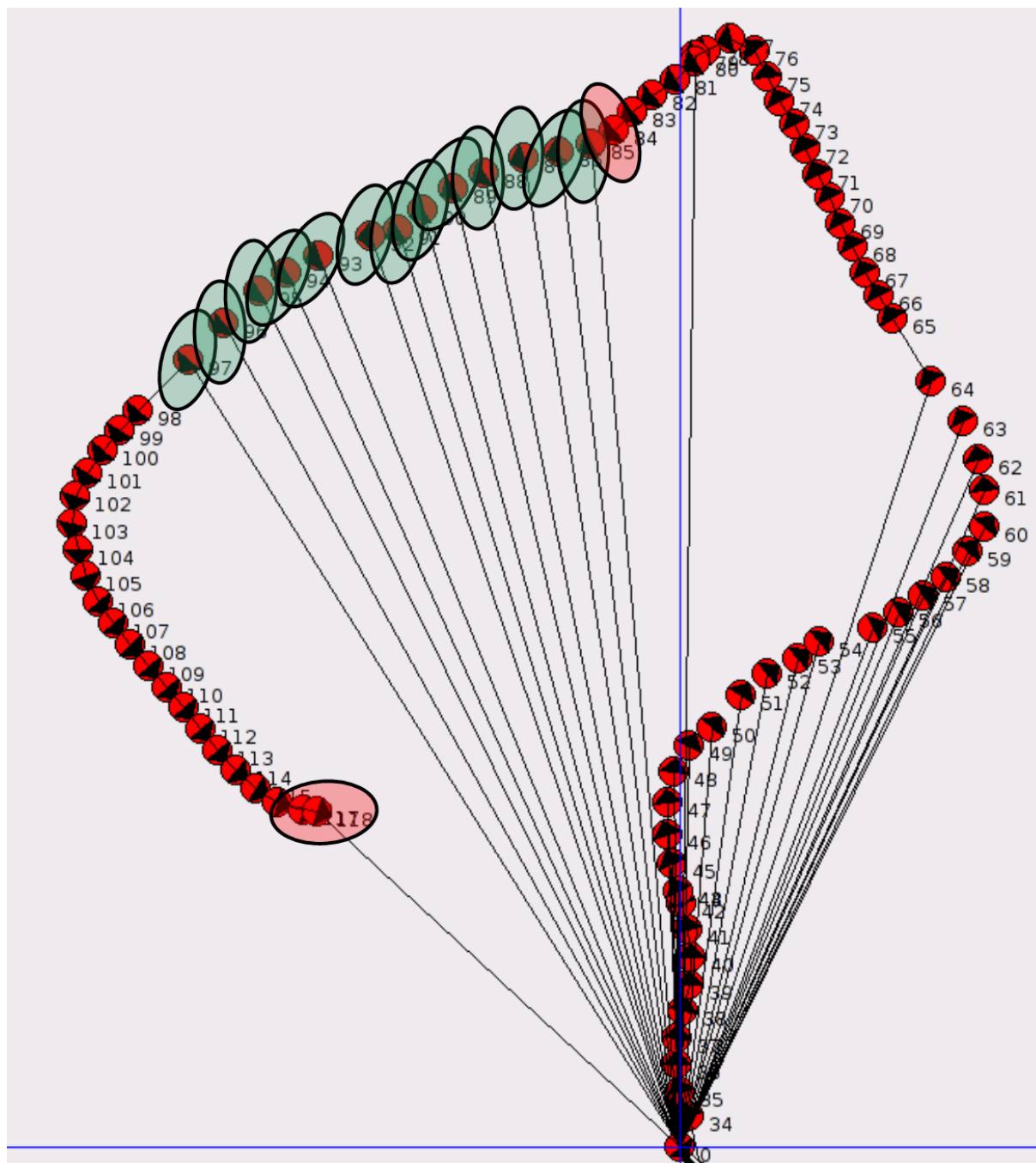


Figura 5.28: Diagrama MLR de huellas después de la recuperación de la señal de GPS. Se muestran las incertidumbres globales del tramo con GPS sin apenas variación. Tras la optimización del mapa, una vez recuperada la señal de GPS, la incertidumbre al final del tramo sin GPS se reduce notablemente. Se aprecia también que este último tramo es el que ha sufrido una mayor corrección.

dos métodos diferentes: uno de ellos se basa en variabilidad estadística y el otro hace uso del método RANSAC. Como resultado del estudio se concluye que el método RANSAC proporciona unos mejores resultados, siendo éste el elegido para la implementación.

- También se ha realizado un estudio entre dos métodos diferentes para realizar la optimización y corrección del mapa global. Por una parte, se ha desarrollado un método basado en una corrección lineal y por otra se ha implementado el método MLR. Los resultados de la comparativa demuestran que, debido a que el primero es una aproximación a la solución, el error en la corrección del mapa es mayor que el obtenido con MLR. Por otra parte, a pesar de que el tiempo requerido para la implementación del MLR es ligeramente mayor que en el caso lineal, ambos permiten un tiempo de cómputo global por debajo del límite de tiempo real establecido. Por lo tanto se elige la implementación del MLR como método de optimización y corrección.
- Por último, se ha presentado un método de integración de los datos de GPS en ambos niveles jerárquicos. Se ha implementado un método de estimación de la incertidumbre de la posición del GPS, así como un método de fusión de estos datos con los obtenidos por el sistema visual, teniendo en cuenta ambas incertidumbres, en el nivel bajo de SLAM. También se ha desarrollado una estrategia de estimación de la orientación, ya que ésta no está disponible por parte del GPS. En el nivel alto de SLAM se ha implementado un método de detección de recuperación de la señal de GPS, para optimizar el mapa después de pérdidas largas de la misma.

Como aportaciones, se destacan las siguientes:

- El desarrollo novedoso de la técnica de división en sub-mapas EKF, combinada con un nivel superior de corrección basado en el método MLR. Esto aporta una gran precisión en las estimaciones, a la par que una importante eficiencia en su implementación y consistencia para aplicaciones de vehículos en grandes entornos.
- El hecho de poder estimar la incertidumbre global del vehículo en todo momento, contribuye a esta eficiencia, en términos de tiempo de cómputo, a la hora de detectar cierres de lazo. En lo que concierne a la detección de lugares previsitados, la implementación de las huellas SIFT permite una gran especificidad en las correlaciones y, por tanto, una alta tasa de detección.
- Se presenta un sistema robusto, ya que usa información visual y GPS para estimar la posición del vehículo y crear un mapa no denso pero suficiente para mantener una buena estimación si falla el GPS, pudiéndose dar correcciones de cierre de lazos únicamente con información visual.
- Por último, la estrategia de integración de las medidas del GPS, de una forma probabilística y en ambos niveles jerárquicos de abstracción, permite el aprovechamiento de la complementariedad entre este sensor y el sensor visual, de una forma más eficiente que en otros métodos existentes en la literatura [51] [45].

Capítulo 6

Implementación práctica y resultados experimentales

6.1. Introducción

En este capítulo se describen los resultados experimentales más relevantes de la Tesis. En función de las diferentes fases de la misma, el sistema formado por la cámara estéreo y el ordenador portátil han sido aplicados tanto en entornos interiores como exteriores, sobre diferentes estructuras que serán descritas a continuación. Inicialmente se describirá el *hardware* utilizado, para posteriormente analizar los resultados de las pruebas.

6.2. Descripción del HW utilizado

La configuración básica en cuanto a HW es la que se muestra en la Fig. 6.1. Éste está constituido por una cámara estéreo conectada a un ordenador portátil.

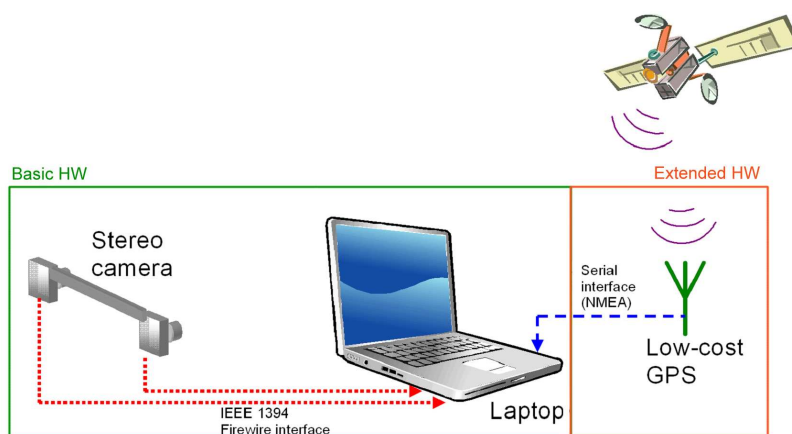


Figura 6.1: Esquema del montaje básico, compuesto por una cámara estéreo conectada a un ordenador portátil.

Los elementos principales utilizados son los que se detallan a continuación:

- *PC portátil.* El PC utilizado es un HP Pavilion DV8000 con un procesador AMD Turion-64 ML-32 (2.0GHz/512KB L2 Cache), 512MB 333MHz DDR SDRAM y disco duro 200GB. Además dispone de 4 puertos USB 2.0 y 1 puerto IEEE 1394 (FireWire).
- *Cámara estéreo.* Compuesta por dos unidades Unibrain Fire-i IEEE1394 de bajo coste (ver Fig. 6.2). El acceso se realiza mediante puerto Firewire IEEE1394, permitiendo tasas de transferencia de hasta 400 Mbps. Posee un CCD SONY® Wfine 1/4", el cual permite resoluciones de hasta 640x480 en diferentes modos de color y fps, tal como se muestra en la tabla 6.1. Por requerimientos de tiempo real, la configuración elegida es: 320x240, escala de grises y 30 fps.

La óptica original ha sido reemplazada por una óptica Unibrain 2047 de gran angular de 100° de campo visual horizontal (ver Fig. 6.3). Cada cámara posee dos puertos IEEE1394 de 6 pines, permitiendo una conexión en serie de ambas cámaras. La alimentación se puede proporcionar por los puertos firewire o alternativamente a través de un conector de alimentación específico. En el montaje de interiores, debido a la disponibilidad de un único puerto Firewire sin alimentación en el PC portátil, se utilizó un convertor DC-DC de 5 VDC a 9 VDC para obtener dicha alimentación. Este convertor obtiene los 5V a partir de un puerto USB.



Figura 6.2: Cámara Unibrain Fire-i utilizada en el sistema.

Frame rates (fps)	30	15	7.5	3.75
Video Modes				
640x480 RGB-24 (24 bits)	-	X	X	X
640x480 YUV 4:2:2 (16 bits)	-	X	X	X
640x480 YUV 4:1:1 (12 bits)	-	X	X	X
640x480 Y-Mono8 (8 bits)	X	X	X	X
320x240 YUV 4:2:2 (16 bits)	X	X	X	X
160x120 YUV 4:4:4 (24 bits)	X	X	X	-

Tabla 6.1: Resoluciones y modos de color disponibles en función de la velocidad de muestreo de la cámara.

Para conectar ambas cámaras al único puerto disponible se utilizó el montaje serie entre ambas cámaras, tal como se muestra en la Fig. 6.4. El montaje final de las cámaras en configuración estéreo establece un *baseline* de 12 cm, tal como se muestra en la misma figura.

Para las pruebas preliminares en interiores el sistema HW de pruebas empleado es el que se muestra en la Fig. 6.5 basado en una plataforma robótica móvil. En este caso, dicha plataforma robótica sólo se usó de forma pasiva, es decir, siendo dirigida de forma teleoperada por el recorrido de prueba.

Para las pruebas en exteriores, además del HW básico descrito en la Fig. 6.1 se ha añadido un GPS de bajo coste conectado al ordenador portátil. Adicionalmente, se ha hecho uso de un RTK-GPS de referencia para poder estimar el error en las estimaciones. Todo ello se muestra en la Fig. 6.6.

El sistema HW extendido empleado en este caso, es el que se muestra en la Fig. 6.7. El vehículo empleado es un Citroën C4 con cambio automático, como se puede ver en la misma figura.

En cuanto a la cámara estéreo, dado que en exteriores las marcas visuales se encuentran generalmente a una distancia mucho mayor que en interiores, se aumentó el *baseline* hasta un valor de 40 cm. El montaje realizado se basa en las mismas cámaras que para interiores (ver Fig. 6.8). Por otro lado, para mejorar la sincronización de las cámaras, se instaló en el PC portátil una tarjeta PCMCIA con 2 puertos Firewire, de forma que ambas cámaras se conectan directamente al portátil en lugar de hacerlo en serie, como en las pruebas de interiores.

Los elementos adicionales utilizados en las pruebas en exteriores son los que se detallan a



Figura 6.3: Lente de gran angular utilizada en la cámara.

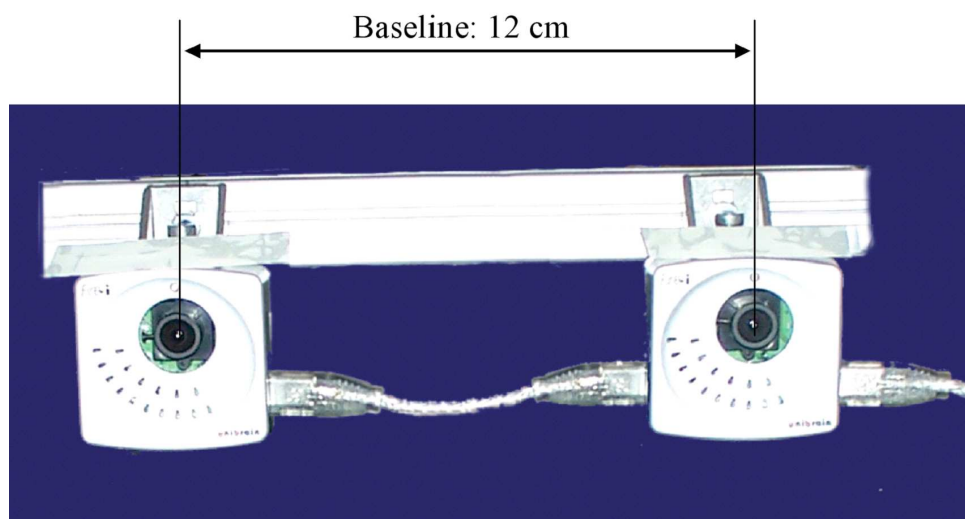


Figura 6.4: Montaje de la cámara estéreo con conexionado en serie para interiores.

continuación:

- *GPS de bajo coste.* El GPS utilizado es un Navilock NL-302U (ver Fig. 6.9) que se conecta a través del puerto USB del PC portátil. Es compatible con el protocolo NMEA 0183 y posee una velocidad de actualización de 1 Hz.
- *GPS de referencia.* Para obtener la referencia en los distintos recorridos de prueba se utilizó un RTK-GPS Maxor GGDТ (ver Fig. 6.10) a una velocidad de 5 Hz de actualización de datos. El sistema es compatible con el protocolo NMEA 0183, transmitiendo vía serie los datos de posición.

Este GPS es capaz de realizar correcciones de medida a partir de las correlaciones con la posición de una estación base ubicada en el tejado de la Escuela Politécnica de la Universidad de Alcalá. El sistema recibe la información de estas correcciones vía radio UHF con una potencia de hasta 2W, a través de una antena ubicada en la parte superior del vehículo. Debido a que algunos recorridos de prueba se encontraban ubicados a una distancia elevada de la estación base (hasta 50 Km), en esos casos se emplearon módulos GSM/GPRS para transmitir las mencionadas correcciones. En el esquema de la Fig. 6.6 mostrado anteriormente se detalla el conexionado realizado para las diferentes pruebas realizadas. En él se muestra cómo el RTK-GPS de referencia recibe la señal de GPS junto con las correcciones. Estas correcciones pueden ser recibidas vía radio a través de la correspondiente antena, o bien vía GSM usando los módulos GSM/GPRS. En ambos casos las correcciones son enviadas desde la estación base. Una vez que el GPS ha procesado la señal, éste la envía al PC portátil vía serie por medio del protocolo estándar NMEA 0183.

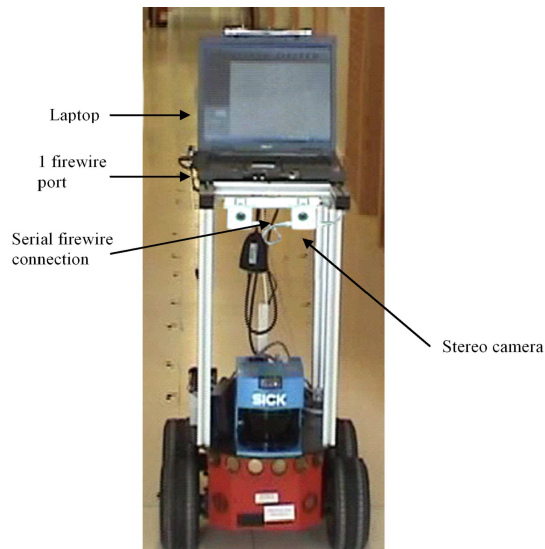


Figura 6.5: Sistema desarrollado para entornos interiores.

6.3. Resultados del sistema SLAM

Para validar el funcionamiento del sistema presentado en esta Tesis, se han realizado numerosas pruebas en distintos entornos y en diferentes condiciones. En los siguientes apartados se muestran los resultados de los recorridos de pruebas más representativos de entre todos los realizados. Inicialmente, se realizaron ensayos del sistema en entornos interiores de grandes dimensiones, para posteriormente trasladar las pruebas a entornos exteriores de mayores dimensiones con un consecuente aumento de la complejidad del problema a resolver.

6.3.1. Resultados preliminares en entornos interiores

En las pruebas realizadas en interiores únicamente se utilizó el sensor de visión ya que la señal GPS en este tipo de entornos no es utilizable. De entre todas ellas destacamos, a modo de ejemplo, la que se detalla a continuación. El montaje llevado a cabo para el ensayo es el que se muestra en la Fig. 6.5. La secuencia de prueba fue tomada mientras el robot recorría la tercera planta de la Escuela Politécnica de la Universidad de Alcalá. El camino completo desde el punto de partida hasta el punto del cierre de lazo tiene un perímetro de 283.25 m (Ver Fig. 6.11).

En la Fig. 6.12 se muestra la estimación del camino recorrido realizada en base únicamente al *low level SLAM*. La estimación se compara con una referencia (*ground truth*) obtenida manualmente. Usando el método propuesto, se observa una desviación en el tercer giro, lo cual conlleva un error acumulado en el resto del camino.

Si se representa el error acumulativo medio, según la ecuación 6.1, en la Fig. 6.13, se puede apreciar que éste no excede los 10 m en el eje Z. Esto es debido fundamentalmente al error aparecido en la tercera curva, mencionado anteriormente.

$$\varepsilon_n = (1/n_{total}) \sum_{i=0}^n |X_i - Xref_i| \quad (6.1)$$

Respecto al *high level SLAM*, en la Fig. 6.14 se muestra la representación del mapa estimado por el sistema, incluyendo todas las marcas y huellas. La secuencia representa el mapa justo

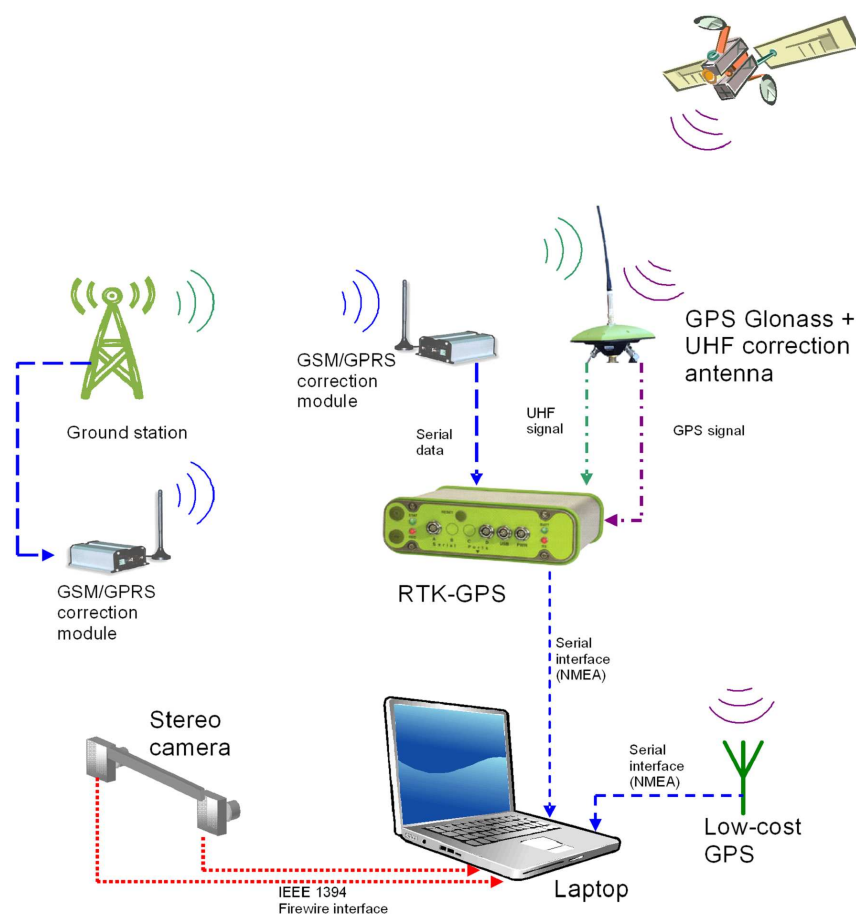


Figura 6.6: Esquema del montaje para exteriores, basado en el hardware básico añadiéndole un GPS de bajo coste. Como sistema de referencia se utiliza un RTK-GPS y la estación base para las correcciones de posición.

antes y después de la situación de cierre de lazo. El método de alto nivel empleado en este ensayo preliminar es el método lineal. Como se puede observar, el sistema mantiene la consistencia global una vez ha sido corregido. Después de esta situación el sistema sigue siendo capaz de volver a detectar huellas previsitadas.

Respecto al tiempo de procesamiento, en la Fig. 6.15 se muestran los tiempos de cómputo totales a lo largo del recorrido. Como se puede observar el sistema es capaz de funcionar por debajo de la restricción de tiempo real (33 ms), manteniendo los tiempos casi constantes a lo largo del recorrido. En la tabla 6.2 se muestran los tiempos promedio de procesamiento para algunas de las tareas más importantes del proceso completo. Éste se ha dividido entre las tareas asociadas al bajo nivel de SLAM y las dedicadas al alto nivel de SLAM. Respecto a las primeras se observa que el mayor tiempo de cómputo es el dedicado a la tarea de inicialización de marcas, en las que el área de búsqueda se extiende a lo largo de toda la recta epipolar. En estas pruebas la restricción de la recta epipolar no había sido aplicada. En cuanto al alto nivel de SLAM, como se puede observar, el tiempo dedicado al proceso de comparación de huellas, así como a la corrección del mapa en el momento del cierre del lazo, conteniendo 1630 marcas, es significativamente mayor que el tiempo real. Debe tenerse en cuenta que ambas tareas están fuera del proceso continuado de autolocalización del robot, el cual es llevado a cabo por parte del bajo nivel de SLAM. Esto significa que no es necesario completar estas tareas dentro del tiempo de un simple frame. Por lo tanto, es posible obtener un resultado positivo en la comparación de



Figura 6.7: Vehículo utilizado para la implementación en exteriores. Abajo a la izquierda: vista frontal del vehículo con la cámara estéreo incorporada. Abajo a la derecha: Vista interior del vehículo con el PC portátil y el GPS de bajo coste.

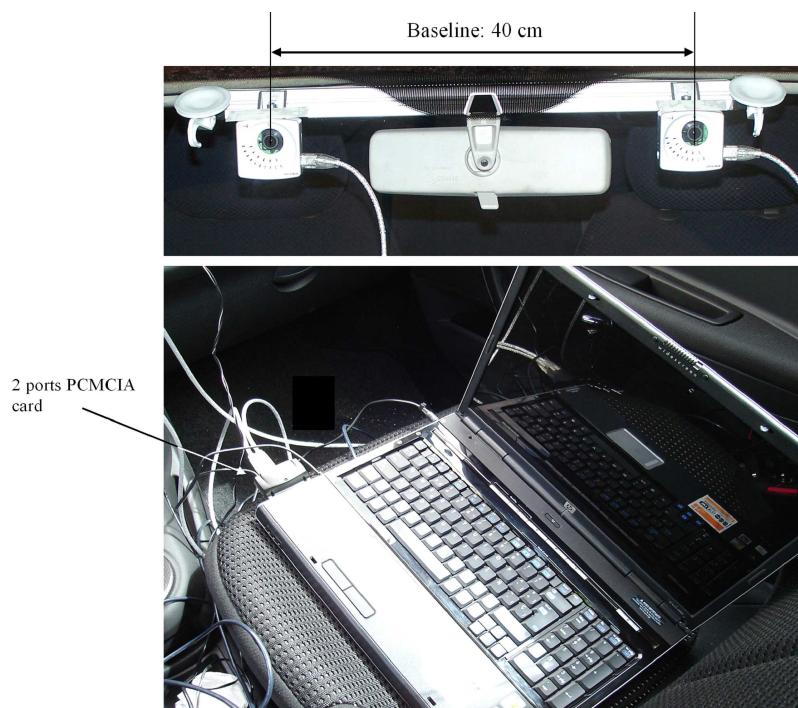


Figura 6.8: Montaje de la cámara estéreo para exteriores. Abajo se muestra el PC portátil con la tarjeta de adquisición IEEE1394 conectada.



Figura 6.9: GPS de bajo coste utilizado en el sistema.



Figura 6.10: RTK-GPS utilizado como referencia. Abajo a la izquierda: Ubicación del RTK-GPS dentro del vehículo de pruebas. Abajo a la derecha: Antena receptora de señal GPS + correcciones de posición vía radio.

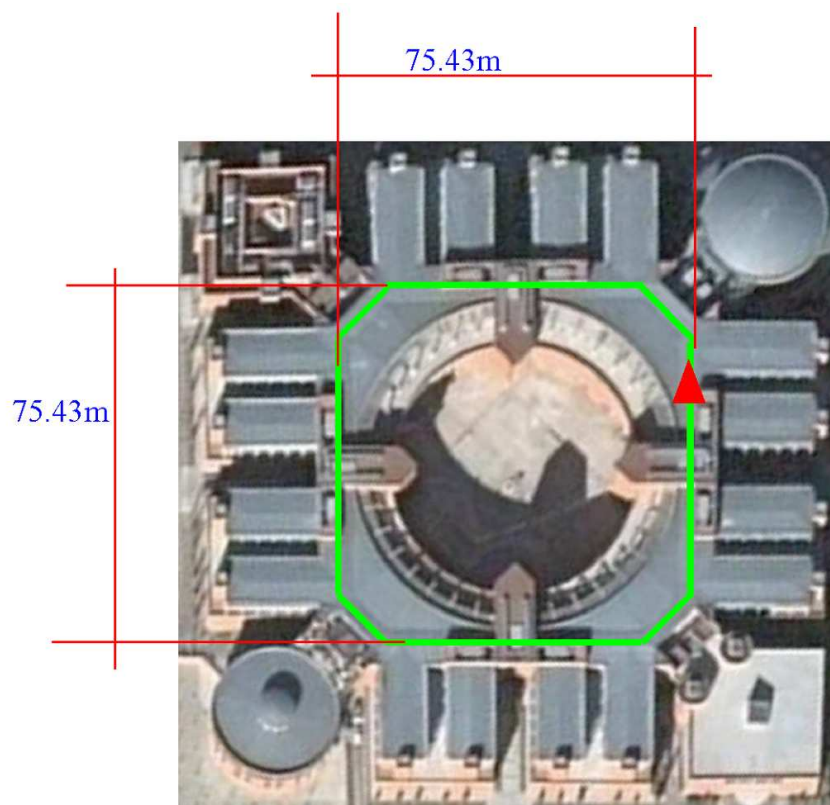


Figura 6.11: Vista aérea del camino recorrido por el robot, marcada en color verde. La flecha roja indica la posición de inicio y cierre de lazo.

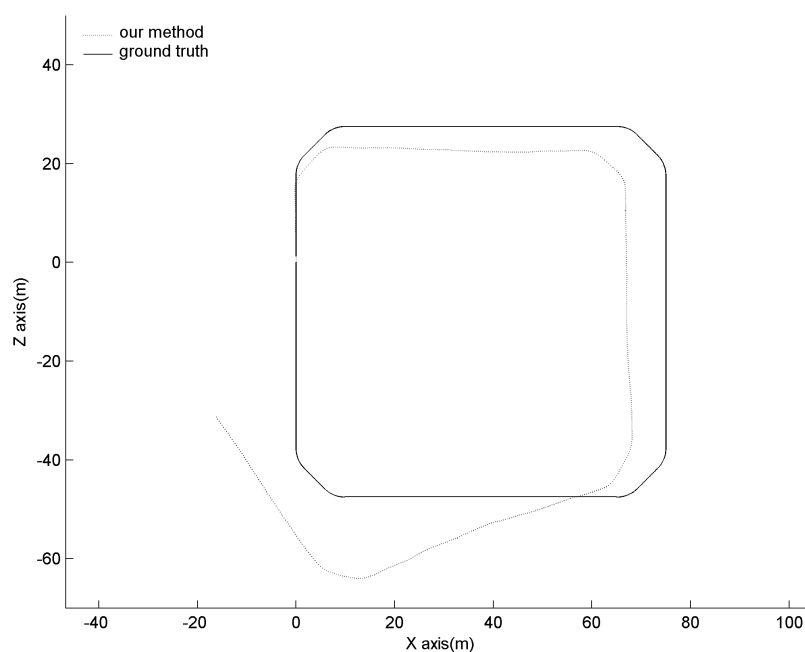


Figura 6.12: Estimación del camino recorrido por el robot realizada por el bajo nivel. La referencia se dibuja en línea continua

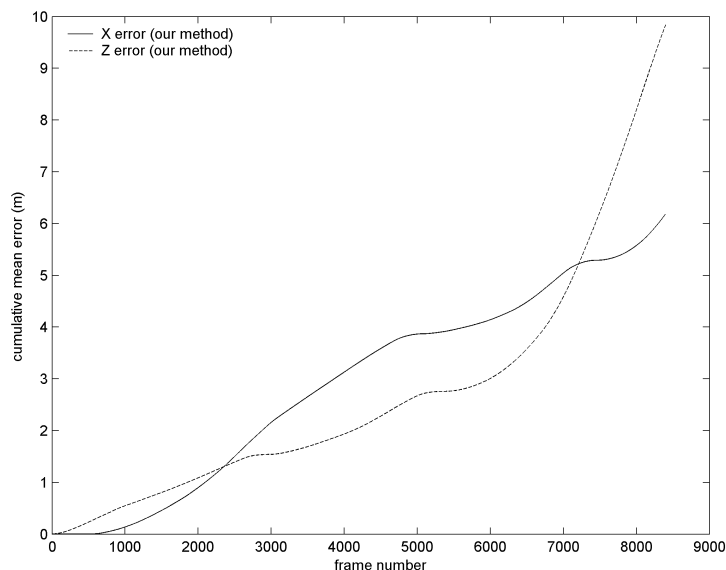


Figura 6.13: Error acumulativo medio para el eje X y Z en función del número de frame n.

huellas unos pocos frames después de que éste haya sido detectado realmente. En ese momento se puede volver atrás en el algoritmo y comenzar la tarea del cierre de lazo, incluyendo también los últimos frames procesados. Todo esto implica que ambas tareas puedan ser procesadas en paralelo, manteniéndolas fuera del cómputo de tiempo real.

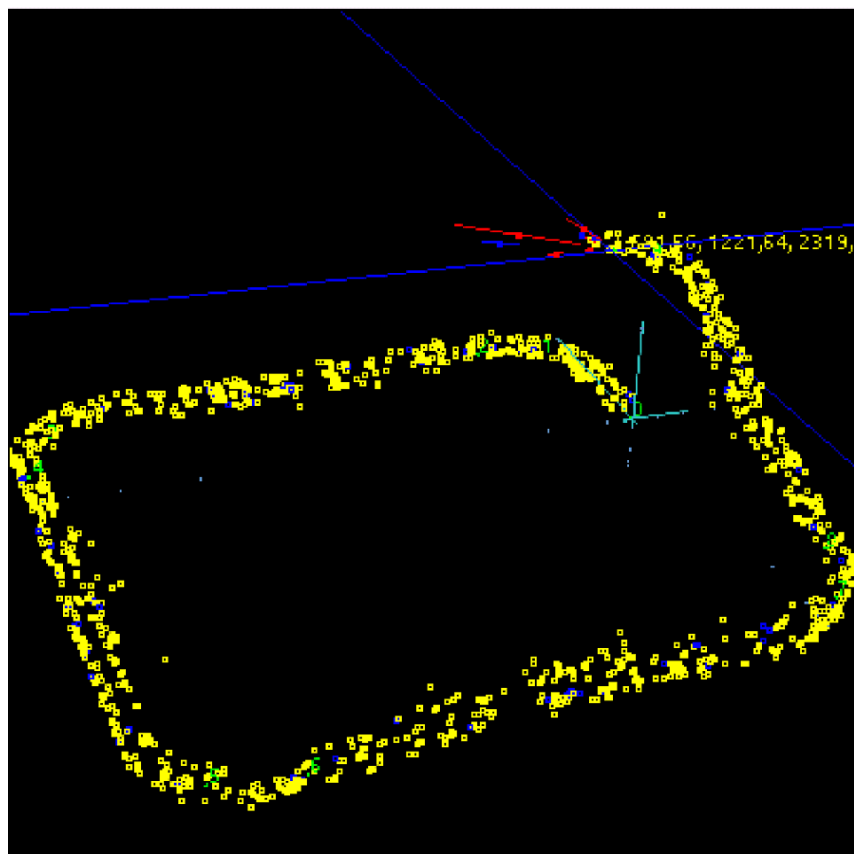
Low level SLAM processing times		High level SLAM processing times	
Implementation		Path length	283.25 m
Number of features / frame	5 (minimum)	Number of loops	1
Filter step	Average Time	Average speed	5 Km/h
Measurements	3 ms	Number of SIFT fingerprints	8
Filter update	5 ms	Number of features	1630
Feature initializations	20 ms	Time	
		Fingerprint matches	630 ms
		Loop closing (Including graphic representation)	6000 ms

Tabla 6.2: Datos relevantes para el recorrido en interiores.

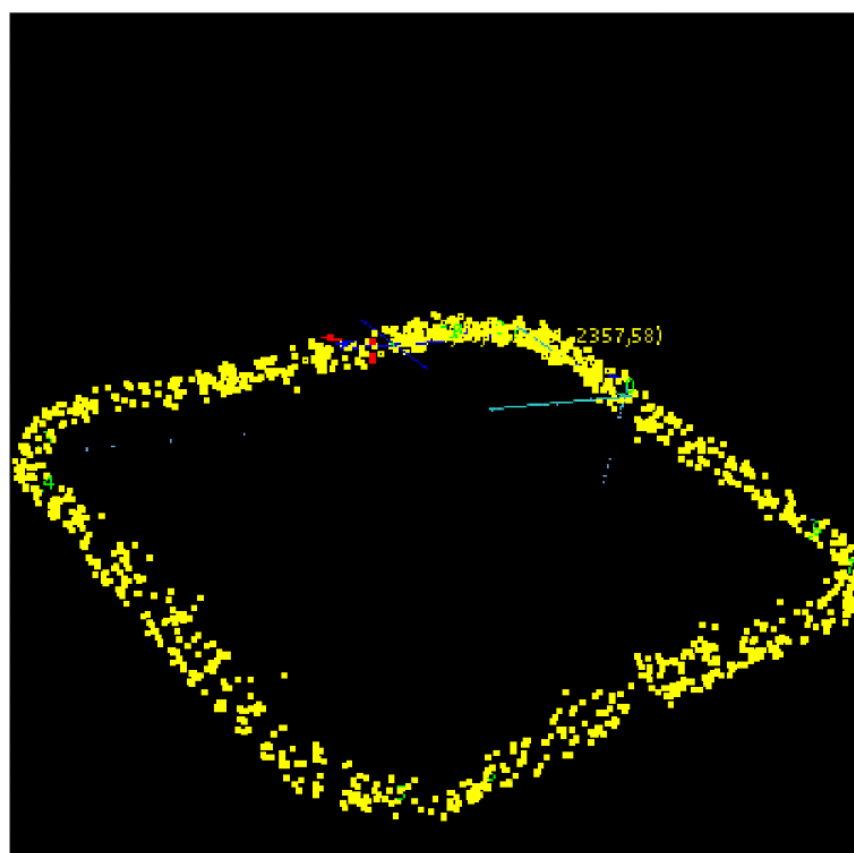
Como conclusiones fundamentales de esta prueba cabe destacar el reducido error obtenido en las estimaciones, incluso empleando únicamente el bajo nivel de SLAM. Los tiempos de cómputo, además, se encuentran por debajo de la restricción de tiempo real en todo el recorrido.

6.3.2. Resultados en grandes entornos exteriores

Una vez validado el sistema en entornos interiores se continuó realizando ensayos en entornos exteriores de mayores dimensiones cercanos a la aplicación real. En las primeras pruebas realizadas los ensayos se llevaron a cabo en entornos semi-urbanos, adaptando el sistema básico para embarcarlo en un automóvil. Sin embargo, dado que en grandes entornos la distancia a la que se encuentran la mayoría de las marcas es superior que para pequeños entornos, el *baseline* del



(a) Representación inmediatamente antes de la corrección del mapa



(b) Representación inmediatamente después de la corrección del mapa

Figura 6.14: Representación del mapa en una situación de cierre de lazo. Las marcas de bajo nivel se representan de color amarillo, mientras que las visibles y correctamente medidas se representan en color rojo. Los números verdes muestran las ubicaciones de las huellas. La posición del robot se indica en el centro de una gran cruz azul.

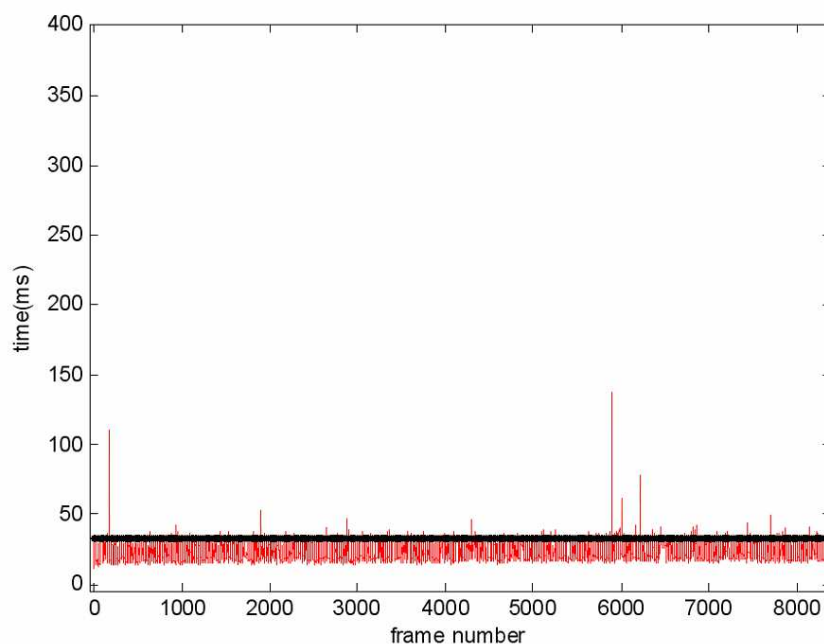


Figura 6.15: Tiempos de cómputo por frame a lo largo del recorrido, representados en rojo. El límite de tiempo real se representa como una línea constante de color negro en 33 ms.

sistema de cámaras utilizado se aumento hasta los 30 cm conforme al estudio teórico realizado en el capítulo 4. Para comprobar el comportamiento del sistema basado únicamente en visión, inicialmente se llevaron a cabo las diversas pruebas sin el uso del sensor GPS. Posteriormente se añadió el GPS y se hicieron pruebas en entornos con dificultades para recibir la señal GPS. A continuación se analizan los resultados de los ensayos más significativos llevados a cabo en el proceso de validación del sistema junto a las configuraciones utilizadas para cada caso.

▪ **Recorrido A. Sin uso del GPS.**

En primer lugar se muestran los resultados obtenidos en el recorrido que se muestra en la Fig. 6.16 en los alrededores del campus externo de la Universidad de Alcalá. El camino completo recorrido, desde el inicio hasta el cierre del segundo lazo fue de 2965 m. El punto de partida del camino está marcado con el indicador *S*, siendo el sentido del recorrido el que se indica con las flechas. Se completó un primer lazo interior (flechas cerradas) al llegar al punto con el indicador *I*. El segundo gran lazo exterior (flechas abiertas) se completó al llegar al punto *F*, coincidiendo con el final del recorrido. Los resultados del método propuesto se muestran junto con la referencia obtenida mediante el RTK-GPS. El método de corrección de alto nivel empleado en este ensayo fue el lineal.

Si se observan los resultados desde el inicio hasta el cierre del primer lazo (ver Fig. 6.18(a)) se aprecia que la similitud entre el camino estimado, una vez cerrado el lazo y realizada la corrección, y la referencia GPS es mucho mayor que antes de la corrección. Para el caso del cierre del lazo general (ver Fig. 6.18(b)), se observa asimismo que la corrección de la estimación del camino recorrido, una vez cerrado el lazo, mejora notablemente con respecto a la situación anterior. Además, si se observa el camino recorrido antes de la corrección, se aprecia que en el último tramo el error es mucho mayor. Esto es debido a dos situaciones: por una parte, en el giro anterior al último segmento se suceden varias oclusiones casi totales producidas por el paso de un autobús, entre otros vehículos. Respecto a la longitud del tramo, esta es más corta que

el valor real debido a la ausencia casi total de marcas en el entorno, al existir zonas de campo abierto, tal como se puede apreciar en la Fig. 6.17. En la Fig. 6.19 se representa el error en cada eje $\varepsilon_n = |X_n - Xref_n|$ una vez completado el recorrido completo, es decir, una vez cerrados ambos lazos. Si se observa la primera parte, correspondiente al primer lazo, se puede ver que el error es relativamente bajo comparado con el del segundo lazo, debido al problema del campo abierto.



Figura 6.16: Vista aérea del camino A recorrido por el vehículo (referencia) dibujado en color magenta. El punto de partida se indica con el flag *S*. El punto de cierre del primer lazo se indica con el flag *I*. El punto de cierre del segundo lazo, y final del recorrido, se indica con el flag *E*

En la Fig. 6.19 se representa el error en cada eje $\varepsilon_n = |X_n - Xref_n|$ una vez completado el recorrido completo, es decir, una vez cerrados ambos lazos. Si se observa la primera parte, correspondiente al primer lazo, se puede apreciar que el error se ha reducido notablemente en ambos ejes.

Respecto al tiempo de procesamiento, la Fig. 6.20 muestra los tiempos a lo largo de la mitad del recorrido del vehículo. La principal conclusión que se extrae del gráfico es que el tiempo de procesamiento medio se mantiene prácticamente constante a lo largo del recorrido del vehículo, siendo éste inferior a la restricción de tiempo real la mayor parte del tiempo y siendo superado únicamente en el cálculo de algunas huellas y en el cierre de lazos. En la tabla 6.3 se muestran los tiempos de cómputo por cada tarea del algoritmo. Como se puede observar, éstos no varían significativamente con respecto a los obtenidos en el recorrido en entornos interiores, explicado previamente.

Como conclusiones principales cabe destacar la validación del funcionamiento del sistema básico propuesto, a pesar del gran tamaño del entorno y de las dificultades inherentes a los



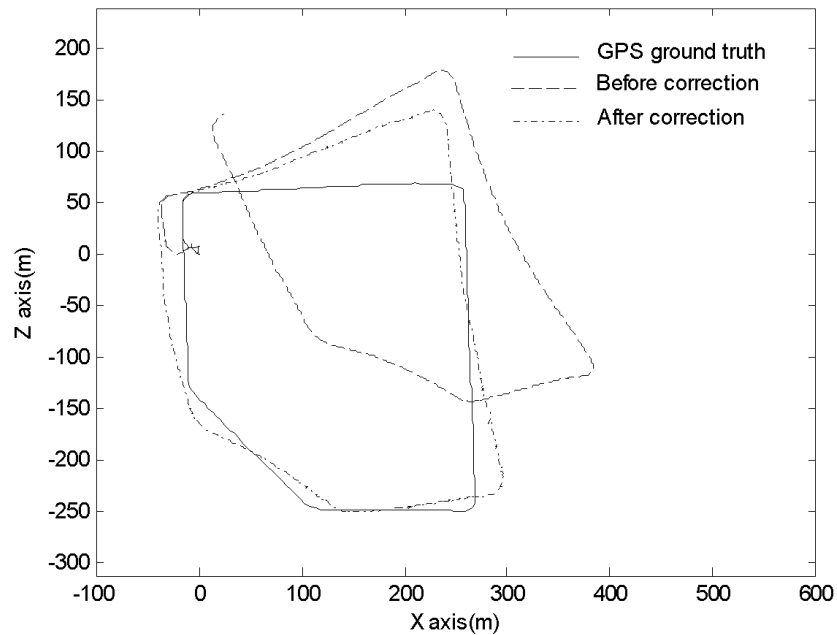
Figura 6.17: Causas del error de estimación en el último tramo. En las imágenes superiores se observa la oclusión casi total de las mismas por parte de un autobús. En las imágenes inferiores se muestra el entorno del último tramo tras la oclusión, en el que se aprecia la ausencia casi total de texturas donde extraer marcas visuales relativamente cercanas.

sistemas de visión computacional en entornos exteriores. Sin embargo, cabe reflexionar, por un lado, en el hecho de que la precisión de la estimación del bajo nivel de SLAM se empobrece, sobre todo en el último tramo explicado anteriormente, debido a la dispersión y gran distancia de las marcas visuales. Esto conduce a la necesidad de aumentar el *baseline* de las cámaras. Por otra parte, se comprueba que, ante errores de estimación importantes, el método lineal es capaz de corregir el mapa, pero no de una manera óptima. Estos pobres resultados preliminares nos llevaron a buscar otros métodos de corrección del mapa de alto nivel más óptimos para esta aplicación, siendo el elegido el método MLR según se ha justificado en el capítulo 5. En cuanto al tiempo de procesamiento, se comprueba que éste sigue cumpliendo con las restricciones de tiempo real.

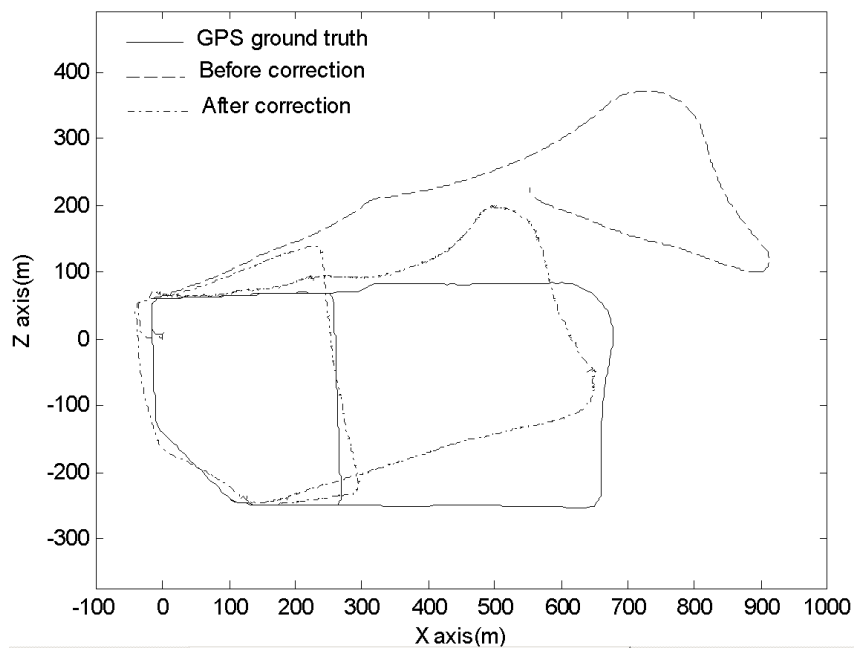
Debido a las conclusiones extraídas de las pruebas preliminares realizadas para grandes entornos, se decidió, por un lado, aumentar el *baseline* de las cámaras un poco más hasta llegar a los 40 cm, optimizando así el funcionamiento del *low level SLAM*. Por otra parte, se decidió sustituir el método de corrección lineal por el método de corrección MLR dentro del *high level SLAM* para mejorar la consistencia global del sistema. Los ensayos mostrados a partir de este momento se realizaron siguiendo la nueva configuración indicada. Estas pruebas se han llevado a cabo en diferentes entornos urbanos de entre los cuales se han destacado los que se analizan a continuación.

- **Recorridos B y C. Sin uso del GPS.**

La secuencia B se tomó también en un entorno urbano y cubrió el recorrido que se muestra en la Fig. 6.21. La velocidad media del vehículo fue aproximadamente la misma que en la primera secuencia, pero en este caso la longitud del recorrido fue de 2.19 Km. Éste contiene 4 lazos interiores, tomándose 8130 marcas de bajo nivel y 230 huellas.



(a) Estimación del camino recorrido hasta el primer cierre de lazo.



(b) Estimación del camino recorrido hasta el primer cierre del segundo lazo.

Figura 6.18: Estimación del camino recorrido. La línea rayada representa la estimación antes del cierre del lazo. La línea punto-raya representa la estimación después del cierre del lazo. La referencia está representada en línea continua.

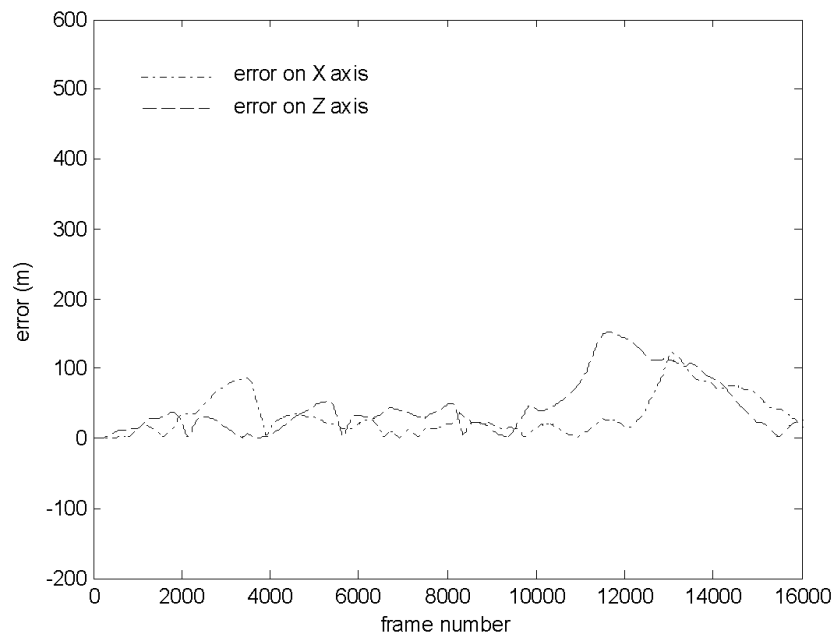


Figura 6.19: Representación del error en la estimación del recorrido en los ejes X y Z respecto de la referencia después de la corrección global

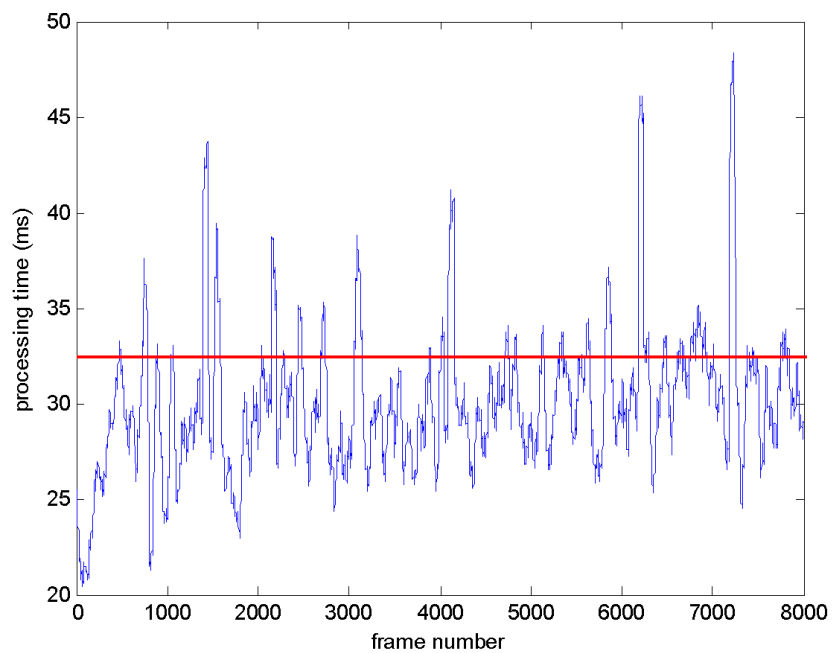


Figura 6.20: Tiempos de procesamiento totales a lo largo de la mitad del recorrido del vehículo.

Low level SLAM processing times		High level SLAM processing times	
Number of features / frame	5 (minimum)	Path length	2965 m
		Number of loops	2
Filter step	Average Time	Average speed	25 Km/h
Measurements	3 ms	Number of features	6210
Filter update	5 ms	Number of fingerprints	150
Feature initializations	7 ms	Number of SIFT fingerprints	9
		Time	
		Fingerprint matches	2 s
		Loop closing	1 s

Tabla 6.3: Datos relevantes para el recorrido A.

La secuencia C se tomó recorriendo con el vehículo el entorno urbano mostrado en la Fig. 6.22. La velocidad media del vehículo fue de aproximadamente 30 Km/h. El recorrido completo fue de 2.27 Km de longitud. Éste contiene 3 lazos, tomándose 7250 marcas de bajo nivel y 235 huellas.

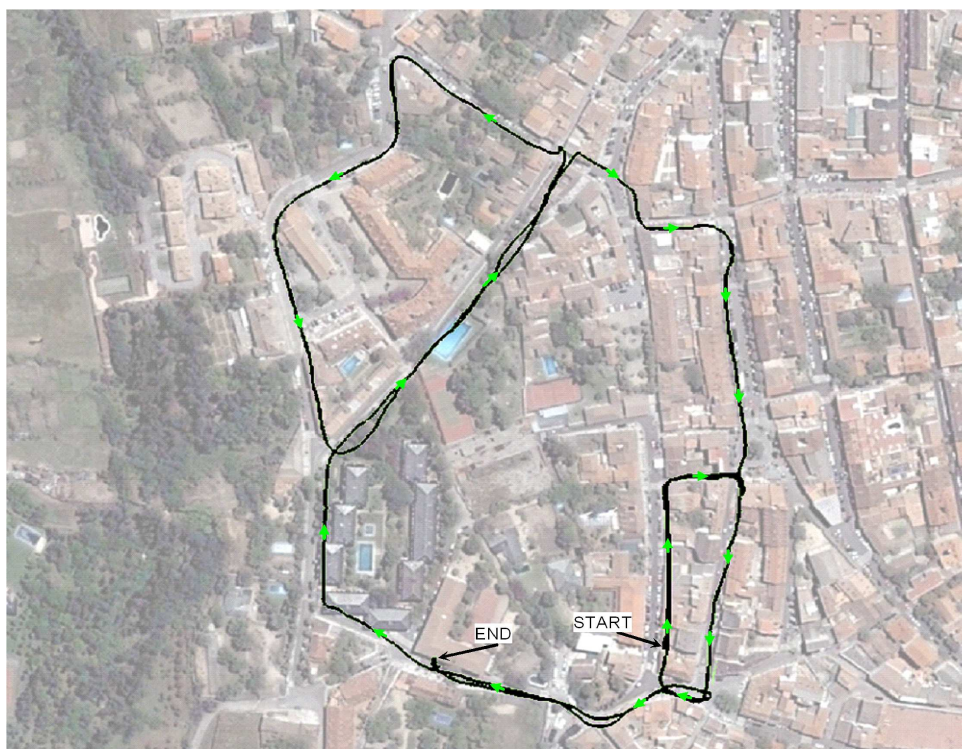


Figura 6.21: Vista aérea del recorrido de prueba B. Los puntos de inicio y final se indican en la figura.

Para evaluar el funcionamiento del sistema se han comparado los resultados con la referencia obtenida con el RTK-GPS. En la Fig. 6.23 se muestra la estimación del sistema SLAM comparada con la referencia para el recorrido B. Se puede observar la gran similitud de ambas estimaciones. A este hecho contribuye, por una parte la existencia de cuatro lazos en total frente a los tres del recorrido anterior, existiendo además un lazo global completo. Por otra parte, la existencia de un mayor número de edificios, aportando una mayor riqueza de textura, mejoró la precisión en

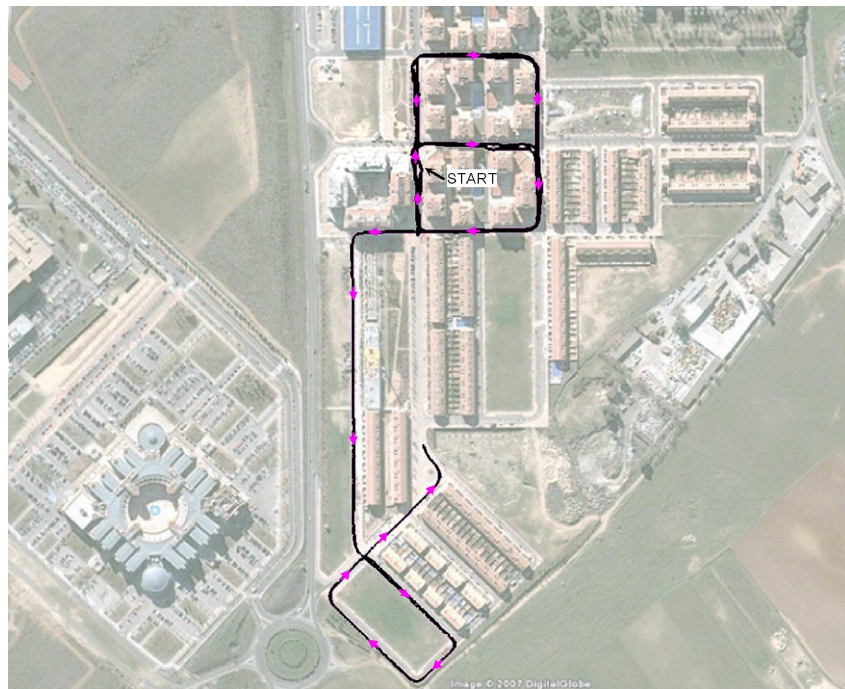


Figura 6.22: Vista aérea del recorrido de prueba C. El punto de inicio se indica en la figura.

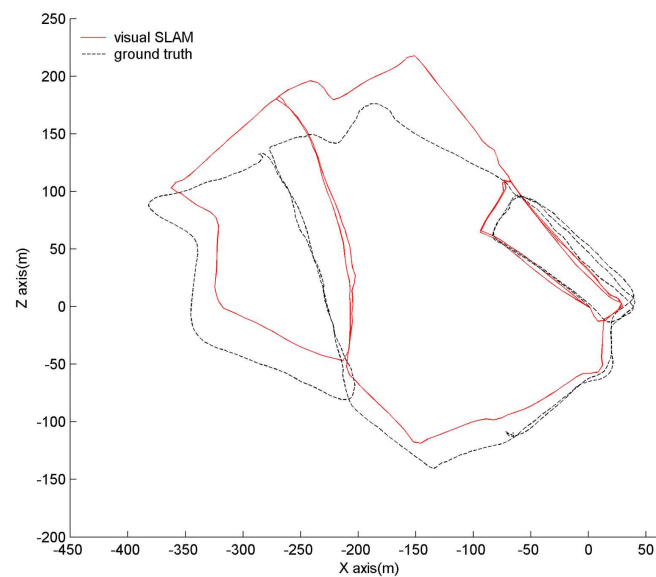
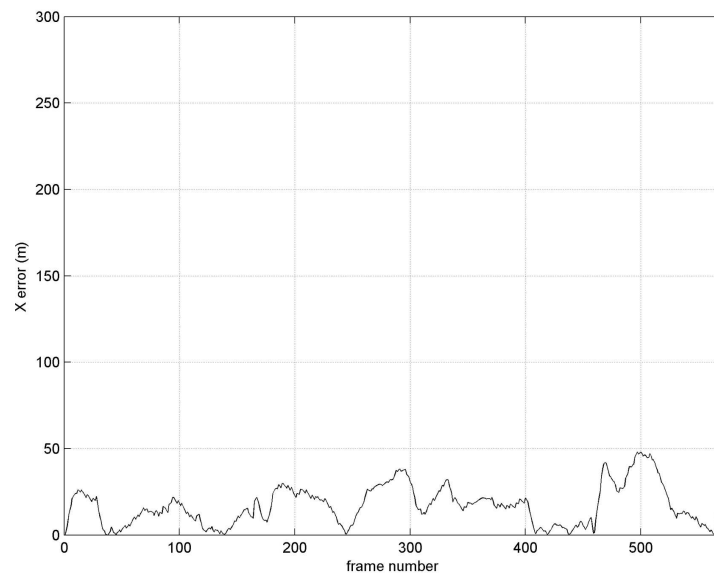
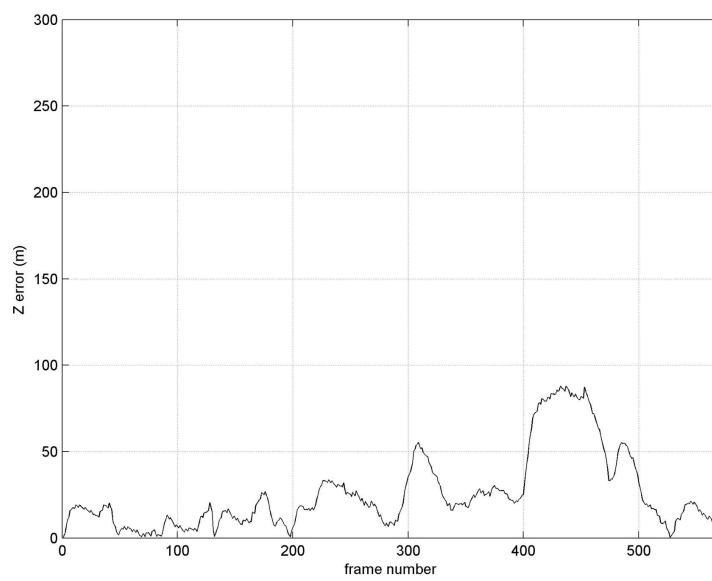


Figura 6.23: Estimación del recorrido B usando el método propuesto (rojo) y la referencia (línea rayada).



(a) Error en el eje X



(b) Error en el eje Z

Figura 6.24: Error en la estimación del recorrido B usando el sistema propuesto respecto a la referencia.

la reconstrucción frente al recorrido C.

En la Fig. 6.24(a) y la Fig. 6.24(b) se muestran los errores de la estimación frente a la referencia, tanto en el eje X como en el eje Z. En ambas gráficas se constata un error en la estimación de la posición menor que el obtenido para el recorrido C debido a las causas mencionadas.

En cuanto al recorrido C, en la Fig. 6.25 se muestra la estimación del sistema de SLAM visual comparada con la referencia. Se puede remarcar el error relativamente bajo en la parte inicial del recorrido estimado por el sistema SLAM, teniendo en cuenta que éste se ha basado únicamente en visión. Por otra parte, se puede observar la elevada longitud del tramo recto que une la parte superior con la parte inferior del recorrido (alrededor de 350 m). Unido a este hecho, el reducido número de marcas cercanas tomadas en este tramo, debido al entorno abierto sin edificios cercanos al recorrido, trae como consecuencia un error acumulado en la trayectoria apreciable. A pesar de ello, puede observarse que el error en la reconstrucción, una vez cerrado el lazo de la parte inferior, se mantiene bajo.

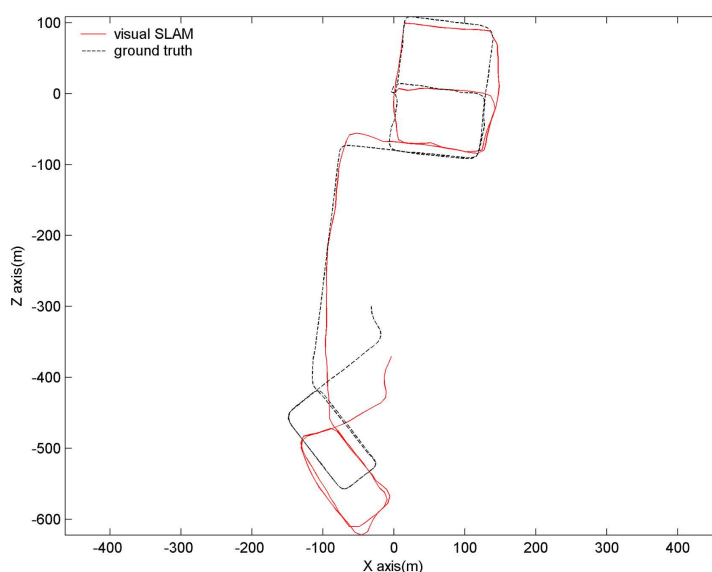
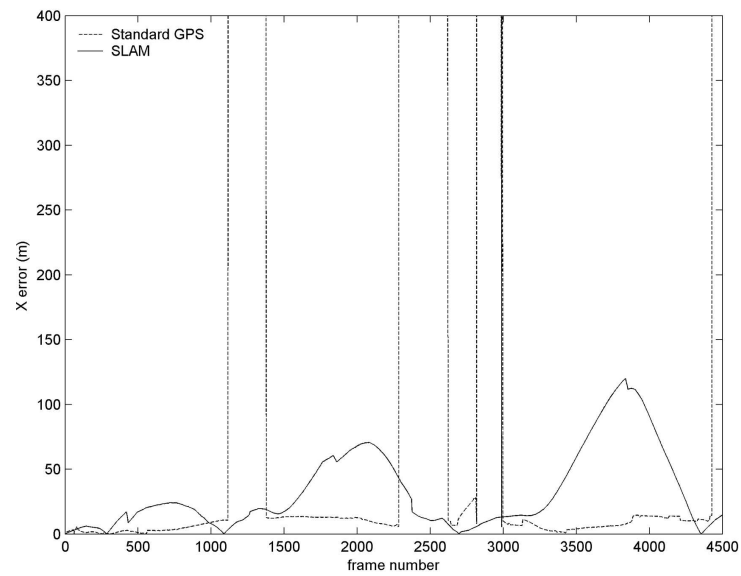


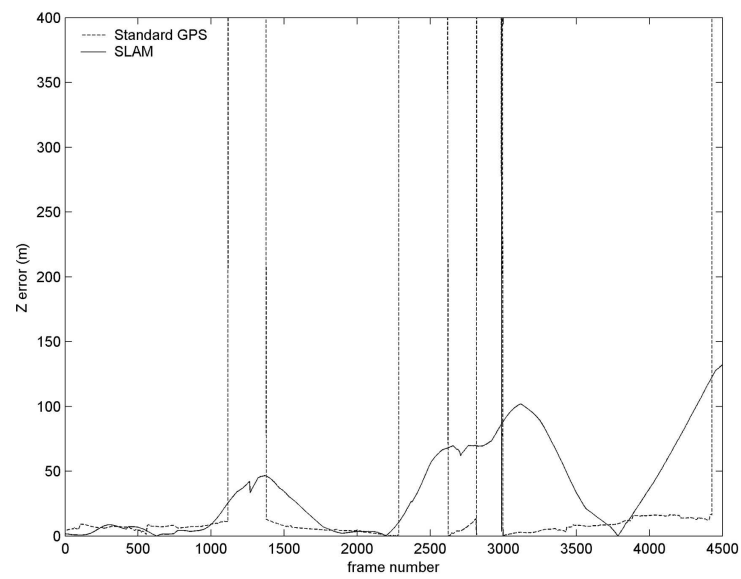
Figura 6.25: Estimación del recorrido C usando el método SLAM visual propuesto (rojo) y la referencia (línea rayada).

Se han calculado los errores en el eje X (ver Fig. 6.26(a)), en el eje Z (ver Fig. 6.26(b)), así como el error medio respecto a la referencia (ver Fig. 6.27). Éstos se comparan, además, con el error cometido en la estimación realizada únicamente con un GPS de bajo coste. La primera conclusión que se extrae es que el error medio usando SLAM visual no supera los 60 m en el peor de los casos. Obviamente, a medida que el vehículo recorre más distancia, el error usando sólo SLAM visual es mayor que usando un GPS. Sin embargo, si se observa el error del GPS, se aprecia que en ciertos tramos del camino el error es muy alto, dado que no hay datos disponibles, es decir, se perdió la cobertura. Este efecto es debido a los diferentes edificios altos ubicados en ese entorno, los cuales provocaron que la señal de satélite no fuera visible y como consecuencia el GPS fuera incapaz de proporcionar la localización del vehículo. Cuantos más edificios altos existan en el entorno, mayor probabilidad de pérdida de GPS existirá. Por lo tanto, en esos casos, incluso el error absoluto en la estimación del sistema es mucho más bajo que el del GPS.

Respecto al tiempo de procesamiento, en la Fig. 6.28 y en la Fig. 6.29 se muestran los tiempos de cómputo totales por frame para los recorridos B y C, respectivamente. Como se puede



(a) Error en el eje X



(b) Error en el eje Z

Figura 6.26: Error en la estimación del recorrido C usando un GPS estándar (línea rayada) y el sistema de SLAM (línea continua) respecto a la referencia.

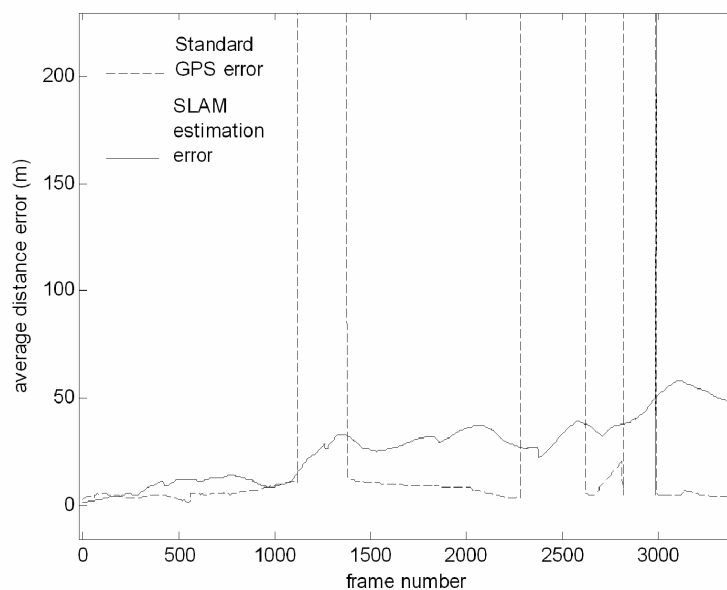


Figura 6.27: Error medio de estimación del recorrido C usando un GPS estándar (línea rayada) y el sistema de SLAM (línea continua) respecto a la referencia.

observar, el método propuesto sigue cumpliendo la restricción de tiempo real, manteniéndose el tiempo de procesamiento medio casi constante a lo largo del recorrido, incluso en situaciones de cierre de lazo. En la tabla 6.4, así como en la tabla 6.5 se observa que en las tareas del *low level SLAM*, la que mayor tiempo consume continúa siendo la inicialización de marcas, aunque éste se ha reducido notablemente gracias a la restricción de la longitud de la línea epipolar (ver Fig. 4.41). En cuanto al *high level SLAM*, el tiempo dedicado a la búsqueda de huellas SIFT previsitadas, así como a la corrección del mapa en el momento del cierre de lazo, es ligeramente superior al tiempo real. Sin embargo, como se explicó anteriormente, estas tareas pueden llevarse a cabo en paralelo a las tareas del bajo nivel, manteniendo el tiempo de cómputo por frame por debajo del tiempo real.

Low level SLAM processing times		High level SLAM processing times	
Number of features / frame	5 (minimum)	Path length	2.19 Km
		Number of loops	4
Filter step	Average Time	Average speed	30 Km/h
Measurements	3 ms	Number of features	8130
Filter update	5 ms	Number of fingerprints	230
Feature initializations	7 ms	Number of SIFT fingerprints	23
		Time	
		Fingerprint matches	3 s
		Loop closing	1 s

Tabla 6.4: Datos relevantes para el recorrido B.

A la vista de los resultados demostrados, se concluye que, tanto la ampliación del *baseline* de las cámaras como el uso del método MLR, mejoran sensiblemente las estimaciones obtenidas por el sistema. En el recorrido C se aprecia, sin embargo, que en los casos en los que no se

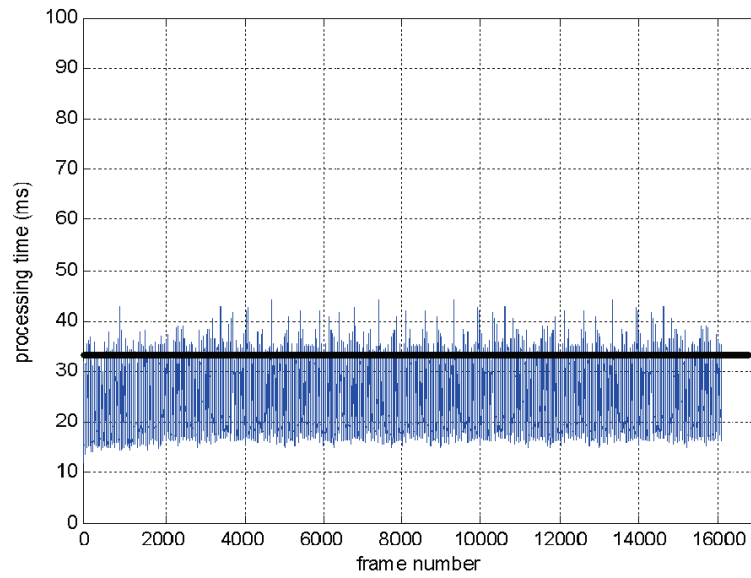


Figura 6.28: Tiempos de cómputo totales por frame para el recorrido B.

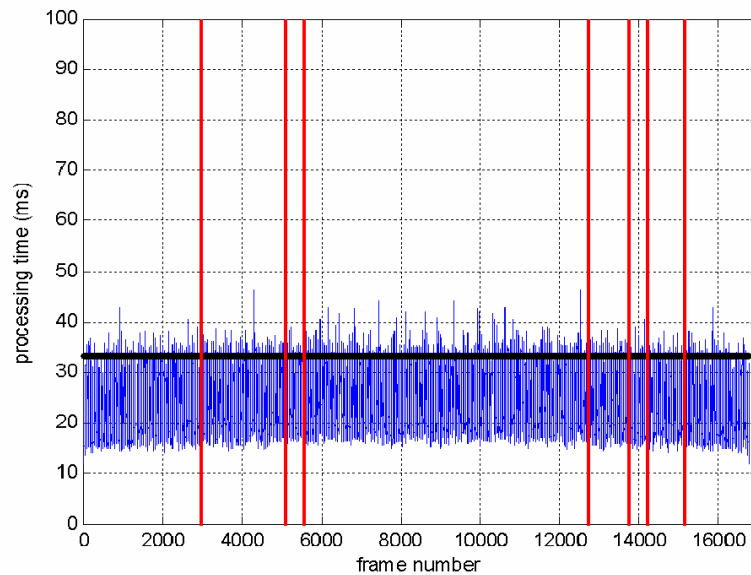


Figura 6.29: Tiempos de cómputo totales por frame para el recorrido C. El límite de tiempo real se representa como una línea constante en 33 ms. Los frames en los que se produjeron cierres de lazo se marcan con líneas verticales rojas.

Low level SLAM processing times		High level SLAM processing times	
Number of features / frame	5 (minimum)	Path length	2.27 Km
		Number of loops	3
Filter step	Average Time	Average speed	30 Km/h
Measurements	3 ms	Number of features	7250
Filter update	5 ms	Number of fingerprints	235
Feature initializations	7 ms	Number of SIFT fingerprints	18
		Time	
		Fingerprint matches	3 s
		Loop closing	1 s

Tabla 6.5: Datos relevantes para el recorrido C.

cierran grandes lazos y la disponibilidad de marcas visuales es reducida, se acumulan errores de estimación que no se corrigen completamente por el alto nivel de SLAM. Además, teniendo en cuenta los resultados, se deduce que combinando la estimación del SLAM visual con la información proporcionada por un GPS de bajo coste, se debería obtener una precisión similar a la de un GPS de alta calidad a un coste menor. En los siguientes experimentos, por tanto, se comprobará el funcionamiento del sistema, añadiendo el uso de un GPS de bajo coste.

- **Recorridos D, E y F. Fusión con GPS.**

En la siguiente serie de pruebas realizadas se ha empleado el sistema completo, tal como se muestra en la Fig. 6.7, fusionando así la información visual y del GPS. De estas pruebas, los recorridos de mayor interés se analizan a continuación.

El recorrido D es el que se muestra en la Fig. 6.30. Éste es parcialmente coincidente con el recorrido C presentado anteriormente. En este caso la velocidad media fue de unos 40 Km/h. El recorrido completo tiene 3.17 Km de longitud, tomándose 8520 marcas de bajo nivel y 281 huellas. En la Fig. 6.30 se aprecia la estimación real del camino recorrido por el vehículo, así como las marcas de bajo nivel capturadas sobrepuestas en una imagen real del entorno. Se puede apreciar que las áreas en las que se localizan edificios altos contienen un mayor número de marcas. Esto ayuda a una localización más precisa proporcionada por el sistema de visión. Por otra parte, como se apuntó anteriormente, en los espacios abiertos sin edificios altos no se proporciona información visual precisa, mientras que, en este caso, la señal de GPS tiene una mayor fiabilidad y proporciona una mejor estimación de localización. Esto prueba que ambos sensores son mutuamente complementarios, proporcionando buenas estimaciones en situaciones diferentes. Por lo tanto, combinándolos apropiadamente se pueden obtener mejores estimaciones que usándolos de forma separada.

En la Fig. 6.31 se muestra una vista en perspectiva de la misma estimación, donde se aprecian las marcas de bajo nivel distribuidas en todo el volumen.

Para evaluar el funcionamiento del sistema se han comparado los resultados que ofrece respecto a la estimación de referencia. Para ello se capturaron, además, las posiciones del vehículo obtenidas usando únicamente el GPS de bajo coste, para así analizarlas y compararlas con el sistema propuesto. La Fig. 6.32 muestra la estimación del sistema de SLAM combinado y del GPS estándar comparados con la referencia. La señal de GPS se perdió en diferentes puntos al comienzo del recorrido, tal como se muestra en la Fig. 6.32. En el tramo más largo de pérdida de GPS, se puede observar fácilmente el incremento en el error de estimación. Sin embargo, aún

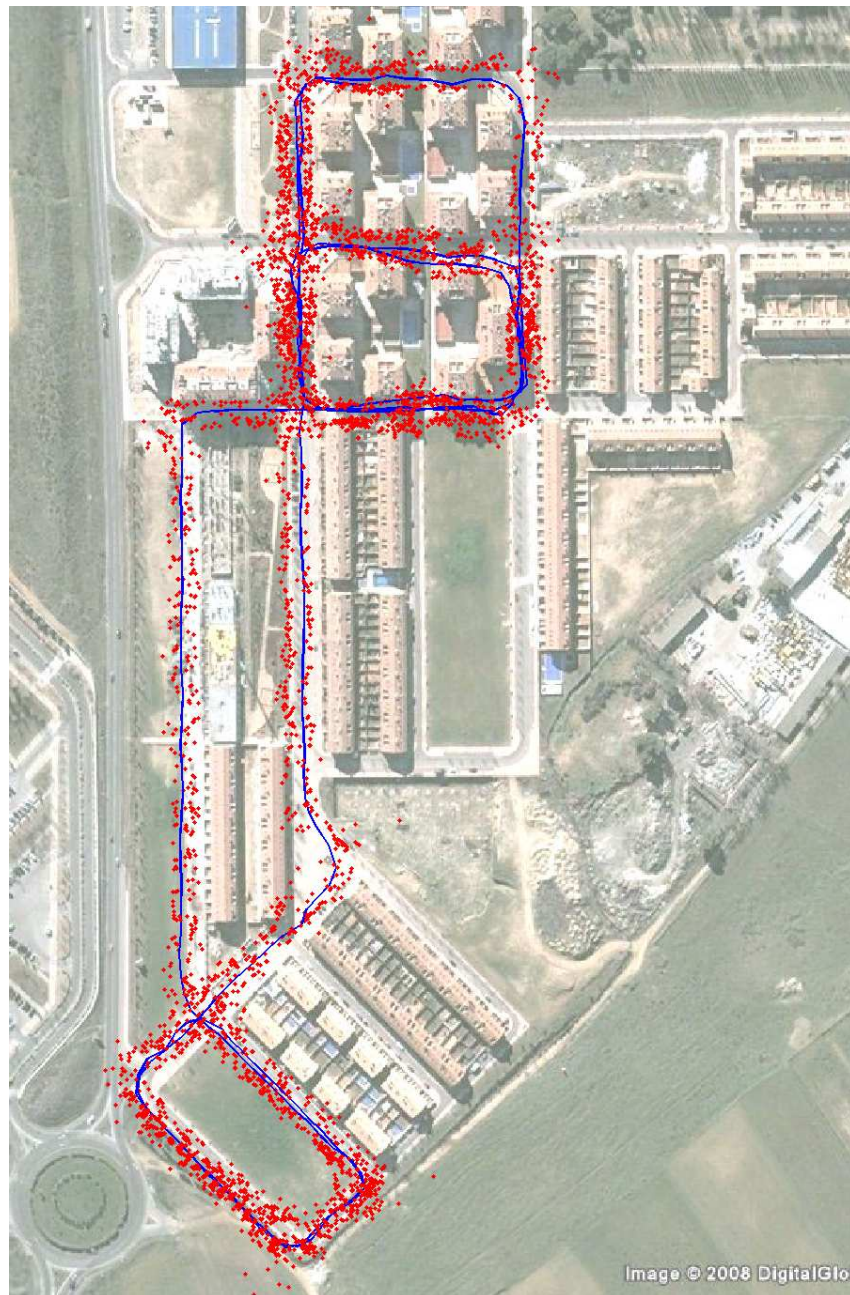


Figura 6.30: Vista aérea del camino recorrido por el vehículo. Se muestran las marcas de bajo nivel como puntos en rojo. El lazo que se muestra en la parte inferior del mapa contiene diversos edificios que aún no aparecían en el mismo.

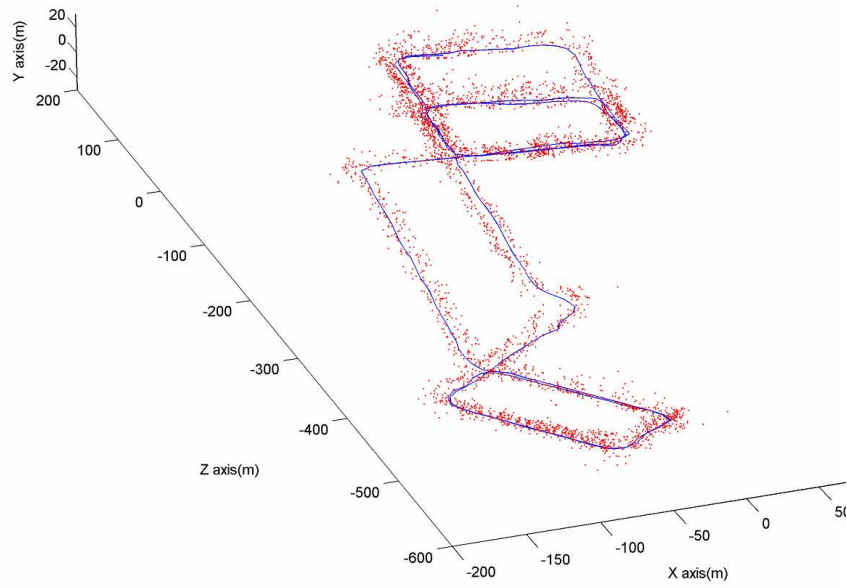


Figura 6.31: Vista en perspectiva del camino recorrido por el vehículo (línea continua). Las marcas de bajo nivel se muestran como puntos.

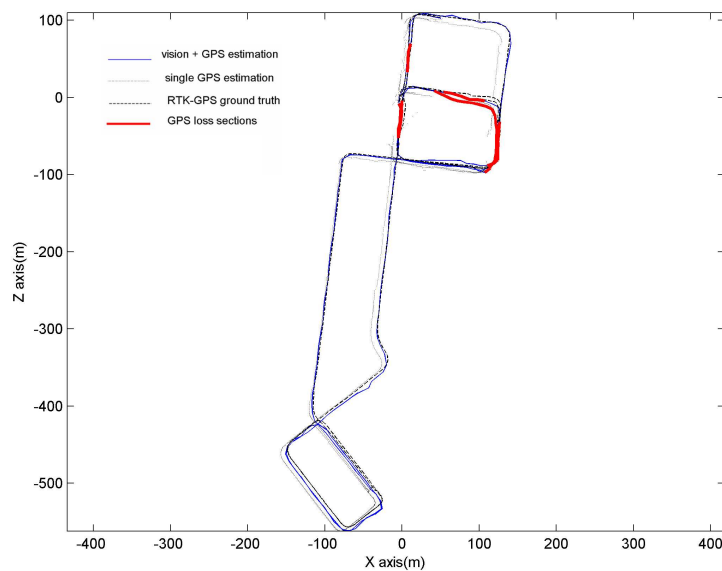
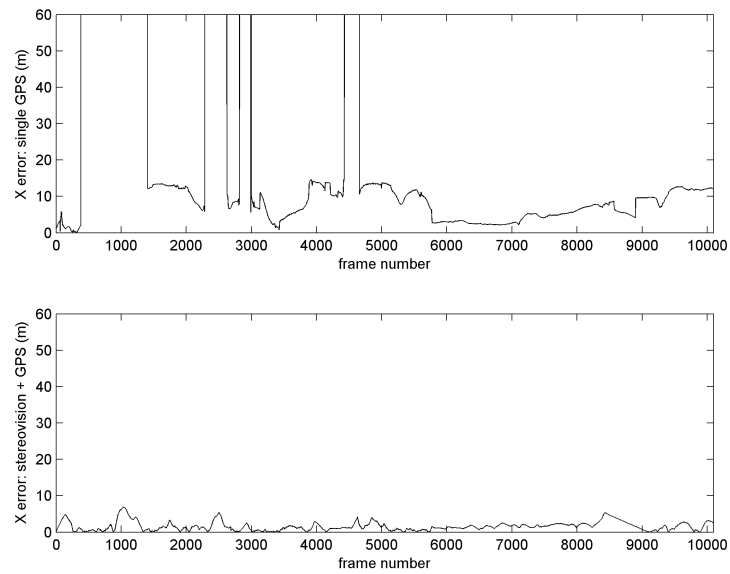
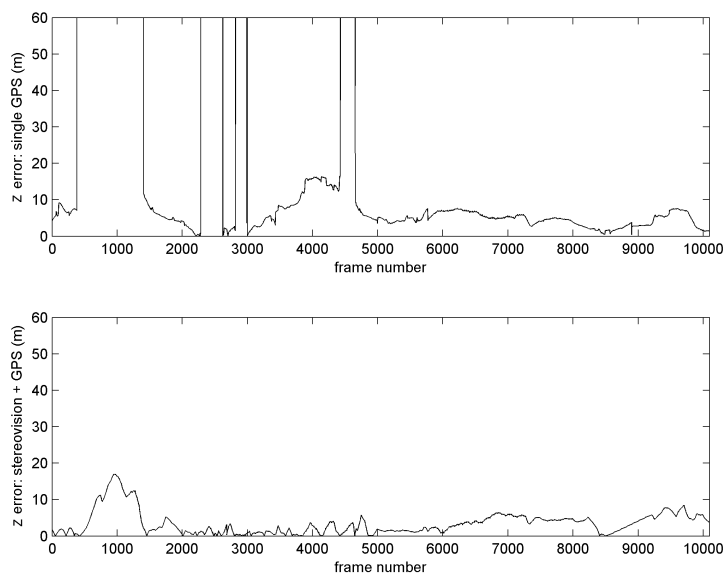


Figura 6.32: Estimación del recorrido D usando únicamente un GPS de bajo coste (línea punteada), sistema SLAM combinado (línea punto-rayada) y la referencia (línea rayada). Los tramos de pérdida de GPS se muestran en línea gruesa de color rojo.



(a) Error en el eje X



(b) Error en el eje Z

Figura 6.33: Error en la estimación del recorrido D usando un GPS estándar (arriba) y el sistema de SLAM combinado (abajo) respecto a la referencia.

se dispone de una estimación relativamente precisa para ser capaz de localizar el vehículo.

Se han calculado los errores en el eje X (ver Fig. 6.33(a)) y el eje Z (ver Fig. 6.33(b)) respecto a la referencia, usando únicamente el GPS estándar y el sistema de SLAM combinado. Como se puede observar, en los momentos de pérdida total de GPS, el error en el sistema presentado en esta Tesis se mantiene apreciablemente bajo. El tramo más largo con pérdida de GPS se encuentra al comienzo del recorrido. En ese periodo el sistema alcanzó el mayor error, que en el eje Z fue de aproximadamente 20 m. Sin embargo, incluso en ese momento, el error en el eje X se mantuvo muy bajo.

El recorrido E, emplea el método de SLAM combinado y se muestra en la Fig. 6.34. La velocidad media, como en el caso anterior, fue de unos 40 Km/h. El recorrido completo tiene una longitud de 1.45 Km, en el que se tomaron 3513 marcas de bajo nivel y 135 huellas. En dicha figura, se aprecia la estimación real del camino recorrido por el vehículo, así como las marcas de bajo nivel capturadas sobrepuestas en una imagen real del entorno.



Figura 6.34: Vista aérea del camino recorrido por el vehículo. Los puntos inicial y final de recorrido se indican con una estrella. Los tres tramos numerados carecían de cobertura GPS.

En la Fig. 6.35 se representa la estimación del sistema de SLAM presentado, comparándola con la referencia. En este caso, también se han capturado los datos de posición provenientes del GPS, de forma aislada. Éstos se comparan igualmente con la referencia en la mencionada figura. Las pérdidas de señal GPS se sucedieron en diferentes partes del recorrido. En este caso,

en la Fig. 6.35 las pérdidas 1 y 3 se debieron a pasos por túneles, mientras que la número 2 fue debida a la oclusión parcial de los edificios altos colindantes. En el caso de la pérdida 3, debido al largo tiempo sin cobertura GPS, además del sensor GPS funcional, el GPS de referencia tampoco obtuvo datos de posición. Por lo tanto, en el mencionado tramo la única estimación es la proporcionada por el sensor visual.

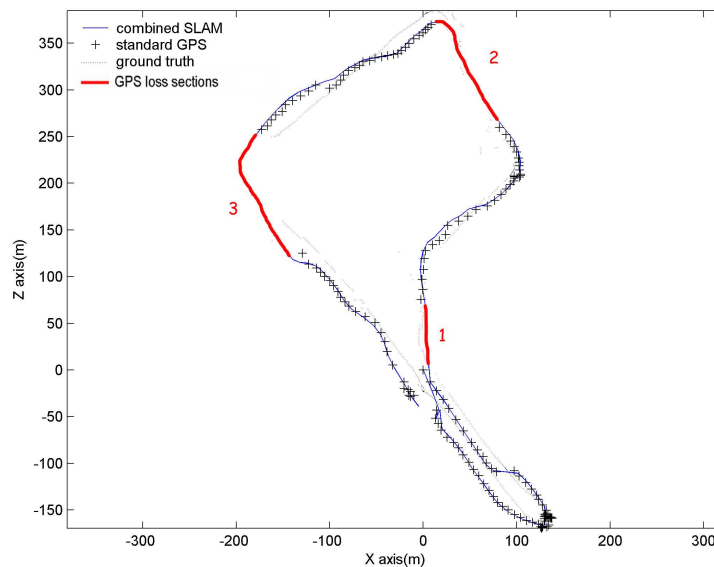


Figura 6.35: Estimación del recorrido E usando únicamente un GPS de bajo coste, el sistema SLAM combinado y la referencia. Los tramos de pérdida de GPS se muestran en línea gruesa de color rojo.

Igualmente, se han calculado los errores en el eje X (ver Fig. 6.36(a)) y el eje Z (ver Fig. 6.36(b)) respecto a la referencia, usando únicamente el GPS estándar y el sistema de SLAM presentado, apreciándose igualmente, que en los momentos de pérdida total de GPS, el error del sistema se mantiene considerablemente bajo. En el tramo explicado anteriormente no se pudo obtener datos de referencia, quedando así reflejado en las gráficas de las figuras 6.36(a) y 6.36(b).

Por último, el recorrido F, se muestra en la Fig. 6.37. La velocidad media, como en el caso anterior, fue de unos 40 Km/h. El recorrido completo tiene una longitud de 1.81 Km, en el que se tomaron 6117 marcas de bajo nivel y 195 huellas. En la Fig. 6.37 se aprecia la estimación real del camino recorrido por el vehículo, sobreimpresionado en una imagen real del entorno.

Siguiendo la metodología empleada hasta el momento, en la Fig. 6.38 se muestra tanto la estimación del sistema de SLAM combinado propuesto como la del GPS estándar funcionando de manera independiente, comparados con la referencia. La señal de GPS, en este caso, se perdió durante un tramo extremadamente largo, marcado como 2, debido a que las calles son muy estrechas. Se puede apreciar que al comienzo del tramo, el error introducido al perder la señal de GPS durante un giro provoca un gran error acumulado durante buena parte del recorrido. En la parte final del camino el error se reduce notablemente debido a la corrección del mapa, una vez recuperada la señal de GPS en el tramo final.

Los errores, tanto del sistema completo como del sensor GPS aislado, en el eje X respecto a la referencia se muestran en la Fig. 6.39(a), mientras que los asociados al eje Z se muestran en la Fig. 6.39(b). De forma análoga a los dos casos anteriores, en los momentos de pérdida total de GPS, el error en el sistema se mantiene apreciablemente bajo, a excepción de la parte final

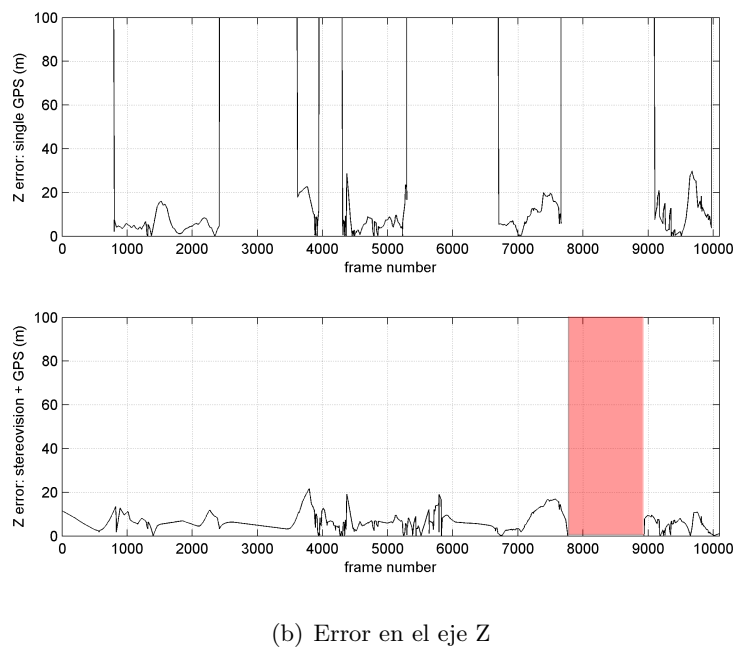
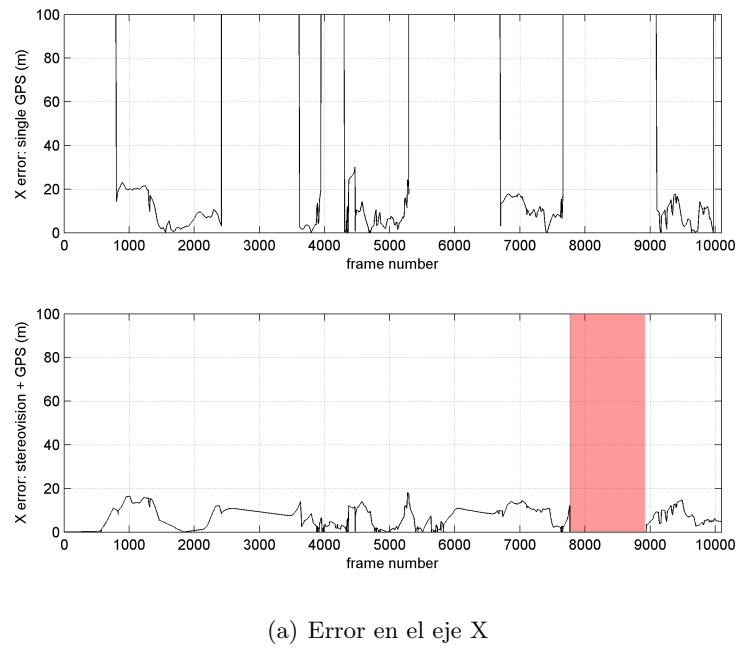


Figura 6.36: Error en la estimación del recorrido E usando un GPS estándar (arriba) y el sistema propuesto (abajo) respecto a la referencia. El tramo sin referencia se marca en color rojo.



Figura 6.37: Vista aérea del camino recorrido por el vehículo. El punto inicial de recorrido se indica con una estrella verde, mientras que el final se muestra con una estrella roja.

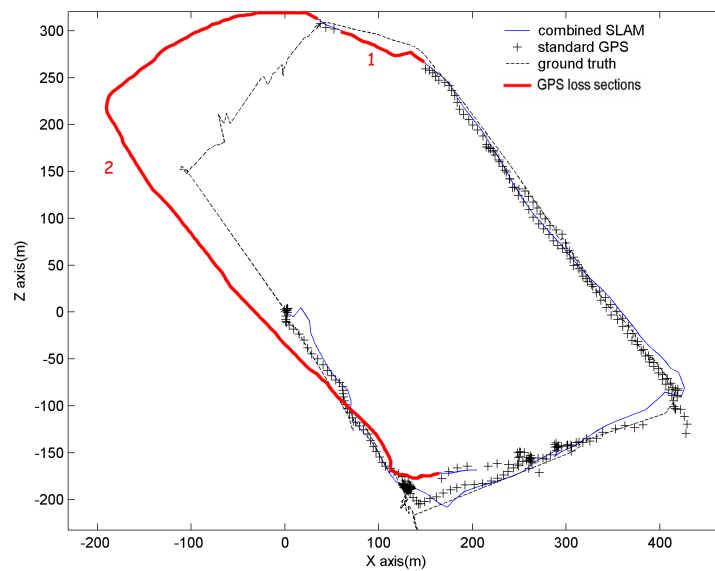


Figura 6.38: Estimación del recorrido F usando únicamente un GPS de bajo coste, el sistema SLAM combinado y la referencia. Los tramos de pérdida de GPS se muestran en línea gruesa de color rojo.

correspondiente al tramo 2 del recorrido.

En cuanto al tiempo de procesamiento, en la tablas 6.6, 6.7 y 6.8, se muestran los tiempos medios de cómputo para cada una de las tareas más importantes del proceso. Estos tiempos son comparables con los obtenidos en el caso del recorrido B. En este caso, se debe añadir el tiempo de cómputo asociado al procesamiento de la señal del GPS, el cual es de 4 ms. Sumando todos los tiempos del bajo nivel, el tiempo global sigue por debajo de la restricción de tiempo real. Más aún, esta tarea se ejecuta con una periodicidad no inferior a 1 s, por lo tanto el impacto final en el tiempo global es despreciable.

Low level SLAM processing times		High level SLAM processing times	
Number of features / frame	5 (minimum)	Path length	3.17 Km
		Number of loops	4
Filter step	Average Time	Average speed	40 Km/h
Measurements	3 ms	Number of features	8250
Filter update	5 ms	Number of fingerprints	281
Feature initializations	7 ms	Number of SIFT fingerprints	24
GPS processing (1 s sampling period)	4 ms		Time
		Fingerprint matches	3.5 s
		Loop closing	1 s

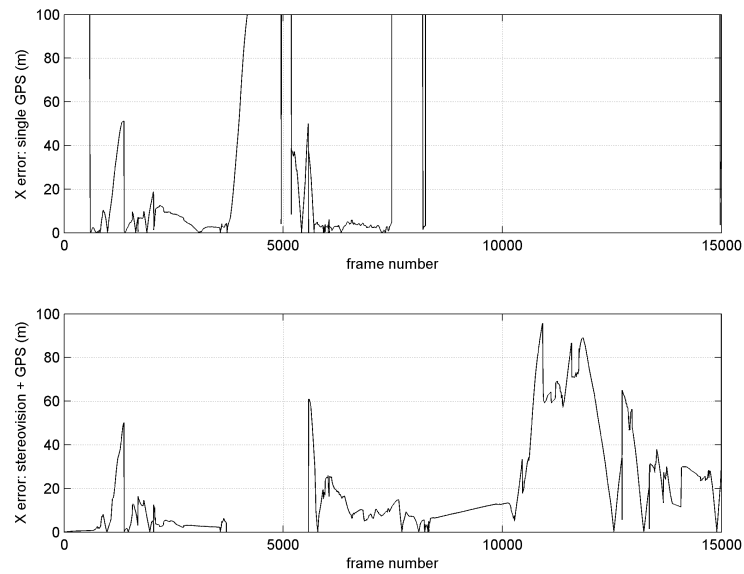
Tabla 6.6: Datos relevantes para el recorrido D.

Low level SLAM processing times		High level SLAM processing times	
Number of features / frame	5 (minimum)	Path length	1.45 Km
		Number of loops	1
Filter step	Average Time	Average speed	40 Km/h
Measurements	3 ms	Number of features	3513
Filter update	5 ms	Number of fingerprints	135
Feature initializations	7 ms	Number of SIFT fingerprints	9
GPS processing (1 s sampling period)	4 ms		Time
		Fingerprint matches	1.5 s
		Loop closing	500 ms

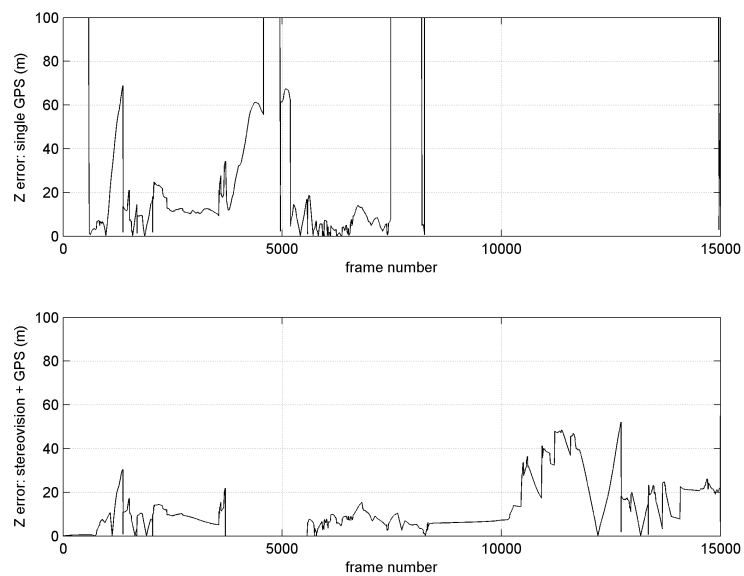
Tabla 6.7: Datos relevantes para el recorrido E.

Las conclusiones principales que se extraen de estas pruebas se centran en la clara mejora de las estimaciones, como consecuencia de la introducción del sensor GPS estándar al sistema. Esto se aprecia claramente en el recorrido D, donde además se extrae la conclusión adicional de que ambos sensores, visión y GPS, son complementarios, es decir, en los casos en los que la riqueza de marcas visuales es baja, generalmente la fiabilidad del GPS es alta, mientras que circulando en calles estrechas con edificios altos, la fiabilidad del GPS baja, aumentando por el contrario la riqueza y precisión de la estimación por parte del sensor visual. Los tiempos de cómputo, por otra parte, se mantienen dentro de los límites en todos los recorridos utilizados. Se concluye asimismo que el funcionamiento del sistema permite su utilización en aplicaciones de navegación autónoma, así como de asistencia a la navegación.

En la Tabla 6.9 se resumen los aspectos más relevantes de cada uno de los ensayos presentados anteriormente.



(a) Error en el eje X



(b) Error en el eje Z

Figura 6.39: Error en la estimación del recorrido F usando un GPS estándar (arriba) y el sistema de SLAM combinado (abajo) respecto a la referencia.

Low level SLAM processing times		High level SLAM processing times	
Number of features / frame	5 (minimum)	Path length	1.81 Km
		Number of loops	1
Filter step	Average Time	Average speed	40 Km/h
Measurements	3 ms	Number of features	6117
Filter update	5 ms	Number of fingerprints	195
Feature initializations	7 ms	Number of SIFT fingerprints	5
GPS processing (1 s sampling period)	4 ms		Time
		Fingerprint matches	2.0 s
		Loop closing	1 s

Tabla 6.8: Datos relevantes para el recorrido F.

Test path \ Results	Indoor path	Path A	Path B	Path C	Path D	Path E	Path F
Optimization method	Linear	Linear	MLR	MLR	MLR	MLR	MLR
Mean error	5 m	50 m	22 m	26 m	5 m	9 m	20 m
Mean processing time	29 ms	23 ms	24 ms	24 ms	25 ms	25 ms	25 ms
Covered distance	283.25 m	2965 m	2190 m	2270 m	3170 m	1450 m	1810 m
GPS	No	No	No	No	Yes	Yes	Yes
Fingerprints	-	150	230	235	281	135	195
SIFT fingerprints	8	9	23	18	24	9	5

Tabla 6.9: Resumen de los datos más relevantes para los diferentes recorridos presentados.

Por último, para evaluar la estabilidad del método de reconocimiento de lugares previamente visitados ante cambios de iluminación, se ha realizado el estudio comparativo que se muestra en la tabla 6.10.

Day times	Daylight morning	Daylight afternoon	At sunset	At night
Daylight morning	0 / 7.5	1 / 10	0 / 10	0 / 40
Daylight afternoon		1 / 7.5	0 / 12.5	0 / 32.5
At sunset			0 / 10	0 / 35
At night				0 / 20

Tabla 6.10: Robustez de la detección de lugares previsitados ante cambios de iluminación. En la tabla se indica: % Falsos positivos / % Falsos negativos.

Dicho estudio está centrado en el proceso de comparación de huellas SIFT. Se construyó una base de datos con 40 imágenes de 10 lugares diferentes, tomadas en diferentes momentos a lo largo de un día. Cada columna de la tabla representa un sub-grupo de huellas SIFT, clasificadas según dichos momentos. Estos sub-grupos se representan, a su vez, en las filas de la tabla, de forma que todas las huellas de la muestra se comparen entre sí una a una. En las intersecciones de cada fila/columna se muestran los resultados de las comparaciones entre los correspondientes grupos de huellas. Estos resultados son: el porcentaje de comparaciones positivas erróneas (falsos positivos) y el porcentaje de comparaciones positivas no detectadas (falsos negativos).

De los resultados se puede concluir que la probabilidad de un falso positivo es extremadamente baja, obteniéndose valores razonablemente buenos para los falsos negativos con luz de día. Por la noche los resultados para los falsos negativos empeoran, debido fundamentalmente a la reducción de zonas bien iluminadas.

6.4. Aplicación a la mejora de un navegador GPS

De cara a comprobar el funcionamiento del sistema de una forma práctica se ha diseñado una aplicación real para la mejora de un navegador GPS.

La mayoría de los sistemas de asistencia a la navegación comerciales se basan únicamente en la señal GPS. Como se explicó anteriormente, estos dispositivos suelen fallar en entornos en los que la cobertura de señal GPS desaparece, como sucede fundamentalmente en túneles, o se vea deteriorada por edificios altos, calles estrechas, etc. Como ha quedado demostrado, el sistema de SLAM combinado es capaz de mejorar los resultados de localización, a un bajo coste, proporcionados por un GPS estándar, incluso en las situaciones mencionadas anteriormente.

Para el desarrollo del sistema práctico se ha realizado el montaje de la Fig. 6.40. Éste consiste fundamentalmente en un PC portátil en el que se ejecuta la aplicación, es decir, los procesos de adquisición y tratamiento de imágenes, el proceso de captura de posición del GPS, así como el proceso de SLAM principal. El sistema estima el estado del vehículo a partir de los datos de la cámara estéreo y el GPS en tiempo real. Como salida el sistema proporciona una señal, en formato NMEA [76], conteniendo los parámetros requeridos para la planificación y visualización de la ruta por parte del navegador utilizado.

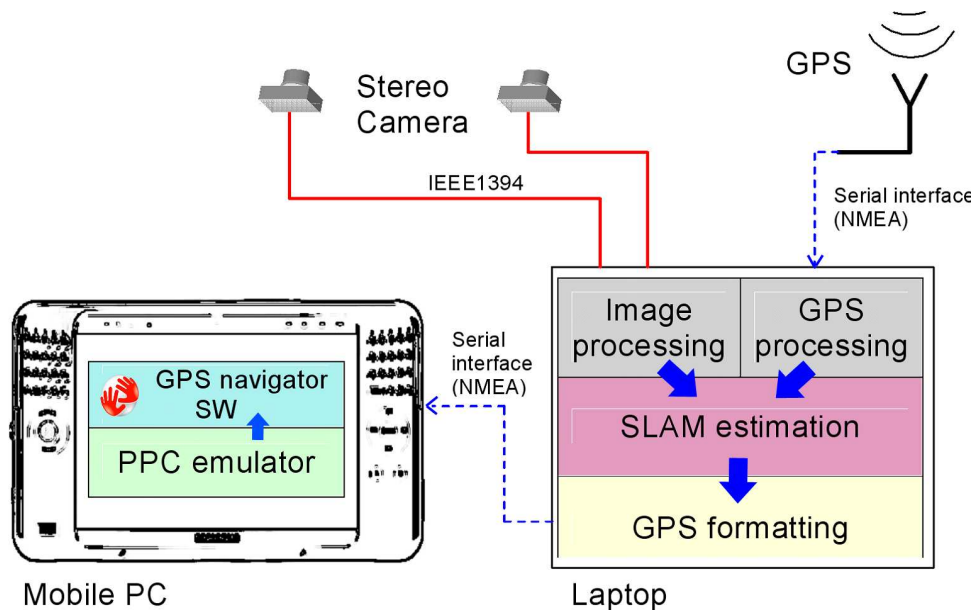


Figura 6.40: Diagrama funcional del sistema de asistencia a la navegación propuesto, basado en visión y GPS.

Estos datos son enviados por un enlace serie USB desde el PC portátil hasta el sistema de asistencia a la navegación. Como sistema de asistencia a la navegación, se ha utilizado un mobile PC al cual se ha instalado un emulador SW de Pocket PC. Sobre este emulador se ha instalado un conocido SW de navegación comercial. El modelo de mobile PC utilizado es un Samsung Q1 Ultra con procesador Intel McCaslin A110 (800MHz). Las comunicaciones con el PC portátil se

realizan a través de un puerto USB (ver Fig. 6.41).



Figura 6.41: Mobile PC empleado para emular el dispositivo navegador.

6.4.1. Resultados de las pruebas

Para probar el sistema práctico explicado se utilizó el recorrido D analizado en el apartado anterior. Además de los resultados de precisión en la reconstrucción del mapa, así como de tiempos de cómputo, ya analizados, en este caso se pretende comprobar cómo el sistema presentado es capaz de proporcionar información lo suficientemente precisa y constante para que el navegador sea capaz de presentar e indicar la ruta correcta en todo momento.

En la figura 6.34 se muestran tres tramos en los que la señal GPS no estaba disponible, y por lo tanto, un navegador estándar habría perdido la cobertura. En estos casos, dichos dispositivos suelen utilizar filtrados predictivos para estimar la posición del vehículo en base únicamente a la trayectoria y velocidad del vehículo hasta ese momento. Sin embargo, estos modelos no son capaces de proporcionar información útil de navegación cuando las pérdidas de señal son muy largas o cuando durante éstas se producen cambios notables de dirección o velocidad.

El primer tramo mencionado es un tramo corto y recto, por lo que un sistema convencional hubiera podido solventar la estimación con relativa precisión. En el caso del tramo dos (ver Fig. 6.42), éste es más largo que el anterior, existiendo además un giro al final de dicho tramo. En la Fig. 6.43 se muestra una secuencia del vídeo tomado a la pantalla del navegador en el momento del giro. En el inicio de la secuencia, mostrado en la Fig. 6.43(a), el vehículo se encontraba parado en un semáforo, de ahí la indicación de velocidad nula en la pantalla del navegador. En éste se aprecia también cómo la indicación de cobertura del GPS muestra que ésta es cero. Posteriormente, en la Fig. 6.43(b) se observa que el vehículo ha comenzado a moverse y está girando hacia la izquierda, todo ello sin cobertura del GPS. Por último, en la Fig. 6.43(c) se muestra cómo el vehículo vuelve a detenerse una vez completado el giro, debido a la existencia de otro semáforo, completándose correctamente la estimación del mencionado giro.

Por otra parte, el tramo 3 (ver Fig. 6.42) discurre por debajo de un túnel con una intersección. De igual modo que en el tramo 2, la señal de GPS no estaba presente en toda la secuencia. En

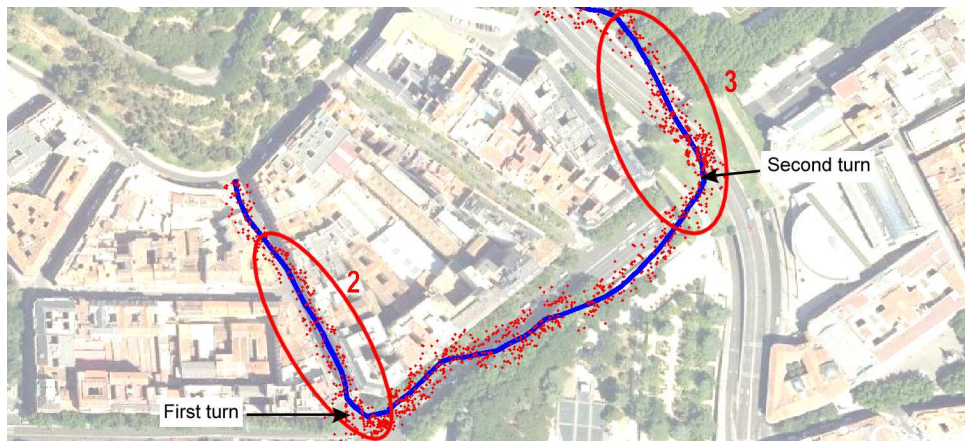


Figura 6.42: Detalle de los tramos 2 y 3 del recorrido D utilizado para las pruebas. Se indican los dos giros realizados por el vehículo en ausencia de señal de GPS

la Fig. 6.44 se muestra una secuencia de imágenes asociada al recorrido en el tramo 3. Al lado derecho de cada representación de la pantalla del navegador, se muestra una imagen del interior del vehículo y de la vía de circulación en el mismo instante. En la Fig. 6.44(a) se muestra el momento en el que el vehículo comienza la entrada en el túnel. En ese momento se observa cómo la señal de GPS vuelve a dejar de estar disponible. En la Fig. 6.44(b) se observa cómo el vehículo comienza a realizar el giro hacia la izquierda, siendo correctamente indicado en el navegador. En el final de la secuencia, mostrado en la Fig. 6.44(c), se observa cómo el vehículo ha terminado de realizar el giro y ha salido del túnel, siendo el itinerario correctamente indicado por el navegador. Se puede apreciar que a pesar de encontrarse el vehículo fuera del túnel, la señal de GPS aún no se ha recuperado, debido al tiempo de respuesta del GPS. Por otra parte, si se observa la indicación de velocidad en el panel de instrumentos del vehículo y se compara con la mostrada por el navegador en ese momento, se aprecia cómo la estimación realizada por el sistema se ajusta de manera fiel a la realidad.

La principal conclusión que se extrae es que con un navegador estándar el giro realizado en ambos tramos explicados, especialmente en el que discurre dentro del túnel no hubiera sido identificado, fallando la indicación de navegación en este tramo. Sin embargo, gracias al sistema de SLAM combinado presentado en esta Tesis, este giro es correctamente identificado. Por otra parte, con el sistema presentado, es posible disponer de la velocidad del vehículo con una mayor precisión y velocidad de refresco. Teniendo en cuenta lo habitual de este tipo de situaciones, la aplicación aporta una clara mejora en el funcionamiento de este tipo de sistemas.

6.5. Conclusiones y aportaciones

En cuanto al hardware requerido para el sistema, éste se ha descrito en el presente capítulo, especificando algunas particularidades en función del tamaño del entorno en el que se realizaron las pruebas. Como conclusión principal se destaca que los equipos y sensores utilizados son de bajo coste y fácilmente obtenibles, ya que se usan en gran cantidad de aplicaciones comerciales.

Los ensayos fundamentalmente consistieron en realizar recorridos con el vehículo tomando datos y analizándolos posteriormente. Los datos obtenidos en cada ensayo son: la estimación del recorrido realizado por el sistema comparada con la referencia, los errores en dicha estimación, así como los tiempos de cómputo por frame y por tarea realizada. Adicionalmente, se construye un mapa no denso para dar soporte a la localización. Las conclusiones más significativas, para



(a) Comienzo del giro.



(b) Durante el giro.



(c) Final del giro.

Figura 6.43: Secuencia de imágenes del display del navegador en el momento del giro del tramo 2. El orden temporal es de arriba a abajo. La posición y orientación del vehículo estimadas se muestran con una flecha azul.



(a) Comienzo del giro.



(b) Durante el giro.



(c) Final del giro.

Figura 6.44: Secuencia de imágenes del display del navegador en el momento del giro del tramo 3. A la derecha se muestran las imágenes del vehículo y su entorno correspondientes a las posiciones mostradas por el navegador (izquierda). El orden temporal es de arriba a abajo.

cada escenario analizado, son las siguientes:

- De los resultados obtenidos en interiores, la principal conclusión es el bajo error obtenido, incluso teniendo únicamente en cuenta la estimación realizada por el bajo nivel de SLAM.
- De los resultados obtenidos en el recorrido A, se puede concluir un funcionamiento correcto del sistema general en grandes entornos exteriores. Sin embargo se observa que, a pesar del aumento del *baseline* de la cámara respecto al utilizado en entornos interiores, éste no es suficiente para obtener una estimación de bajo nivel aceptable. Además, se concluye que el sistema de corrección lineal empleado no proporciona los resultados de corrección requeridos para el propósito de esta Tesis.
- En cuanto a los recorridos B y C, se concluye que, gracias a la ampliación del *baseline* y al uso del método de corrección MLR, las estimaciones del recorrido son notablemente mejores que en las pruebas anteriores. Sin embargo, se aprecia que en circunstancias en las que no existen suficientes marcas visuales, la estimación se deteriora sensiblemente con la distancia recorrida.
- Por último, de los recorridos D, E y F, se concluye que, gracias a la incorporación de las medidas de un GPS de bajo coste, las estimaciones mejoran notablemente, teniendo estas un error suficientemente bajo en la mayoría de las situaciones. Se concluye, además, que ambos sensores (visión y GPS) son complementarios entre sí, es decir, en los casos en los que la riqueza de marcas visuales es baja, generalmente la fiabilidad del GPS es alta y a la inversa.
- Como conclusión general, se observa que en cualquiera de las pruebas realizadas, el tiempo de cómputo total permanece siempre por debajo del límite de tiempo real establecido.
- Adicionalmente, se ha validado el sistema desarrollando una aplicación práctica de asistencia a la navegación.

Respecto a las aportaciones de este capítulo, se destacan las siguientes:

- La principal aportación es la validación de un sistema SLAM jerárquico, formado por un nivel métrico y otro topológico y basado en la fusión de un sensor de visión estéreo y un GPS de bajo coste, aplicado a la localización de vehículos sobre grandes entornos exteriores fundamentalmente urbanos. Dada la complementariedad de los sensores utilizados, el sistema propuesto es capaz de funcionar únicamente con el sensor de visión cuando no está disponible la señal GPS, mejorando así otras propuestas existentes en la literatura con un bajo coste. Adicionalmente, también se ha demostrado la utilidad del método, usando únicamente el sensor de visión, para grandes entornos interiores.
- Otra aportación destacable es la presentación de una aplicación práctica del sistema completo, dedicada a la mejora de un sistema de asistencia a la navegación comercial. De las pruebas realizadas se concluye que la fiabilidad de las planificaciones, cuando el vehículo circula por calles estrechas y edificios altos, mejora gracias a la aportación del sistema desarrollado en esta Tesis. Esta mejora es realmente considerable en situaciones en las que la cobertura del sensor GPS se ha perdido por completo (por ejemplo en túneles) y la planificación depende de que el vehículo se haya desviado por una u otra ruta.

Capítulo 7

Conclusiones y trabajos futuros

7.1. Conclusiones y trabajos futuros

En este apartado se resumen las principales conclusiones y aportaciones obtenidas en la Tesis, además de proponer las posibles futuras líneas de investigación derivadas de la misma.

7.2. Conclusiones generales y aportaciones

En la presente Tesis se ha propuesto un sistema de localización y mapeado de entornos de grandes dimensiones, destinado fundamentalmente a entornos urbanos. Para ello se hace uso de un sistema de visión estéreo de gran angular en combinación con un sensor GPS de bajo coste. Para llevar a cabo las tareas de localización, creación y gestión de los mapas, se ha desarrollado e implementado un método SLAM jerárquico de tipo Topológico-Métrico compuesto por dos niveles. El SLAM de bajo nivel se encarga de localizar y mapear el entorno localmente, haciendo uso de un filtro EKF y marcas visuales, además de los datos proporcionados por el GPS. El SLAM de alto nivel, por su parte, se encarga de gestionar y optimizar, utilizando el método MLR, un mapa topológico relacionado con los sub-mapas del bajo nivel. Además, se encargará de la detección robusta de lugares previamente visitados, empleando las denominadas Huellas SIFT. Se demuestra que el sistema completo es capaz de funcionar en tiempo real.

Las conclusiones generales y aportaciones de esta Tesis se resumen en las siguientes:

- En primer lugar se ha analizado la mejor alternativa de implementación del método de SLAM, llegando a la conclusión de que el método más adecuado para esta aplicación era un SLAM jerárquico de tipo Topológico-Métrico. La principal aportación del método desarrollado se centra, tanto en la estrategia robusta de asociación de datos basada en huellas SIFT como en una gestión de la relación entre nodos más simple y ajustada a los objetivos de la Tesis que otras propuestas existentes en la literatura.
- Se ha llevado a cabo un análisis exhaustivo del sensor visual estéreo para sacar el mayor partido del mismo, con el mínimo coste computacional. Además, se ha desarrollado un método de aplicación del filtro EKF, adaptado específicamente a la cámara estéreo utilizada. Se han realizado diversos estudios para determinar el tipo de marcas más apropiado para el sistema. También se ha implementado un modelo de movimiento adaptado específicamente a un vehículo que circula sobre un terreno esencialmente plano.
- La principal aportación del uso de un sensor estéreo radica en la posibilidad de estimar la escala real del entorno sin necesidad de un conocimiento previo del mismo. Además, el proceso de inicialización de marcas se realiza en un único paso, robusteciendo el funcionamiento del sistema. Asimismo, se ha desarrollado un método que agiliza la realización de correlaciones epipolares. También se aporta el desarrollo de un método de transformación de los *patches* asociados a las marcas para mejorar su eficiencia y el consiguiente aumento de precisión en la estimación. Por último, se propone un método de corrección de la distorsión de las cámaras adecuado para el gran campo de visión de las cámaras de gran angular utilizadas y que es más preciso que el propuesto por A. Davison.
- En cuanto al nivel alto de SLAM, se ha implementado un método de relación entre sub-mapas locales independientes, desarrollando métodos de cambio de sistemas de referencia probabilísticos para adecuar las transiciones entre los sub-mapas. Se han implementado métodos de detección de lugares previsitados, basados en las denominadas huellas SIFT, realizando diversos estudios de implementación de las necesarias correlaciones entre huellas. Asimismo, se han estudiado alternativas de implementación para la realización de la

optimización del mapa global, concluyendo en la elección del MLR como el método más adecuado. Por último, se han desarrollado estrategias para la integración de la información de GPS en ambos niveles jerárquicos, proporcionando medios para obtener una estimación de la orientación del vehículo, así como la optimización de tramos con pérdidas largas de señal.

- Una importante aportación es el desarrollo novedoso de la técnica de división en submapas EKF, combinada con un nivel superior de corrección basado en el método MLR, aportando una gran precisión en las estimaciones, a la par que una importante eficiencia en su implementación. Todo ello permite la estimación de mapas globalmente consistentes. Se proporcionan medios para estimar la incertidumbre global del vehículo, contribuyendo así a la eficiencia, en términos computacionales, de la detección de cierre de lazos. Asimismo, el desarrollo de las huellas SIFT constituye en sí mismo una aportación, debido a su gran especificidad y alta tasa de reconocimiento.
- Se muestran numerosos resultados de las pruebas de validación realizadas al sistema, tanto en entornos interiores como en exteriores de grandes dimensiones. El hardware de bajo coste utilizado para la implementación del sistema permite una posible futura implementación comercial. De los resultados obtenidos en las diferentes pruebas, se muestra una evolución en cuanto a las configuraciones idóneas de la cámara estéreo para adaptarla a grandes entornos, así como el método de optimización del mapa más adecuado. Los datos más relevantes extraídos en cada prueba son: la estimación del recorrido realizado por el vehículo comparada con una referencia precisa, los errores de la estimación así como los tiempos de cómputo por frame y por tarea realizada. Como conclusión final de los resultados se obtiene que, con la configuración de sensores y métodos más adecuada, los errores en la estimación, así como los tiempos de cómputo empleados se ajustan suficientemente a los requerimientos de las aplicaciones objetivo de esta Tesis.
- La aportación práctica más representativa de esta Tesis es la aplicación del sistema propuesto para la mejora de un sistema de asistencia a la navegación. De las pruebas realizadas se concluye que la fiabilidad de las planificaciones, cuando el vehículo circula por calles estrechas y edificios altos, es efectivamente mejorada gracias a la aportación del sistema presentado, demostrándose una mejoría mucho más evidente en situaciones en las que la cobertura del sensor GPS se ha perdido por completo (por ejemplo en túneles) y la planificación depende de que el vehículo se haya desviado por una u otra ruta.

7.3. Trabajos futuros

Una vez revisadas las principales conclusiones y aportaciones de la presente Tesis, a continuación se describen las líneas de investigación abiertas para posibles trabajos futuros.

a) *Adaptación del método MLR a 3D.*

Como se ha descrito anteriormente, el objetivo fundamental del trabajo desarrollado en esta Tesis es la localización y mapeado de un vehículo en grandes entornos, fundamentalmente urbanos, para sistemas de navegación autónoma o para mejorar sistemas de asistencia a la navegación. Por lo tanto, desde el punto de vista de la localización global, sólo se necesitará estrictamente la información 2D de la pose del vehículo. Por otra parte, desde el punto de vista del SLAM local, el uso de la posición 3D de las marcas proporciona una estimación de la pose del vehículo más precisa. El sistema se ha diseñado para entornos urbanos,

esencialmente planos, por lo tanto, la información de altura no es especialmente relevante, tanto para la detección de cierre de lazos como para la reconstrucción del mapa. Por otra parte, los resultados experimentales muestran unos mapas con una razonable precisión, con una implementación 2D en el alto nivel, cuando la desigualdad del terreno no es muy importante.

Sin embargo, para ampliar el espectro de posibles entornos adecuados para el sistema, se plantea la posibilidad de adaptar el método MLR para su funcionamiento en 3D. Para ello, las modificaciones esenciales del mismo pasarían por la utilización de jacobianos en las fases de linealización del método SLR, así como la agrupación de los bloques de la matriz dispersa en tamaños de 7×7 en lugar de los 3×3 utilizados. En cuanto a la optimización multinivel, se deberán realizar ciertas modificaciones en la definición del interpolador, ya que éste depende claramente del ángulo de rotación, que en 3D se define como un vector de cuatro componentes.

b) *Sustitución del método de adaptación de patches por métodos de distorsión afín.*

Una de las aportaciones de esta Tesis es el desarrollo de un método para modificar la apariencia de los patches correspondientes a las marcas. El objetivo de dicha modificación es que su apariencia sea lo más parecida a la que realmente tendría al modificarse el punto de vista del vehículo. Para ello, la técnica desarrollada se basa en una aproximación basada en la suposición de que los patches están orientados paralelamente a la dirección de avance del vehículo. Una solución más precisa es llevar a cabo una *distorsión afín* de los patches en función del punto de vista real del vehículo. En [77] se hace uso de este método, quedando como trabajo futuro su implementación en el sistema desarrollado en esta Tesis, así como su posterior evaluación.

c) *Utilización de los dos pares de componentes de las proyecciones de imagen como vector de medida.*

En el capítulo 4 dedicado al SLAM de bajo nivel, se describe la implementación del filtro EKF. Como vector de medida se justifica el uso de la posición 3D de cada marca, relativa a la cámara. El uso de este vector de tres componentes reduce el tamaño de la matriz de covarianza, reduciendo asimismo la complejidad del filtro y, por tanto, el tiempo de cómputo global. Como contrapartida, la linealidad del EKF queda deteriorada ya que la incertidumbre de dicho vector de medida no es completamente Gaussiana. Como trabajo futuro se plantea la implementación del filtro usando las cuatro componentes de los vectores de proyección, evaluando las ventajas y desventajas de dicha alternativa.

d) *Detección de objetos no estáticos.*

Como ya se ha descrito anteriormente en esta Tesis, un problema asociado al proceso de estimación de la pose del vehículo basándose en marcas del entorno, es la suposición de que éste sea estático. Dicha suposición no es completamente cierta debido a la aparición de diversos objetos móviles en el campo visual de la cámara, como son el tráfico rodado, peatones, árboles movidos por el viento, etc. A pesar de que el funcionamiento del proceso de selección de marcas es capaz de filtrar la mayoría de estas situaciones, en algunos casos el número de marcas incorrectamente asumidas como estáticas podría ser lo suficientemente alto como para provocar errores apreciables en la estimación. Para robustecer este proceso se propone la incorporación de un método de detección de objetos no estáticos del entorno, en la línea del método presentado en [42]

e) *Métodos alternativos de implementación para el SLAM de bajo nivel.*

Respecto al problema de estimación robusta de la trayectoria del vehículo sin tener en cuenta las correcciones debidas a los cierres de lazo, es posible hacer uso de técnicas alternativas a las usadas en el bajo nivel de esta Tesis, como son la *Odometría visual* y el *Bundle adjustment*. Como trabajo futuro se propone evaluar el funcionamiento del sistema sustituyendo el SLAM de bajo nivel, basado actualmente en un filtro EKF, por uno de los dos métodos alternativos mencionados anteriormente.

f) *Desarrollo de un sistema de navegación completo.*

Como continuación de la línea de investigación de esta Tesis, el siguiente paso a implementar será el desarrollo de un sistema de navegación autónomo, fundamentalmente en circuitos urbanos, preferiblemente cíclicos. Las tareas esenciales a implementar comprenderán, en un primer lugar, la navegación de bajo nivel (local). En ésta se implementarán sistemas de control reactivo, evitación de obstáculos, etc. Con este objetivo serán de utilidad las marcas visuales obtenidas en el SLAM de bajo nivel. Además se deberán implementar métodos de reconocimiento de marcas de carretera, como líneas, señalización, etc. Por encima de la navegación local deberán implementarse métodos de navegación global, que incluyan políticas de planificación utilizando procesos de decisión como los POMDPs (*Proceso de Decisión de Markov Parcialmente Observable*) y modelos de navegación basados en el mapa topológico del SLAM de alto nivel.

g) *Implantación Comercial.*

De cara a una posible implantación comercial, la aplicación candidata más inmediata es el sistema de asistencia a la navegación presentada en el capítulo de resultados 6.4. Los aspectos fundamentales a tener en cuenta se centran, por un lado, en el sistema de cámaras estéreo. Como se explicó anteriormente, este sistema se basa en las cámaras Fire-i de Unibrain. Para realizar un uso intensivo de las mismas en una aplicación embarcada en automóviles, el tiempo de vida debe ser lo suficientemente alto en comparación con los otros sistemas embarcados. Además, se deberá mejorar la integración de los sensores y del interfaz HMI en el habitáculo. En cuanto al sistema de procesamiento, éste se basa en el uso de un PC portátil de uso general. Para una mejor integración en el vehículo, así como una reducción del coste del sistema para su producción a gran escala, la implementación del mismo deberá realizarse en plataformas hardware diseñadas específicamente para la aplicación. Éstas, de forma general, podrían estar basadas en el uso de FPGAs, junto con drivers de protocolo IEEE1394 y USB. Por último, el sistema debería cumplir con todos los requerimientos de homologación en vehículos según la legislación vigente.

Capítulo 8

Bibliografía

Bibliografía

- [1] R. Smith, M. Self, and P. Cheeseman, “Estimating uncertain spatial relationships in robotics,” pp. 167–193, 1990.
- [2] H. Durrant-Whyte, “Uncertain geometry in robotics,” *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, vol. 4, pp. 851–856, Mar 1987.
- [3] H. Durrant-Whyte and T. Bailey, “Simultaneous localization and mapping: part i,” *Robotics and Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99–110, June 2006.
- [4] T. Bailey and H. Durrant-Whyte, “Simultaneous localization and mapping (slam): part ii,” *Robotics and Automation Magazine, IEEE*, vol. 13, no. 3, pp. 108–117, Sept. 2006.
- [5] S. J. Julier and J. K. Uhlmann, “A new extension of the kalman filter to nonlinear systems,” in *In Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, 1997, pp. 182–193.
- [6] A. J. Davison, “Mobile robot navigation using active vision,” Ph.D. dissertation, University of Oxford, 1998.
- [7] A. J. Davison, “Real-time simultaneous localisation and mapping with a single camera,” in *Proceedings of the 9th International Conference on Computer Vision*, 2003, pp. 1403–1410.
- [8] J. Folkesson, P. Jensfelt, and H. I. Christensen, “Vision slam in the measurement subspace,” in *In Proc. of the IEEE International Conference on Robotics and Automation (ICRA, 2005*, pp. 30–35.
- [9] P. Jensfelt, D. Kragic, J. Folkesson, and M. Björkman, “A framework for vision based bearing only 3D SLAM,” in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA’06)*, Orlando, FL, May 2006.
- [10] C. Harris and M. Stephens, “A combined corner and edge detection,” in *Proceedings of The Fourth Alvey Vision Conference*, 1988, pp. 147–151.
- [11] D. Rodriguez and F. Matia, “Integrating segments and edges in feature based slam,” in *Proc. of the IEEE International Conference on Advanced Robotics*, Coimbra, Portugal, 2003.
- [12] A. Doucet, “On sequential simulation-based methods for bayesian filtering,” Tech. Rep., 1998.
- [13] M. Montemerlo, “Fastslam: A factored solution to the simultaneous localization and mapping problem with unknown data association,” Ph.D. dissertation, Carnegie Mellon University, 2003.
- [14] T. Barfoot, “Online visual motion estimation using fastslam with sift features,” *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pp. 579–585, Aug. 2005.

- [15] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *In Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, 2003, pp. 1151–1156.
- [16] A. Gil, O. Reinoso, O. Martinez Mozos, C. Stachniss, and W. Burgard, "Improving data association in vision-based slam," *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 2076–2081, Oct. 2006.
- [17] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. of the International Conference on Computer Vision ICCV, Corfu*, 1999, pp. 1150–1157.
- [18] S. Se, D. Lowe, and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *International Journal of Robotics Research*, vol. 21, pp. 735–758, 2002.
- [19] R. Luke, J. Keller, M. Skubic, and S. Senger, "Acquiring and maintaining abstract landmark chunks for cognitive robot navigation," *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pp. 2566–2571, Aug. 2005.
- [20] F. Fraundorfer, C. Engels, and D. Nister, "Topological mapping, localization and navigation using image collections," *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp. 3872–3877, 29 2007–Nov. 2 2007.
- [21] M. Cummins and P. Newman, "Probabilistic appearance based navigation and loop closing," *Robotics and Automation, 2007 IEEE International Conference on*, pp. 2042–2048, April 2007.
- [22] E. Nerurkar and S. Roumeliotis, "Power-slam: A linear-complexity, consistent algorithm for slam," *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp. 636–643, 29 2007–Nov. 2 2007.
- [23] F. Dellaert and M. Kaess, "Square root sam: Simultaneous localization and mapping via square root information smoothing," *Int. J. Rob. Res.*, vol. 25, no. 12, pp. 1181–1203, 2006.
- [24] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *Int. J. Robotics Research*, vol. 21, p. 2002, 2002.
- [25] P. Piniés and J. Tardós, "Scalable slam building conditionally independent local maps," *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp. 3466–3471, 29 2007–Nov. 2 2007.
- [26] L. A. Clemente, A. J. Davison, I. D. Reid, J. Neira, and J. D. Tardós, "Mapping large loops with a single hand-held camera." in *Robotics: Science and Systems*, W. Burgard, O. Brock, and C. Stachniss, Eds. The MIT Press, 2007.
- [27] L. M. Paz, P. Piniés, J. D. Tardós, and J. Neira, "6 dof slam with stereo-in-hand," in *IEEE International Conference on Robotics and Automation*, 2008.
- [28] (2008, Nov.) Bumblebee. stereo vision camera systems. Point Grey Research. [Online]. Available: http://www.ptgrey.com/products/bumblebee2/bumblebee2_xb3_datasheet.pdf
- [29] J. Leonard and H. Feder, "Decoupled stochastic mapping," Massachusetts Institute of Technology, Marine Robotics Laboratory, Tech. Rep., 1999.
- [30] K. Chong and L. Kleeman, "Large scale sonarray mapping using multiple connected local maps," in *International Conference on Field and Service Robotics*, 1997, pp. 538–545.

- [31] S. B. Williams, “Efficient solutions to autonomous mapping and navigation problems,” Ph.D. dissertation, University of Sydney, Australian Centre for Field Robotics, 2001.
- [32] T. Bailey, “Mobile Robot Localisation and Mapping in Extensive Outdoor Environments,” Ph.D. dissertation, ACFR - University of Sydney, August 2002.
- [33] M. Bosse, P. Newman, J. Leonard, and S. Teller, “An atlas framework for scalable mapping,” in *in IEEE International Conference on Robotics and Automation*, 2003, pp. 1899–1906.
- [34] U. Frese, “Treemap: An $o(\log n)$ algorithm for indoor simultaneous localization and mapping,” *Autonomous Robots*, vol. 21, pp. 103–122, 2006.
- [35] U. Frese, “Closing a million-landmarks loop,” in *In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing. submitted*, 2006, pp. 5032–5039.
- [36] E. Eade and T. Drummond, “Monocular slam as a graph of coalesced observations,” in *Proc. 11th IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007.
- [37] B. Steder, G. Grisetti, S. Grzonka, C. Stachniss, A. Rottmann, and W. Burgard, “Learning maps in 3d using attitude and noisy vision sensors,” in *iros*, San Diego, CA, USA, 2007.
- [38] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in *European Conference on Computer Vision*, vol. 1. Springer, 2006, pp. 404–417.
- [39] O. Chum and J. Matas, “Matching with prosac - progressive sample consensus,” in *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 220–226.
- [40] H. Andreasson, T. Duckett, and A. Lilienthal, “Mini-slam: Minimalistic visual slam in large-scale environments based on a new interpretation of image similarity.” in *ICRA*. IEEE, 2007, pp. 4096–4101.
- [41] U. Frese, P. Larsson, and T. Duckett, “A multilevel relaxation algorithm for simultaneous localization and mapping,” *Robotics, IEEE Transactions on*, vol. 21, no. 2, pp. 196–207, April 2005.
- [42] A. I. Comport, E. Malis, and P. Rives, “Accurate quadrifocal tracking for robust 3d visual odometry.” in *ICRA*. IEEE, 2007, pp. 40–45.
- [43] M. J. Milford and G. F. Wyeth, “Single camera vision-only slam on a suburban road network,” in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, 2008, pp. 3684–3689.
- [44] M. Milford, G. Wyeth, and D. Prasser, “Ratslam on the edge: Revealing a coherent representation from an overloaded rat brain,” *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 4060–4065, Oct. 2006.
- [45] M. Agrawal and K. Konolige, “Real-time localization in outdoor environments using stereo vision and inexpensive gps,” in *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 1063–1068.
- [46] C. Engels, H. Stewenius, and D. Nister, “Bundle adjustment rules,” in *PCV06*, 2006.

- [47] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," pp. 726–740, 1987.
- [48] R. M. Haralick, C. Lee, K. Ottenberg, and M. Nölle, "Review and analysis of solutions of the three point perspective pose estimation problem," *Int. J. Comput. Vision*, vol. 13, no. 3, pp. 331–356, 1994.
- [49] E. Royer, M. Lhuillier, M. Dhome, and T. Chateau, "Localization in urban environments: monocular vision compared to a differential gps sensor," *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, pp. 114–121 vol. 2, June 2005.
- [50] E. Mouragnon, F. Dekeyser, P. Sayd, M. Lhuillier, and M. Dhome, "Real time localization and 3d reconstruction," *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, pp. 363–370, June 2006.
- [51] K. Ohno, T. Tsubouchi, and S. Yuta, "Outdoor map building based on odometry and rtk-gps positioning fusion." in *ICRA*. IEEE, 2004, pp. 684–690.
- [52] R. Toledo-Moreo, M. Zamora-Izquierdo, B. Ubeda-Miarro, and A. Gomez-Skarmeta, "High-integrity imm-ekf-based road vehicle navigation with low-cost gps/sbas/ins," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 3, pp. 491–511, Sept. 2007.
- [53] OXTS, *Inertial+*. [Online]. Available: <http://www.oxts.co.uk>
- [54] G. Olague and R. Ramírez, "Síntesis de imágenes a partir de fotografías," in *DYNA 133*, 2001.
- [55] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proc. of IEEE Computer Vision and Pattern Recognition*, 1997, pp. 1106–1112.
- [56] O. D. Faugeras and G. Toscani, "Camera calibration for 3d computer vision," in *Proc. International Workshop on Industrial Applications of Machine Vision and Machine Intelligence*, 1987, pp. 240–247.
- [57] J. Weng, P. Cohen, and M. Herniou, "Camera calibration with distortion models and accuracy evaluation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992, pp. 965–980.
- [58] G. Q. Wei and S. D. Ma, "A complete two-plane camera calibration method and experimental comparisons," in *Proc. 4th International Conference on Computer Vision, Berlin*, 1993, pp. 439–446.
- [59] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," in *IEEE Journal of Robotics and Automation*, 1987, pp. 323–344.
- [60] Matlab. (2007) Camera calibration toolbox for matlab. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/
- [61] A. J. Davison, A. G. Cid, and N. Kita, "Real-time 3d slam with wide-angle vision," in *in Proc. IFAC Symposium on Intelligent Autonomous Vehicles*, 2004.

- [62] D. Fernández, “Sistema de detección de peatones mediante visión estereoscópica para la asistencia a la conducción,” Ph.D. dissertation, Departamento de Electrónica, University of Alcalá, 2008.
- [63] A. M. Martinez and A. C. Kak, “Pca versus lda,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 2, pp. 228–233, 2001.
- [64] A. K. Jain, R. P. W. Duin, and J. Mao, “Statistical pattern recognition: A review,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.
- [65] C. K. Chui, *An introduction to wavelets*. San Diego, CA, USA: Academic Press Professional, Inc., 1992.
- [66] M. Pontil and A. Verri, “Support vector machines for 3d object recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 6, pp. 637–646, 1998.
- [67] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.
- [68] S. Se, D. Lowe, and J. Little, “Vision-based mobile robot localization and mapping using scale-invariant features,” in *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA, 2001)*, pp. 2051–2058.
- [69] P. Zarchan and H. Musoff, Eds., *Fundamentals of Kalman Filtering: A Practical Approach. Progress in Astronautics and Aeronautics*. American Institute of Aeronautics and Astronautics, Inc., 2000, vol. 190.
- [70] J. Shi and C. Tomasi, “Good features to track,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR’94)*, Seattle, June 1994.
- [71] H. Choset and K. Nagatani, “Topological simultaneous localization and mapping (slam): Toward exact localization without explicit localization,” 2001.
- [72] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in FORTRAN; The Art of Scientific Computing*. New York, NY, USA: Cambridge University Press, 1993.
- [73] D. of the Army, *NAVSTAR Global Positioning System Surveying Engineering Manual*. US Army Corps of Engineers, 2003.
- [74] A. Yasuda, *Satellite navigation system, GPS*. Advanced Topics for Marine Technology, 2005.
- [75] A. Stroupe, M. Martin, and T. Balch, “Distributed sensor fusion for object position estimation by multi-robot systems,” *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 2, pp. 1092–1098 vol.2, 2001.
- [76] M. Corporation, *Data Transmission Protocol Specification for Magellan Products*, 1999.
- [77] B. Williams, G. Klein, and I. Reid, “Real-time slam relocation,” in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1–8.

