

Navegación Autónoma de Robot Móvil en Ambientes Parcialmente Estructurados utilizando Visión Artificial

Defensa de Tesis para obtener el grado de Doctor en Ingeniería

Gonzalo Fernando Perez Paina

Director: Dr. Eduardo Destéfanis

Codirectores: Dr. Miguel Ré y Dr. Julián Pucheta

Universidad Tecnológica Nacional, Regional Córdoba
Centro de Investigación en Informática para la Ingeniería



Córdoba, abril de 2015

- 1 **Introducción**
 - Robots móviles vs. manipuladores. Sensores
 - Problemas de la robótica móvil
 - Motivación y aportes
- 2 Localización y Mapeo Simultáneos - SLAM
- 3 Robot Móvil RoMAA-II y visión robótica
- 4 SLAM con mapeo bidimensional
- 5 SLAM con mapeo tridimensional
- 6 Conclusiones y trabajos futuros

Robots móviles vs. manipuladores

Robot móvil



Robot manipulador



- Dispositivos de locomoción
 - Aplicación en espacio de gran escala (movilidad)
 - Debe ser capaz de *navegar* de forma *inteligente*
- Entorno de trabajo limitado
 - Importantes avances en controles (industria)

Robótica móvil

AGV/UGV
(terrestres)

Automated Guided Vehicle
Autonomous Ground Vehicle
Unmanned Ground Vehicle
(3DOF o 6DOF)

UAV
(aéreos)

Unmanned Aerial Vehicle
(6DOF)

AUV
(acuáticos)

Autonomous Underwater Vehicle
(6DOF)



Robótica móvil

AGV/UGV
(terrestres)

Automated Guided Vehicle
Autonomous Ground Vehicle
Unmanned Ground Vehicle
(3DOF o 6DOF)

UAV
(aéreos)

Unmanned Aerial Vehicle
(6DOF)

AUV
(acuáticos)

Autonomous Underwater Vehicle
(6DOF)

- Teleoperado
- Semi-autónomo
- Autónomo



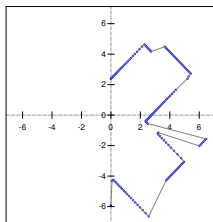
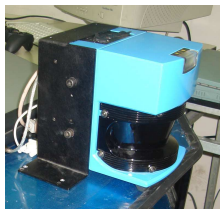
Sensores en robots móviles

Barrido láser

Cámaras

Sensores en robots móviles

Barrido láser

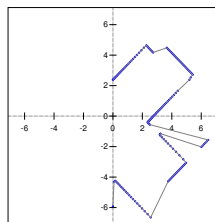
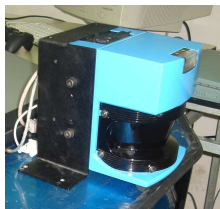


Cámaras

- Distancia y dirección (range&bearing)

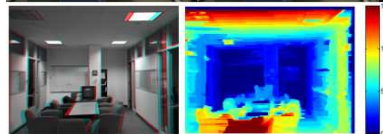
Sensores en robots móviles

Barrido láser



- Distancia y dirección (range&bearing)

Cámaras

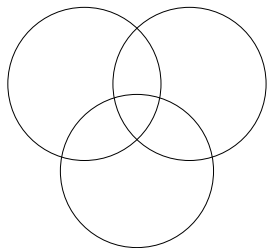
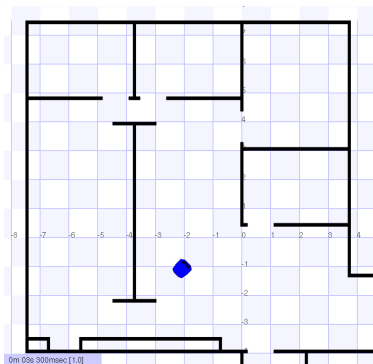


- Par estéreo: Distancia y dirección
- Monocular: solo dirección (bearing-only)

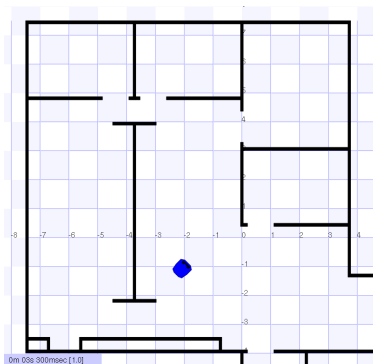
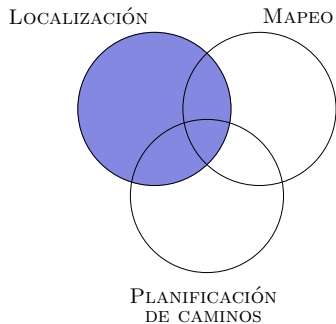
Problemas de la robótica móvil

LOCALIZACIÓN

MAPEO

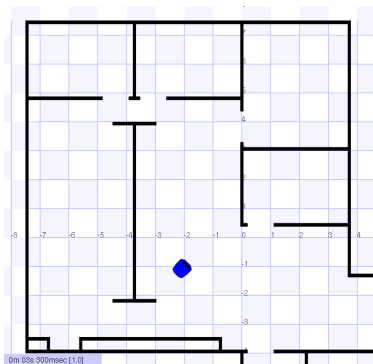
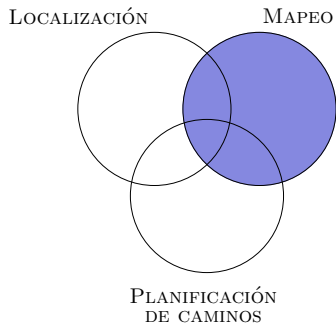
PLANIFICACIÓN
DE CAMINOS

Problemas de la robótica móvil



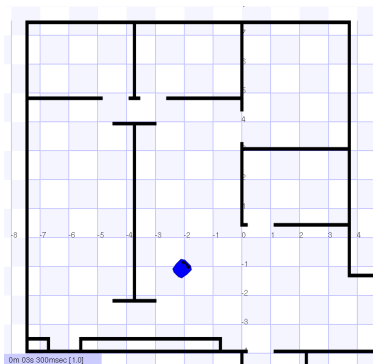
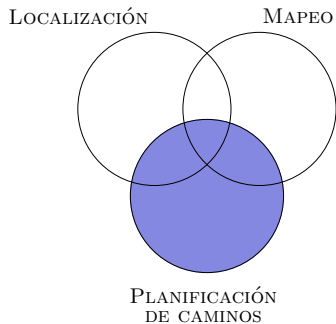
- **Localización**
- Mapeo
- Planificación

Problemas de la robótica móvil



- Localización
- Mapeo
- Planificación

Problemas de la robótica móvil

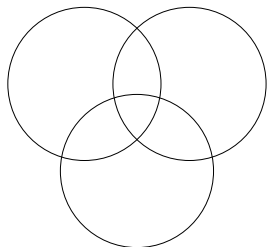
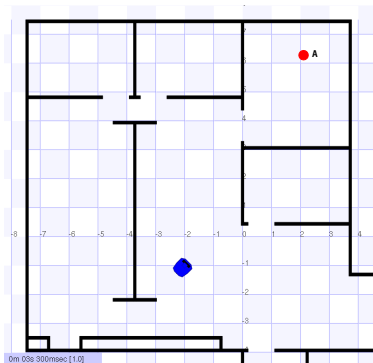


- Localización
- Mapeo
- Planificación

Problemas de la robótica móvil

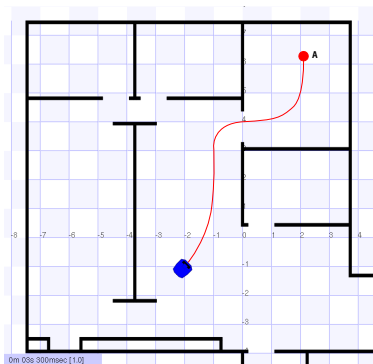
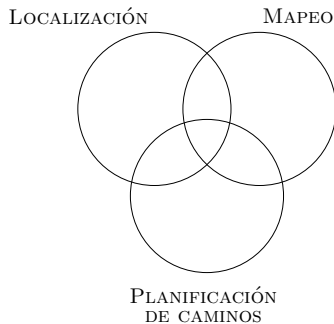
LOCALIZACIÓN

MAPEO

PLANIFICACIÓN
DE CAMINOS

- Localización
- Mapeo
- Planificación

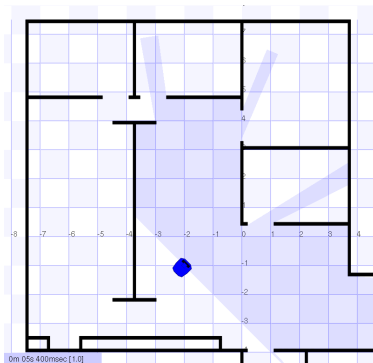
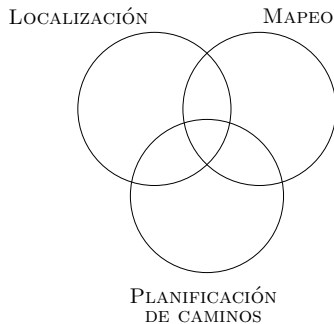
Problemas de la robótica móvil



- Localización
- Mapeo
- Planificación

Se necesita conocer la localización y el mapa

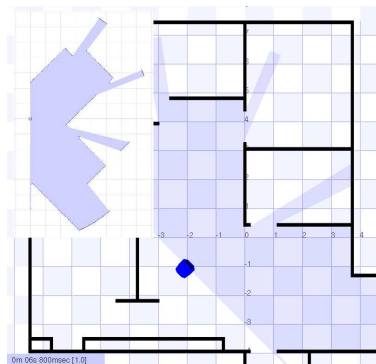
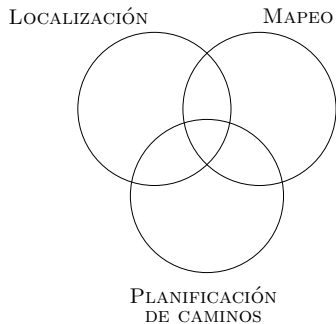
Problemas de la robótica móvil



- Localización
- Mapeo
- Planificación

Localización con mapa conocido

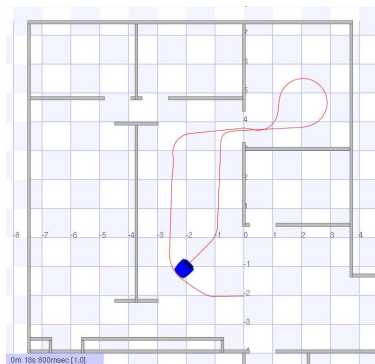
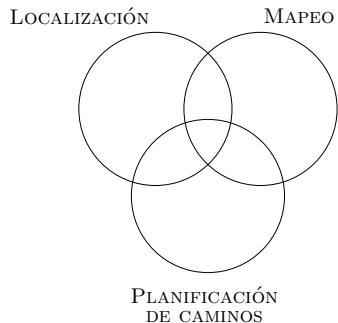
Problemas de la robótica móvil



- **Localización**
- Mapeo
- Planificación

Localización con mapa conocido

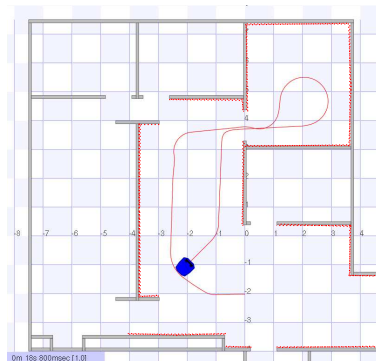
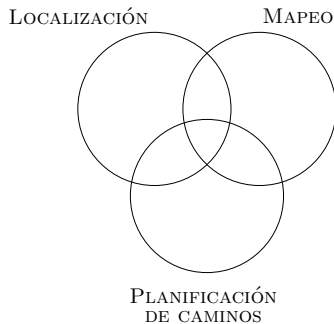
Problemas de la robótica móvil



- Localización
- Mapeo
- Planificación

Mapeo con localización conocida

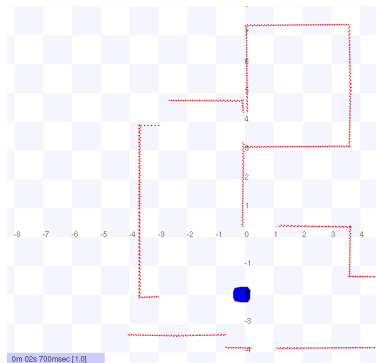
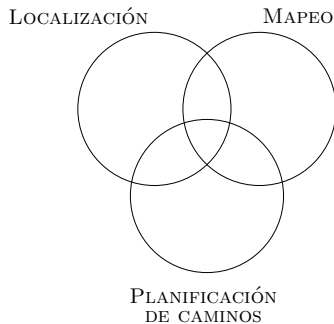
Problemas de la robótica móvil



- Localización
- Mapeo
- Planificación

Mapeo con localización conocida

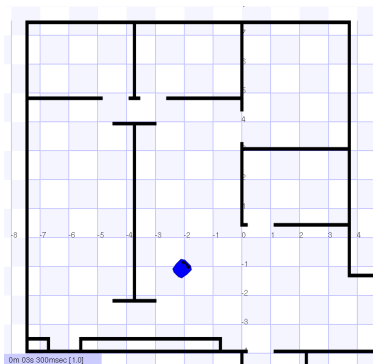
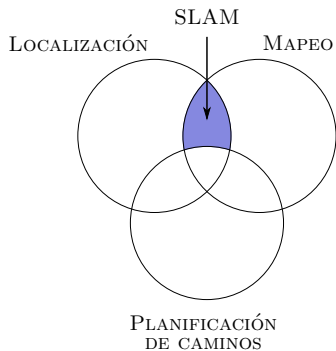
Problemas de la robótica móvil



- Localización
- Mapeo
- Planificación

Mapeo con localización conocida

Problemas de la robótica móvil



- Localización
- Mapeo
- Planificación

Localización y Mapeo Simultáneos

Motivación y aportes

Motivación

- El SLAM es un **bloque fundamental** de los esquemas de navegación autónoma
 - ▶ Utilizado para la Localización y/o el Mapeo
- Gran interés en los últimos años en **SLAM visual**
- Cualquier solución al SLAM depende fuertemente del sensor utilizado

Motivación y aportes

Motivación

- El SLAM es un **bloque fundamental** de los esquemas de navegación autónoma
 - ▶ Utilizado para la Localización y/o el Mapeo
- Gran interés en los últimos años en **SLAM visual**
- Cualquier solución al SLAM depende fuertemente del sensor utilizado

Aportes

- Diseño y desarrollo de robot móvil RoMAA-II
- Procedimiento de calibración robot/cámara
- Esquemas de **SLAM visual** para casos particulares
 - ▶ SLAM visual para mapeo de entornos de agricultura
 - ▶ SLAM visual para localización con cámara monocular

- 1 Introducción
- 2 Localización y Mapeo Simultáneos - SLAM**
 - Formulación del problema de SLAM (filtrado Bayesiano)
- 3 Robot Móvil RoMAA-II y visión robótica
- 4 SLAM con mapeo bidimensional
- 5 SLAM con mapeo tridimensional
- 6 Conclusiones y trabajos futuros

Formulación del problema de SLAM

Formulación

Se describe en términos probabilísticos utilizando **funciones de densidad de probabilidad conjunta** (pdf)

Formulación del problema de SLAM

Formulación

Se describe en términos probabilísticos utilizando **funciones de densidad de probabilidad conjunta** (pdf)

Se estima

- Full SLAM: $p(\mathcal{R}_k, \mathbf{m} | \mathcal{Z}_k, \mathcal{U}_k, \mathbf{r}_0)$
- SLAM online: $p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_k, \mathcal{U}_k, \mathbf{r}_0)$

donde

- $\mathcal{R}_k = \{\mathbf{r}_i, i = 0, \dots, k\}$, \mathbf{r} estado del robot (3DOF o 6DOF)
- \mathbf{m} mapa del entorno, el cual se asume estático $\mathbf{m}_{k-1} = \mathbf{m}_k = \mathbf{m}$
- $\mathcal{U}_k = \{\mathbf{u}_i, i = 0, \dots, k\}$, \mathbf{u} acciones de control (sensores propioceptivos)
- $\mathcal{Z}_k = \{\mathbf{z}_i, i = 0, \dots, k\}$, \mathbf{z} mediciones u observaciones (sensores exteroceptivos)

Formulación del problema de SLAM

Formulación

Se describe en términos probabilísticos utilizando **funciones de densidad de probabilidad conjunta** (pdf)

Se estima

- Full SLAM: $p(\mathcal{R}_k, \mathbf{m} | \mathcal{Z}_k, \mathcal{U}_k, \mathbf{r}_0)$
- SLAM online: $p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_k, \mathcal{U}_k, \mathbf{r}_0)$ **Filtrado bayesiano**

donde

- $\mathcal{R}_k = \{\mathbf{r}_i, i = 0, \dots, k\}$, \mathbf{r} estado del robot (3DOF o 6DOF)
- \mathbf{m} mapa del entorno, el cual se asume estático $\mathbf{m}_{k-1} = \mathbf{m}_k = \mathbf{m}$
- $\mathcal{U}_k = \{\mathbf{u}_i, i = 0, \dots, k\}$, \mathbf{u} acciones de control (sensores propioceptivos)
- $\mathcal{Z}_k = \{\mathbf{z}_i, i = 0, \dots, k\}$, \mathbf{z} mediciones u observaciones (sensores exteroceptivos)

SLAM online por Filtrado Bayesiano

Solución secuencial que opera en dos etapas

SLAM online por Filtrado Bayesiano

Solución secuencial que opera en dos etapas

Predicción

$$p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_k, \mathbf{r}_0) = \int p(\mathbf{r}_k | \mathbf{r}_{k-1}, \mathbf{u}_k) p(\mathbf{r}_{k-1}, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_{k-1}, \mathbf{r}_0) d\mathbf{r}_{k-1}$$

SLAM online por Filtrado Bayesiano

Solución secuencial que opera en dos etapas

Predicción

$$p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_k, \mathbf{r}_0) = \int p(\mathbf{r}_k | \mathbf{r}_{k-1}, \mathbf{u}_k) p(\mathbf{r}_{k-1}, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_{k-1}, \mathbf{r}_0) d\mathbf{r}_{k-1}$$

Corrección

$$p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_k, \mathcal{U}_k, \mathbf{r}_0) = \frac{p(\mathbf{z}_k | \mathbf{r}_k, \mathbf{m}) p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_k, \mathbf{r}_0)}{p(\mathbf{z}_k | \mathcal{Z}_{k-1}, \mathcal{U}_k)}$$

SLAM online por Filtrado Bayesiano

Solución secuencial que opera en dos etapas

Predicción

$$p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_k, \mathbf{r}_0) = \int p(\mathbf{r}_k | \mathbf{r}_{k-1}, \mathbf{u}_k) p(\mathbf{r}_{k-1}, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_{k-1}, \mathbf{r}_0) d\mathbf{r}_{k-1}$$

Corrección

$$p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_k, \mathcal{U}_k, \mathbf{r}_0) = \frac{p(\mathbf{z}_k | \mathbf{r}_k, \mathbf{m}) p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_k, \mathbf{r}_0)}{p(\mathbf{z}_k | \mathcal{Z}_{k-1}, \mathcal{U}_k)}$$

Modelos

- Proceso: $p(\mathbf{r}_k | \mathbf{r}_{k-1}, \mathbf{u}_k)$
- Observación: $p(\mathbf{z}_k | \mathbf{r}_k, \mathbf{m})$

SLAM online por Filtrado Bayesiano

Solución secuencial que opera en dos etapas

Predicción

$$p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_k, \mathbf{r}_0) = \int p(\mathbf{r}_k | \mathbf{r}_{k-1}, \mathbf{u}_k) p(\mathbf{r}_{k-1}, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_{k-1}, \mathbf{r}_0) d\mathbf{r}_{k-1}$$

Corrección

$$p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_k, \mathcal{U}_k, \mathbf{r}_0) = \frac{p(\mathbf{z}_k | \mathbf{r}_k, \mathbf{m}) p(\mathbf{r}_k, \mathbf{m} | \mathcal{Z}_{k-1}, \mathcal{U}_k, \mathbf{r}_0)}{p(\mathbf{z}_k | \mathcal{Z}_{k-1}, \mathcal{U}_k)}$$

Modelos

- Proceso: $p(\mathbf{r}_k | \mathbf{r}_{k-1}, \mathbf{u}_k)$
- Observación: $p(\mathbf{z}_k | \mathbf{r}_k, \mathbf{m})$

Implementación

- Filtro de partículas (no gaussiano)
- Filtro de Kalman (gaussiano)
EKF, UKF, EIF, etc.

SLAM con Filtro de Kalman, KF-SLAM

Ecuaciones estocásticas en espacio de estados

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_k,$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k,$$

$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k): \text{ruido de proceso}$$

$$\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k): \text{ruido de medición}$$

SLAM con Filtro de Kalman, KF-SLAM

Ecuaciones estocásticas en espacio de estados

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_k,$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k,$$

$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k): \text{ruido de proceso}$$

$$\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k): \text{ruido de medición}$$

Vector de estado

Asumiendo el mapa estático, o sea

$$\mathbf{m}_k = \mathbf{m}_{k-1} = \mathbf{m}.$$

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{r}_k \\ \mathbf{m} \end{bmatrix}, \quad \text{con} \quad \mathbf{m} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \vdots \\ \mathbf{m}_N \end{bmatrix}$$

Matriz de covarianza

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm} \\ \mathbf{P}_{mr} & \mathbf{P}_{mm} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \cdots & \mathbf{P}_{rm_N} \\ \mathbf{P}_{m_1r} & \mathbf{P}_{m_1m_1} & \cdots & \mathbf{P}_{m_1m_N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m_Nr} & \mathbf{P}_{m_Nm_1} & \cdots & \mathbf{P}_{m_Nm_N} \end{bmatrix}$$

SLAM con Filtro de Kalman, KF-SLAM

Ecuaciones estocásticas en espacio de estados

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_k,$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k,$$

$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k): \text{ruido de proceso}$$

$$\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k): \text{ruido de medición}$$

Vector de estado

Asumiendo el mapa estático, o sea

$$\mathbf{m}_k = \mathbf{m}_{k-1} = \mathbf{m}.$$

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{r}_k \\ \mathbf{m} \end{bmatrix}, \quad \text{con} \quad \mathbf{m} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \vdots \\ \mathbf{m}_N \end{bmatrix}$$

Matriz de covarianza

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm} \\ \mathbf{P}_{mr} & \mathbf{P}_{mm} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \cdots & \mathbf{P}_{rm_N} \\ \mathbf{P}_{m_1r} & \mathbf{P}_{m_1m_1} & \cdots & \mathbf{P}_{m_1m_N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m_Nr} & \mathbf{P}_{m_Nm_1} & \cdots & \mathbf{P}_{m_Nm_N} \end{bmatrix}$$

SLAM con Filtro de Kalman, KF-SLAM

Ecuaciones estocásticas en espacio de estados

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_k,$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k,$$

$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k): \text{ ruido de proceso}$$

$$\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k): \text{ ruido de medición}$$

Vector de estado

Asumiendo el mapa estático, o sea

$$\mathbf{m}_k = \mathbf{m}_{k-1} = \mathbf{m}.$$

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{r}_k \\ \mathbf{m} \end{bmatrix}, \quad \text{con} \quad \mathbf{m} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \vdots \\ \mathbf{m}_N \end{bmatrix}$$

Matriz de covarianza

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm} \\ \mathbf{P}_{mr} & \mathbf{P}_{mm} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \cdots & \mathbf{P}_{rm_N} \\ \mathbf{P}_{m_1r} & \mathbf{P}_{m_1m_1} & \cdots & \mathbf{P}_{m_1m_N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m_Nr} & \mathbf{P}_{m_Nm_1} & \cdots & \mathbf{P}_{m_Nm_N} \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{r}_k \\ \mathbf{m}_k \end{bmatrix} = \begin{bmatrix} \mathbf{f}_r(\mathbf{r}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_k \\ \mathbf{m}_{k-1} \end{bmatrix}$$

SLAM con Filtro de Kalman, KF-SLAM

Ecuaciones estocásticas en espacio de estados

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_k,$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k,$$

$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k): \text{ ruido de proceso}$$

$$\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k): \text{ ruido de medición}$$

Vector de estado

Asumiendo el mapa estático, o sea

$$\mathbf{m}_k = \mathbf{m}_{k-1} = \mathbf{m}.$$

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{r}_k \\ \mathbf{m} \end{bmatrix}, \quad \text{con} \quad \mathbf{m} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \vdots \\ \mathbf{m}_N \end{bmatrix}$$

Matriz de covarianza

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm} \\ \mathbf{P}_{mr} & \mathbf{P}_{mm} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \cdots & \mathbf{P}_{rm_N} \\ \mathbf{P}_{m_1r} & \mathbf{P}_{m_1m_1} & \cdots & \mathbf{P}_{m_1m_N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m_Nr} & \mathbf{P}_{m_Nm_1} & \cdots & \mathbf{P}_{m_Nm_N} \end{bmatrix}$$

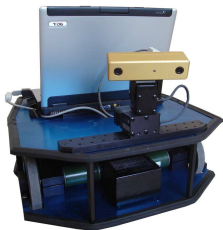
$$\begin{bmatrix} \mathbf{r}_k \\ \mathbf{m}_k \end{bmatrix} = \begin{bmatrix} \mathbf{f}_r(\mathbf{r}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_k \\ \mathbf{m}_{k-1} \end{bmatrix}$$

- Modelo de movimiento: $\mathbf{f}_r(\cdot)$
- Modelo de observación: $\mathbf{h}(\cdot)$

- 1 Introducción
- 2 Localización y Mapeo Simultáneos - SLAM
- 3 Robot Móvil RoMAA-II y visión robótica
 - Robot Móvil de Arquitectura Abierta, RoMAA
 - Parámetros de cámaras y calibración
- 4 SLAM con mapeo bidimensional
- 5 SLAM con mapeo tridimensional
- 6 Conclusiones y trabajos futuros

Robot Móvil de Arquitectura Abierta, RoMAA

RoMAA (prototipo)



Robot móvil de tracción diferencial

RoMAA-II



Robot Móvil de Arquitectura Abierta, RoMAA

RoMAA (prototipo)

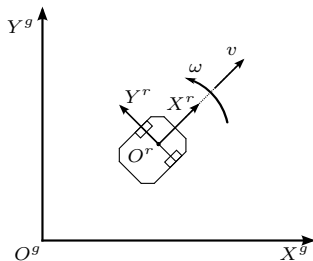


RoMAA-II



Robot móvil de tracción diferencial

- El robot se controla mediante velocidad lineal v y angular ω



Robot Móvil de Arquitectura Abierta, RoMAA

RoMAA (prototipo)

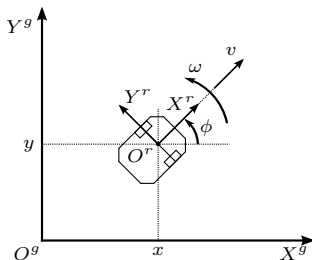


RoMAA-II



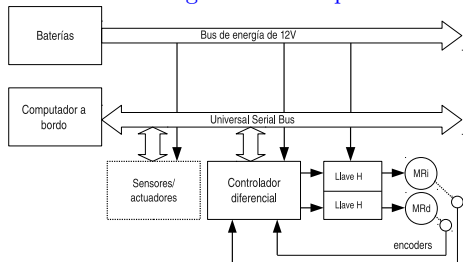
Robot móvil de tracción diferencial

- El robot se controla mediante velocidad lineal v y angular ω
- La odometría permite conocer la localización (x, y, ϕ)



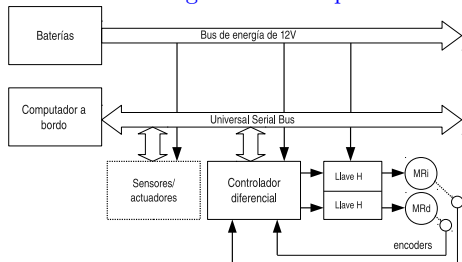
El robot móvil RoMAA-II

Diagrama en bloques



El robot móvil RoMAA-II

Diagrama en bloques



Parámetros de cámaras y calibración (modelo pinhole)



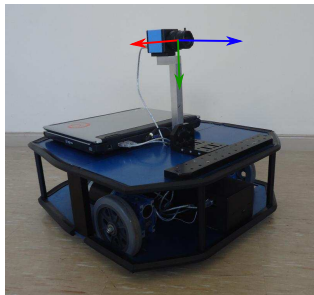
Parámetros de cámaras y calibración (modelo pinhole)



Matriz de proyección

$$\mathbf{P} = \mathbf{K} \left[\mathbf{R} \mid \mathbf{t} \right]$$

Parámetros de cámaras y calibración (modelo pinhole)

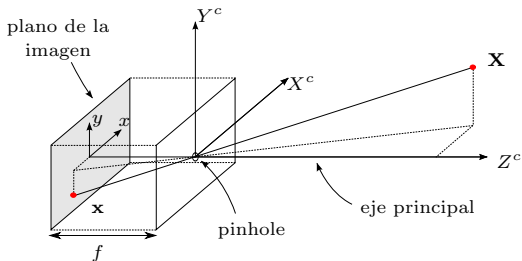


Matriz de proyección

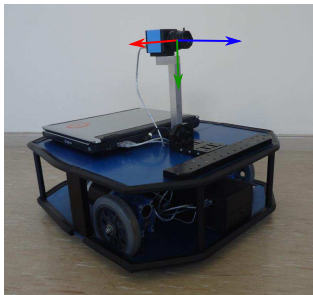
$$\mathbf{P} = \mathbf{K} \left[\mathbf{R} \mid \mathbf{t} \right]$$

- Parámetros intrínsecos

$$\mathbf{K} = \begin{bmatrix} f/h_u & 0 & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$



Parámetros de cámaras y calibración (modelo pinhole)

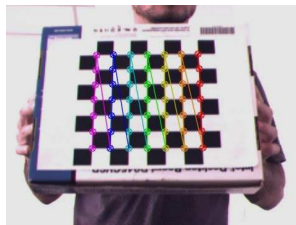
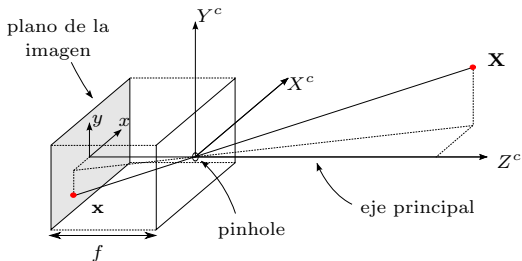


Matriz de proyección

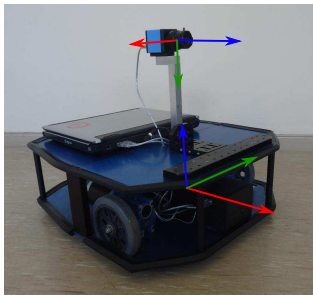
$$\mathbf{P} = \mathbf{K} \left[\mathbf{R} \mid \mathbf{t} \right]$$

- Parámetros intrínsecos

$$\mathbf{K} = \begin{bmatrix} f/h_u & 0 & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$



Parámetros de cámaras y calibración (modelo pinhole)

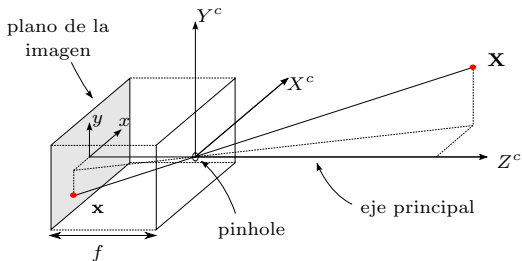


Matriz de proyección

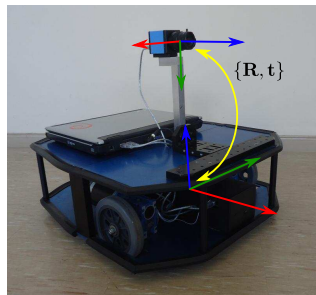
$$\mathbf{P} = \mathbf{K} \left[\mathbf{R} \mid \mathbf{t} \right]$$

- Parámetros intrínsecos

$$\mathbf{K} = \begin{bmatrix} f/h_u & 0 & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$



Parámetros de cámaras y calibración (modelo pinhole)



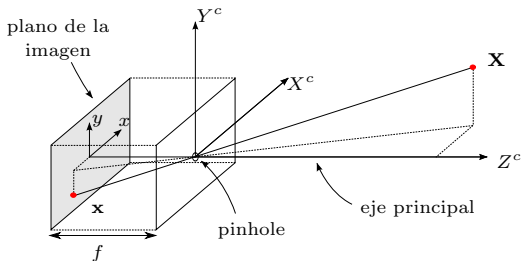
Matriz de proyección

$$\mathbf{P} = \mathbf{K} \left[\mathbf{R} \mid \mathbf{t} \right]$$

- Parámetros intrínsecos

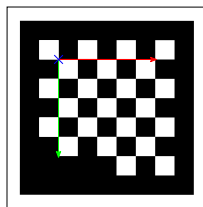
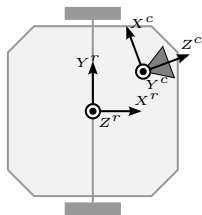
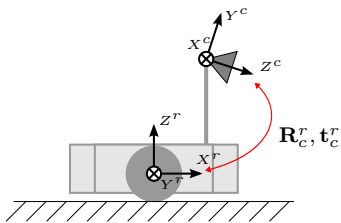
$$\mathbf{K} = \begin{bmatrix} f/h_u & 0 & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Parámetros extrínsecos $\{\mathbf{R}, \mathbf{t}\}$



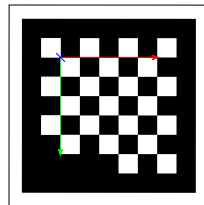
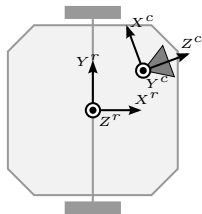
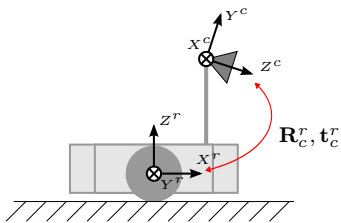
Calibración robot/cámara

- Basado en trayectorias simple
- Estimación en la estimación de pose entre cámara y fiducial



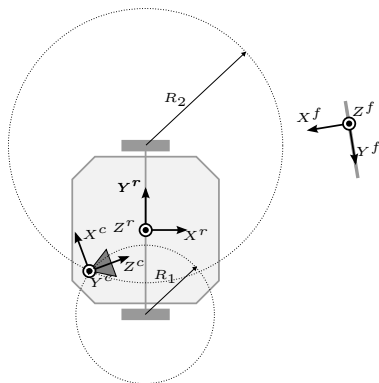
Calibración robot/cámara

- Basado en trayectorias simple
- Estimación en la estimación de pose entre cámara y fiducial



Estimación del vector de traslación t_c^r

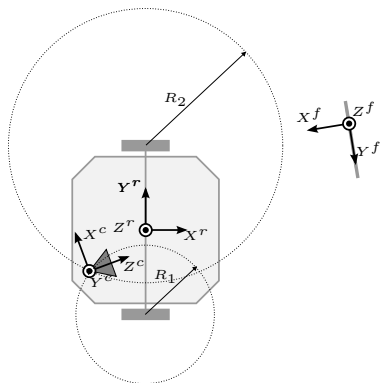
Determinar los radios R_1 y R_2 de las trayectorias.



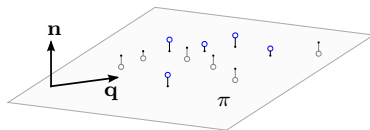
$$x_r^2 + \left(y_r + \frac{b_w}{2} \right)^2 = R_1^2$$

$$x_r^2 + \left(y_r - \frac{b_w}{2} \right)^2 = R_2^2$$

Estimación del vector de traslación t_c^r



La trayectoria de la cámara es paralela al piso, por ende pertenece a un plano.

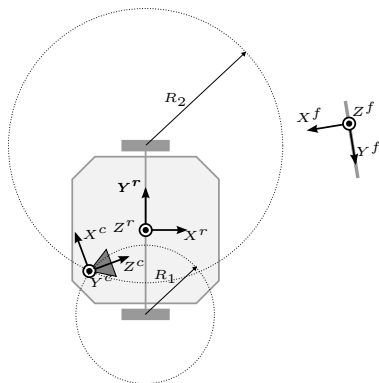


Determinar \mathbf{n} y \mathbf{q} (plano más cercano), luego se recupera la rotación.

Y se transforma el conjunto de puntos 3D (mismo z en el SCF).

Estimación del vector de traslación t_c^r

La intersección determina las coordenadas x_c^r e y_c^r de la cámara en el sistema de coordenadas del robot.

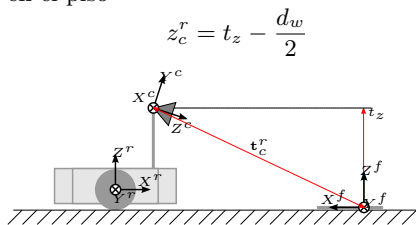
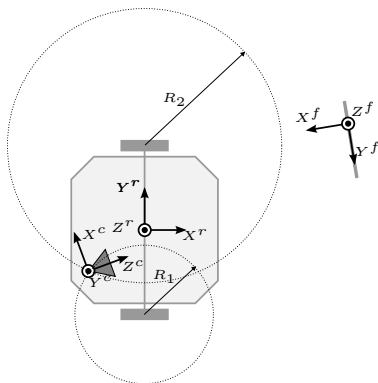


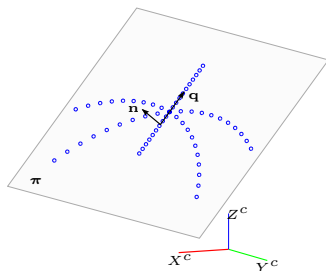
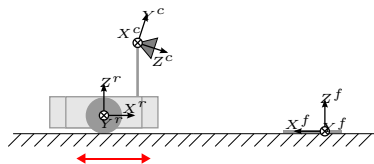
$$x_c^r = \pm \sqrt{R_1^2 - \left(y_c^r + \frac{b_w}{2}\right)^2}$$

$$y_c^r = \frac{R_1^2 - R_2^2}{2b_w}$$

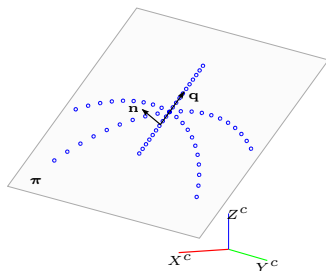
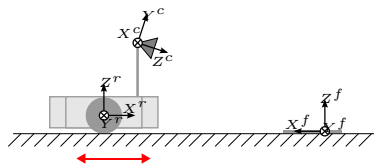
Estimación del vector de traslación \mathbf{t}_c^r

La calibración de \mathbf{t}_c^r se completa estimando la pose de la cámara respecto de un patrón en el piso



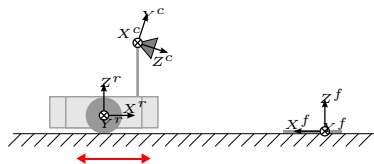
Calibración de la matriz de rotación R_c^r 

Con una trayectoria recta se determina la dirección del eje X^r (vector \mathbf{q})

Calibración de la matriz de rotación R_c^r 

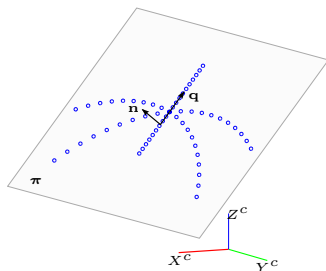
Con una trayectoria recta se determina la dirección del eje X^r (vector \mathbf{q})

La normal al plano más cercano \mathbf{n} es la dirección del eje Z^r

Calibración de la matriz de rotación R_c^r 

El vector producto $\mathbf{n} \times \mathbf{q}$ determina la dirección del eje Y_r , luego

$$\mathbf{R}_c^r = [\mathbf{q} \quad (\mathbf{n} \times \mathbf{q}) \quad \mathbf{n}]$$



Con una trayectoria recta se determina la dirección del eje X^r (vector \mathbf{q})

La normal al plano más cercano \mathbf{n} es la dirección del eje Z^r

Calibración de un sistema robot/cámara real



Resultados de la calibración

parámetro	estimado media (std. dev.)	medido a mano
x_C^r	72,6 (1,86)	70
y_C^r	16,5 (2,6)	20
z_C^r	277,9 (0,7)	270
inclinación	29,09 (0,04)	30

Unidades en $[mm]$ y $[^\circ]$.

$$b_w = 455mm, \quad d_w = 138mm$$

Calibración de un sistema robot/cámara real

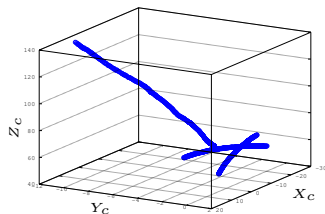


Resultados de la calibración

parámetro	estimado media (std. dev.)	medido a mano
x_C^r	72,6 (1,86)	70
y_C^r	16,5 (2,6)	20
z_C^r	277,9 (0,7)	270
inclinación	29,09 (0,04)	30

Unidades en [mm] y [°].

$$b_w = 455mm, \quad d_w = 138mm$$

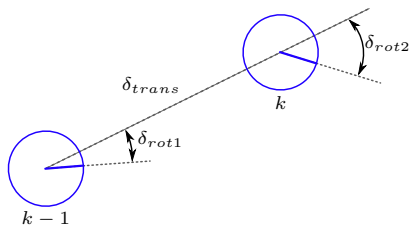


Alta **sensibilidad** a

- ruido en la estimación de la pose
- longitud de los arcos de circunferencia

- 1 Introducción
- 2 Localización y Mapeo Simultáneos - SLAM
- 3 Robot Móvil RoMAA-II y visión robótica
- 4 SLAM con mapeo bidimensional**
 - SLAM con sensor de barrido láser
 - SLAM en entornos de agricultura
- 5 SLAM con mapeo tridimensional
- 6 Conclusiones y trabajos futuros

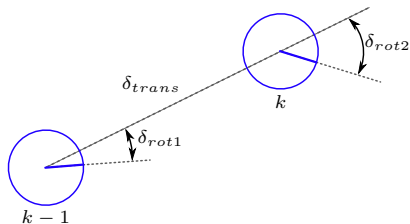
SLAM2D–Modelo de movimiento basado en odometría



$$\mathbf{r}_k = \mathbf{f}_r(\mathbf{r}_{k-1}, \mathbf{u}_{k-1})$$

$$\mathbf{r} = \begin{bmatrix} x \\ y \\ \phi \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \delta_{rot1} \\ \delta_{trans} \\ \delta_{rot2} \end{bmatrix}$$

SLAM2D–Modelo de movimiento basado en odometría

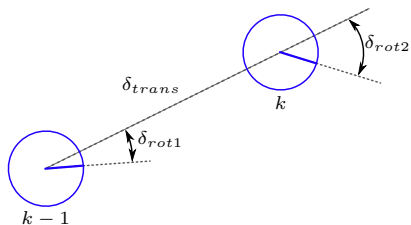


$$\mathbf{r}_k = \mathbf{f}_r(\mathbf{r}_{k-1}, \mathbf{u}_{k-1})$$

$$\mathbf{r} = \begin{bmatrix} x \\ y \\ \phi \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \delta_{rot1} \\ \delta_{trans} \\ \delta_{rot2} \end{bmatrix}$$

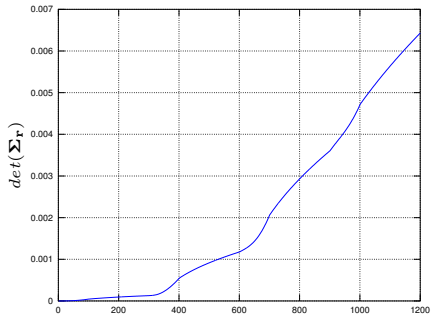
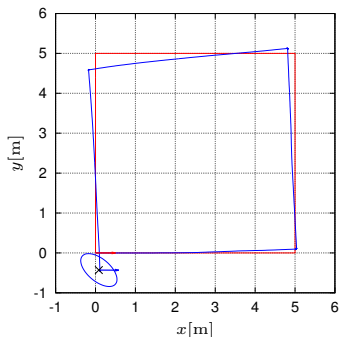
$$\begin{bmatrix} x_k \\ y_k \\ \phi_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \phi_{k-1} \end{bmatrix} + \begin{bmatrix} \delta_{trans} \cos(\phi_{k-1} + \delta_{rot1}) \\ \delta_{trans} \sin(\phi_{k-1} + \delta_{rot1}) \\ \delta_{rot1} + \delta_{rot2} \end{bmatrix}$$

SLAM2D–Modelo de movimiento basado en odometría



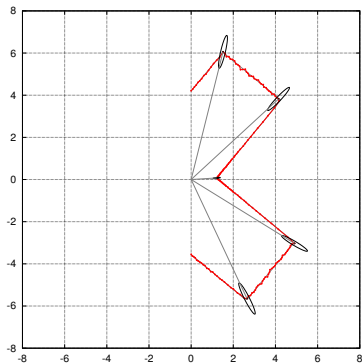
$$\mathbf{r}_k = \mathbf{f}_r(\mathbf{r}_{k-1}, \mathbf{u}_{k-1})$$

$$\mathbf{r} = \begin{bmatrix} x \\ y \\ \phi \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \delta_{rot1} \\ \delta_{trans} \\ \delta_{rot2} \end{bmatrix}$$

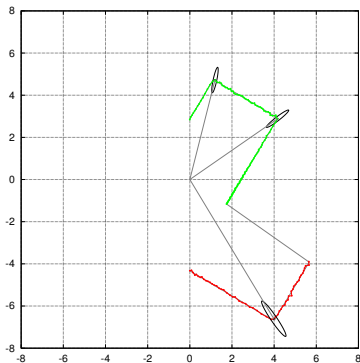


SLAM2D–Clustering y extracción de esquinas

Un único cluster

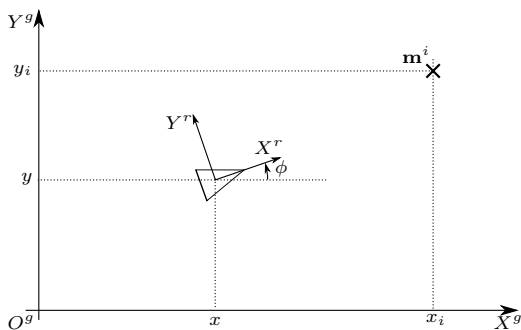


Dos clusters



SLAM2D–Modelo de medición

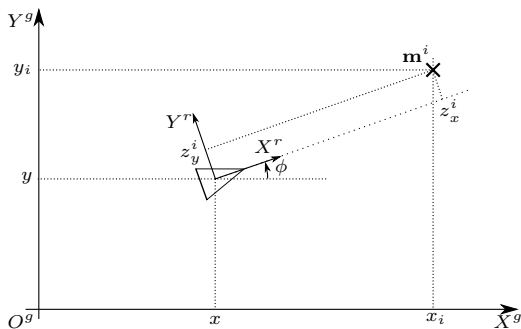
- Dada la pose del robot \mathbf{r}_k y el i -ésimo landmark \mathbf{m}_k^i (en el SCG)



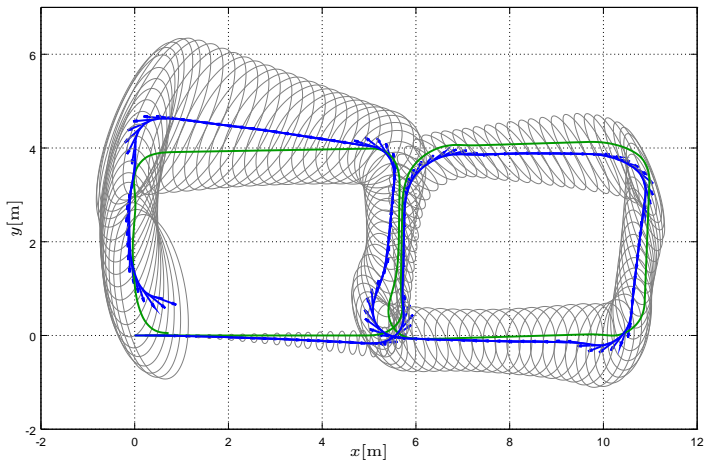
SLAM2D–Modelo de medición

- Dada la pose del robot \mathbf{r}_k y el i -ésimo landmark \mathbf{m}_k^i (en el SCG)
- $\mathbf{z}_k^i = [z_{x,k}^i \quad z_{y,k}^i]^T$, i -ésima característica en el SCR

$$\mathbf{z}_k^i = \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k)$$

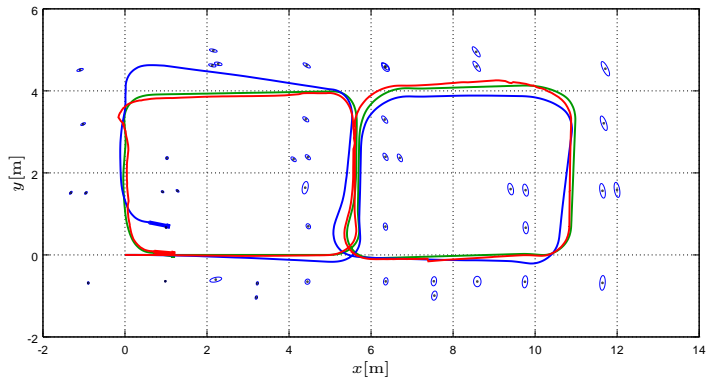


Resultados de simulación de SLAM2D



Resultados de simulación de SLAM2D

Ground thruth - Odometría - Estimación



SLAM2D en entornos de agricultura

- Vehículo autónomo no tripulado (tracción Ackerman)
- Sistema de visión monocular + sensor de barrido láser
- Troncos de árboles como landmarks bidimensionales



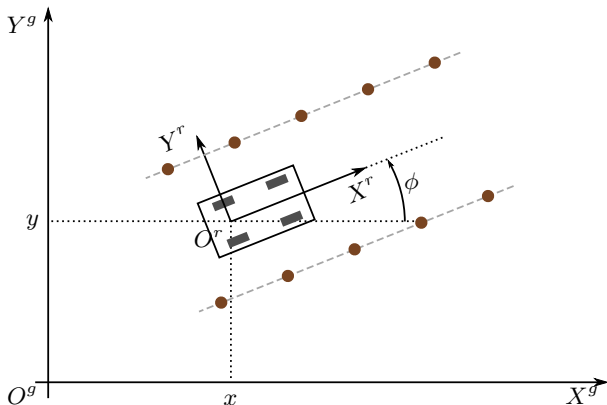
SLAM2D en entornos de agricultura–Modelo de movimiento

$$\dot{x} = v \cos \phi$$

$$\dot{y} = v \sin \phi$$

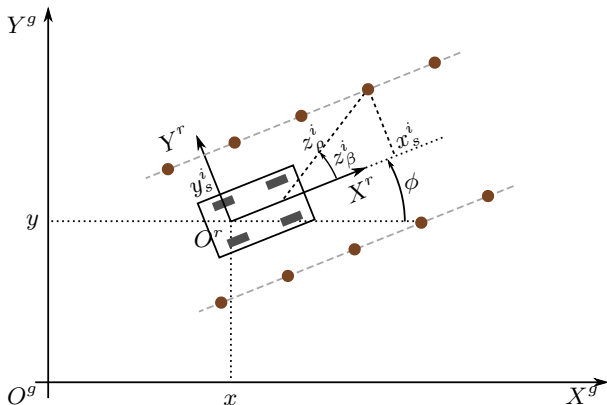
$$\dot{\phi} = \frac{v}{L} \tan \zeta$$

- $\mathbf{r} = [x \quad y \quad \phi]^T$: estado del robot
- $\mathbf{u} = [v \quad \zeta]^T$: acción del control
- ζ : ángulo de la rueda de dirección
- v : velocidad lineal



SLAM2D en entornos de agricultura–Modelo de observación

$$\mathbf{z}_k^i = \mathbf{h}_i(\mathbf{r}_k, x_s^i, y_s^i, \mathbf{v}_k) = \begin{bmatrix} \sqrt{(x_k - x_s^i)^2 + (y_k - y_s^i)^2} \\ \arctan\left(\frac{y_k - y_s^i}{x_k - x_s^i}\right) - \phi \end{bmatrix} + \begin{bmatrix} v_\rho \\ v_\beta \end{bmatrix} = \begin{bmatrix} z_{\rho,k}^i \\ z_{\beta,k}^i \end{bmatrix}$$



SLAM2D en entornos de agricultura–Detección de troncos de árboles

- Determinar la información de ángulo
- Determinar la distancia

SLAM2D en entornos de agricultura–Detección de troncos de árboles

- Determinar la información de ángulo
- Determinar la distancia

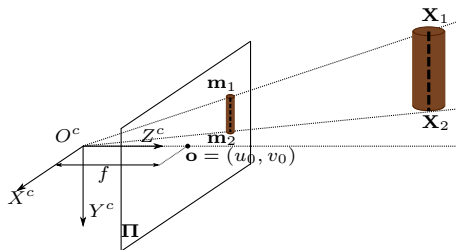


Sistema de visión monocular

- Máquina de soporte vectorial (SVM, Support Vector Machine)
- Descriptor en espacio de características con Histograma de Gradientes Orientados (HOG, Histogram of Oriented Gradients)

SLAM2D en entornos de agricultura–Detección de troncos de árboles

- Determinar la información de ángulo
- Determinar la distancia

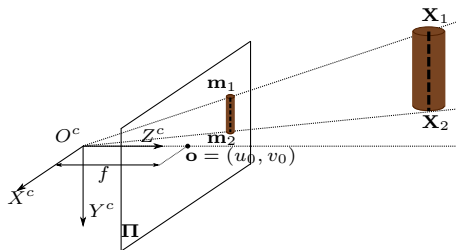


Proyección basada en el modelo pin-hole

$$\mathbf{m}_i \simeq \mathbf{P}\mathbf{X}_i = \mathbf{K}[\mathbf{R} | \mathbf{t}]\mathbf{X}_i = \mathbf{K}\mathbf{x}_i$$

SLAM2D en entornos de agricultura–Detección de troncos de árboles

- Determinar la información de ángulo
- Determinar la distancia



Proyección basada en el modelo pin-hole

$$\mathbf{m}_i \simeq \mathbf{P}\mathbf{X}_i = \mathbf{K}[\mathbf{R} \mid \mathbf{t}]\mathbf{X}_i = \mathbf{K}\mathbf{x}_i$$

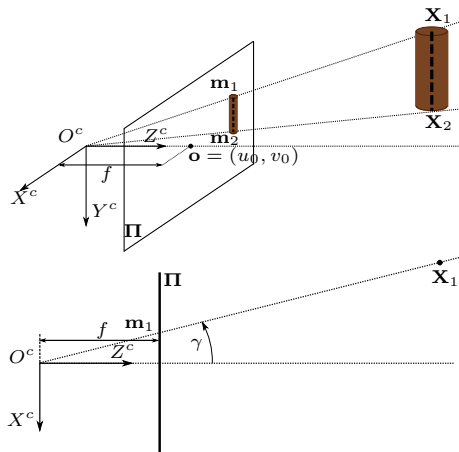
Conociendo los parámetros intrínsecos \mathbf{K}

$$\mathbf{x}_i = \mathbf{K}^{-1}\mathbf{m}_i,$$

donde $\mathbf{x}_i = [x_i^c \quad y_i^c \quad f]^T$.

SLAM2D en entornos de agricultura–Detección de troncos de árboles

- Determinar la información de ángulo
- Determinar la distancia



Proyección basada en el modelo pin-hole

$$\mathbf{m}_i \simeq \mathbf{P}\mathbf{X}_i = \mathbf{K}[\mathbf{R} | \mathbf{t}]\mathbf{X}_i = \mathbf{K}\mathbf{x}_i$$

Conociendo los parámetros intrínsecos \mathbf{K}

$$\mathbf{x}_i = \mathbf{K}^{-1}\mathbf{m}_i,$$

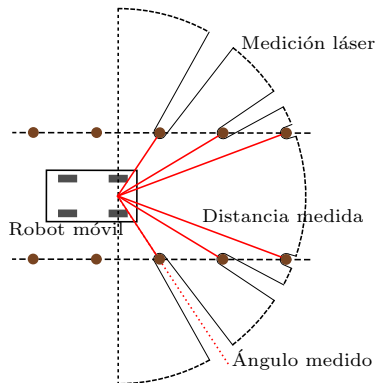
donde $\mathbf{x}_i = [x_i^c \ y_i^c \ f]^T$.

La información angular o de dirección del landmark resulta

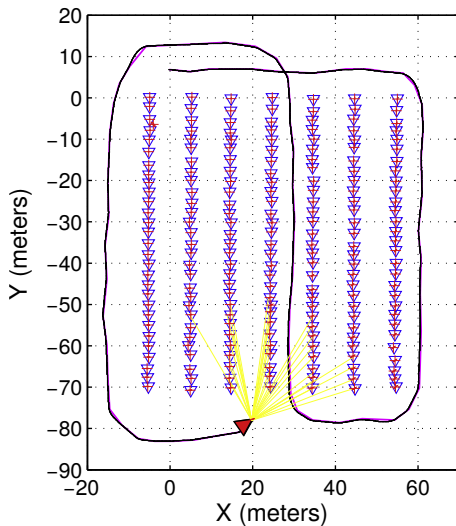
$$\gamma_i = \arctan(x_i^c/f).$$

SLAM2D en entornos de agricultura–Detección de troncos de árboles

- Determinar la información de ángulo
- **Determinar la distancia**



SLAM2D en entornos de agricultura—Resultados

Camino estimado y **Ground truth (GPS)**

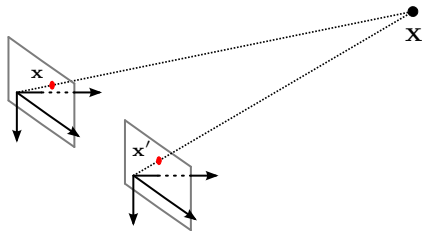
- 1 Introducción
- 2 Localización y Mapeo Simultáneos - SLAM
- 3 Robot Móvil RoMAA-II y visión robótica
- 4 SLAM con mapeo bidimensional
- 5 SLAM con mapeo tridimensional**
 - Resultados de simulación
 - Resultados experimentales para localización
- 6 Conclusiones y trabajos futuros

Introducción al SLAM visual

SLAM visual

- Utilizando visión **estéreo** o **monocular**

Visión estéreo - Triangulación

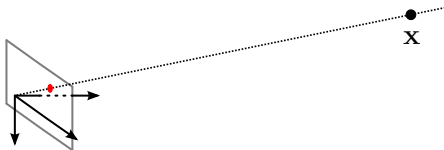


Introducción al SLAM visual

SLAM visual

- Utilizando visión **estéreo** o **monocular**
- SLAM monocular
 - ▶ Imposibilidad de determinar la profundidad de los landmarks

Visión monocular

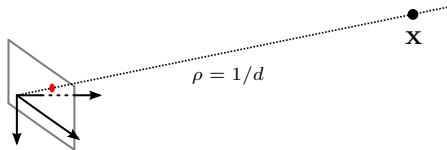


Introducción al SLAM visual

SLAM visual

- Utilizando visión **estéreo** o **monocular**
- SLAM monocular
 - ▶ Imposibilidad de determinar la profundidad de los landmarks

Parametrización con profundidad inversa



Introducción al SLAM visual

SLAM visual

- Utilizando visión **estéreo** o **monocular**
- SLAM monocular
 - ▶ Imposibilidad de determinar la profundidad de los landmarks
 - ▶ Esquemas clásicos basados en **EKF** con **modelo de velocidad constante**

Introducción al SLAM visual

SLAM visual

- Utilizando visión **estéreo** o **monocular**
- SLAM monocular
 - ▶ Imposibilidad de determinar la profundidad de los landmarks
 - ▶ Esquemas clásicos basados en **EKF** con **modelo de velocidad constante**

Se presenta

- Esquema de SLAM3D con cámara monocular
 - ▶ Modelo de movimiento de odometría (escala)
 - ▶ Comparación entre EKF y UKF (simulación)

Introducción al SLAM visual

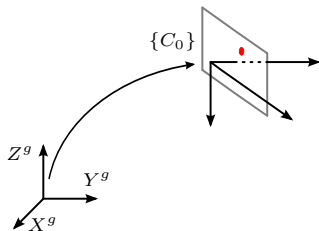
SLAM visual

- Utilizando visión **estéreo** o **monocular**
- SLAM monocular
 - ▶ Imposibilidad de determinar la profundidad de los landmarks
 - ▶ Esquemas clásicos basados en **EKF** con **modelo de velocidad constante**

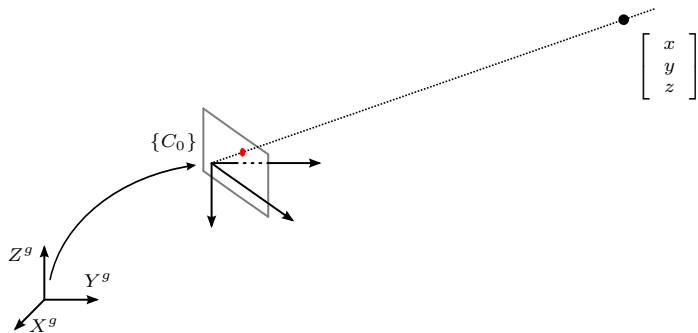
Se presenta

- Esquema de SLAM3D con cámara monocular
 - ▶ Modelo de movimiento de odometría (escala)
 - ▶ Comparación entre EKF y UKF (simulación)
- Esquema de SLAM monocular para localización con UKF
 - ▶ Resultados experimentales utilizando el robot RoMAA-II

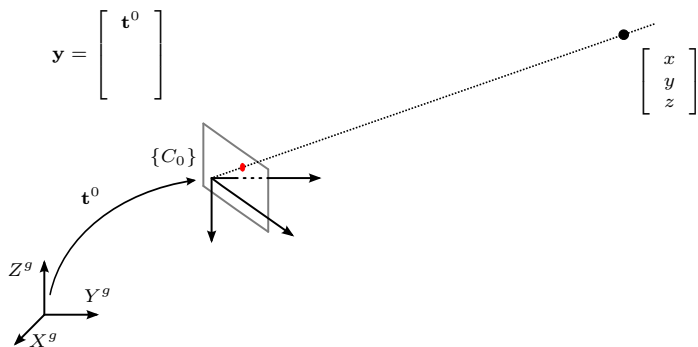
Parametrización en profundidad inversa



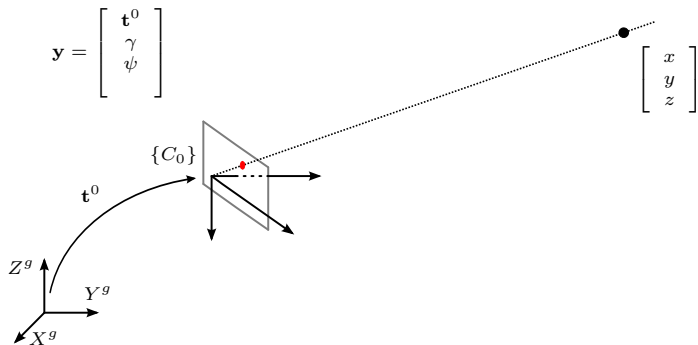
Parametrización en profundidad inversa



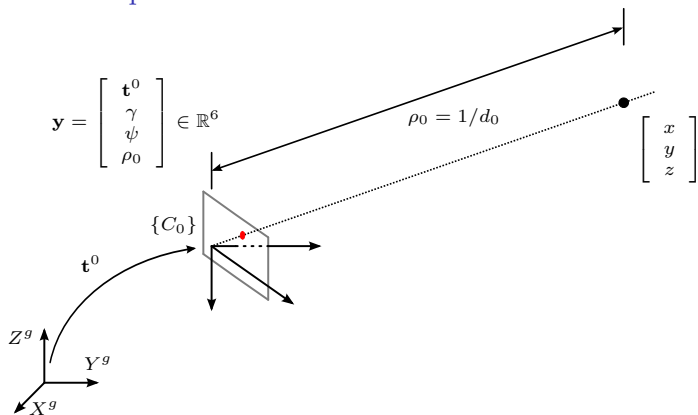
Parametrización en profundidad inversa



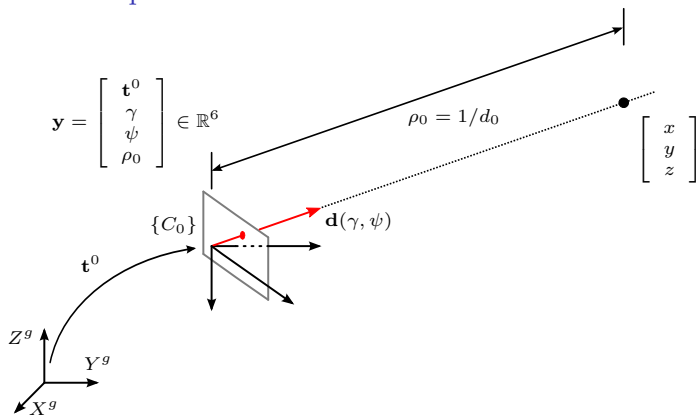
Parametrización en profundidad inversa



Parametrización en profundidad inversa



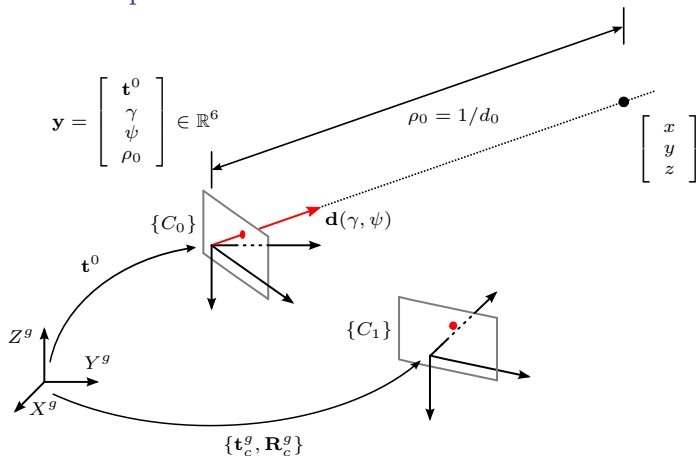
Parametrización en profundidad inversa



Conversión ID a EU

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \\ z^0 \end{bmatrix} + \frac{1}{\rho} \mathbf{d}(\gamma, \psi),$$

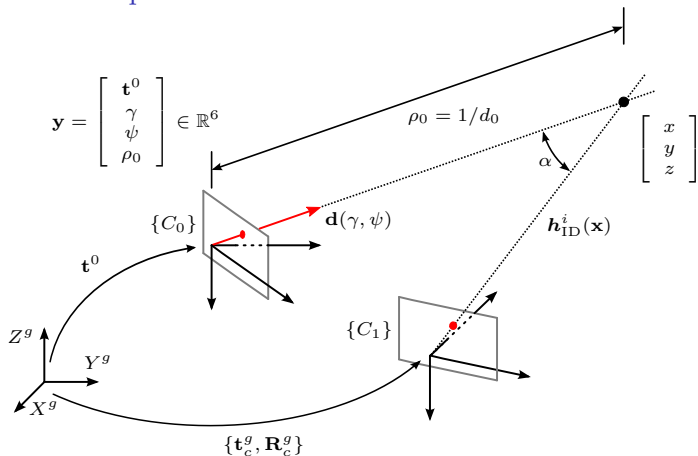
Parametrización en profundidad inversa



Conversión ID a EU

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \\ z^0 \end{bmatrix} + \frac{1}{\rho} \mathbf{d}(\gamma, \psi),$$

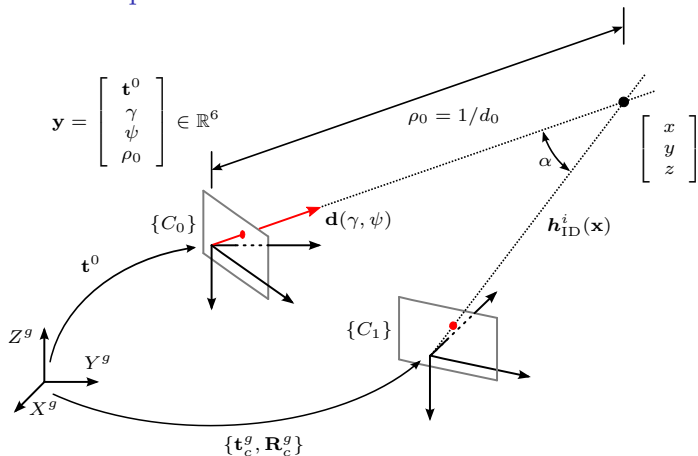
Parametrización en profundidad inversa



Conversión ID a EU

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \\ z^0 \end{bmatrix} + \frac{1}{\rho} \mathbf{d}(\gamma, \psi),$$

Parametrización en profundidad inversa



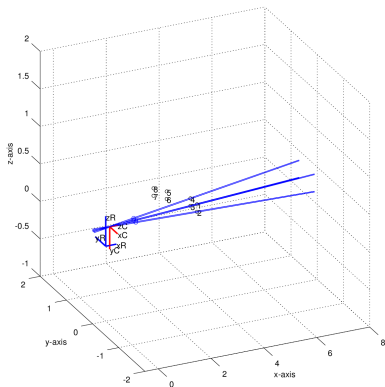
Conversión ID a EU

Modelo de medición

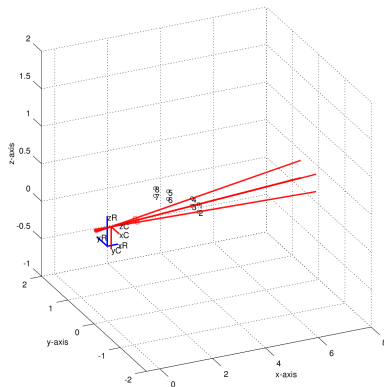
$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \\ z^0 \end{bmatrix} + \frac{1}{\rho} \mathbf{d}(\gamma, \psi), \quad h_{ID}^i(\mathbf{x}) = \mathbf{R}_c^g \left(\begin{bmatrix} x_i^0 \\ y_i^0 \\ z_i^0 \end{bmatrix} + \frac{1}{\rho_i} \mathbf{d}(\gamma_i, \psi_i) - \mathbf{t}_c^g \right)$$

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

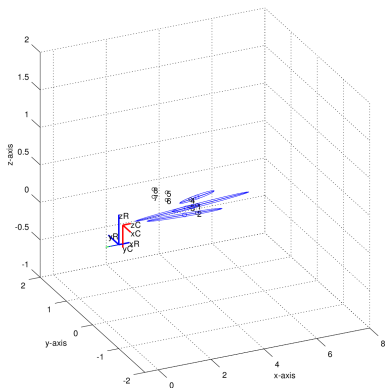


Filtro de Kalman Unscented

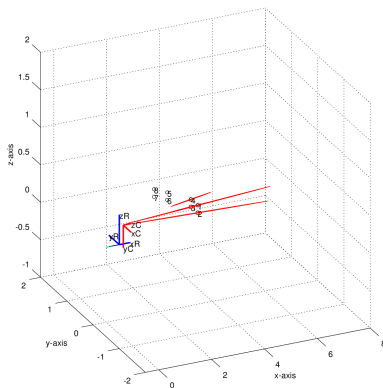
Paso temporal $k = 1$ (1/8)

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

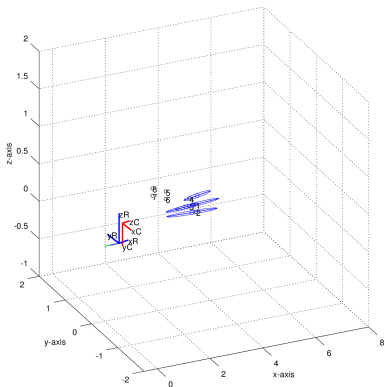


Filtro de Kalman Unscented

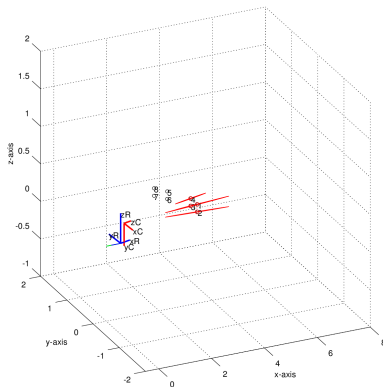
Paso temporal $k = 100$ (2/8)

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

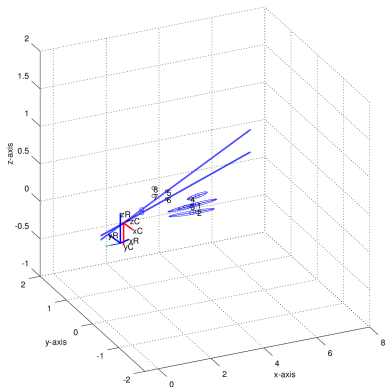


Filtro de Kalman Unscented

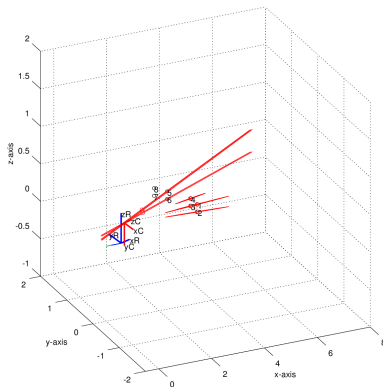
Paso temporal $k = 150$ (3/8)

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

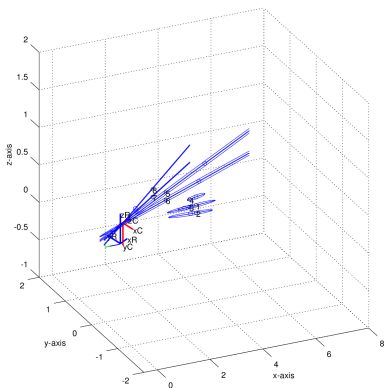


Filtro de Kalman Unscented

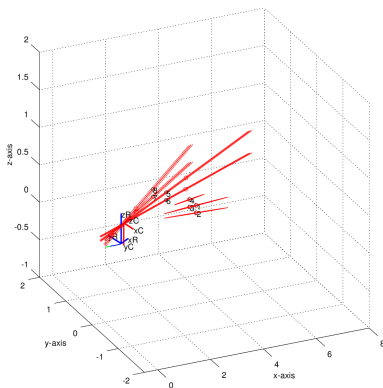
Paso temporal $k = 170$ (4/8)

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

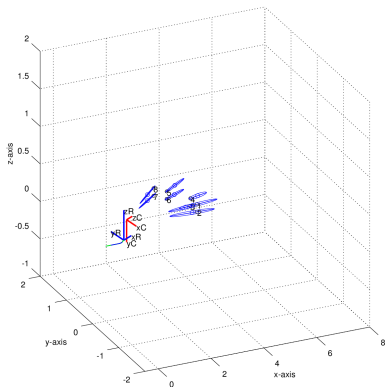


Filtro de Kalman Unscented

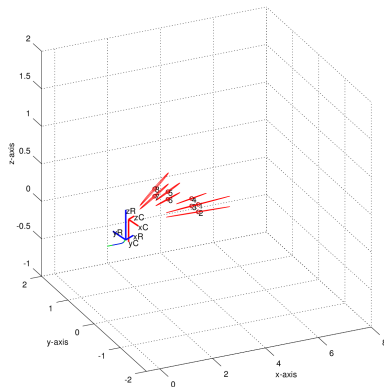
Paso temporal $k = 220$ (5/8)

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

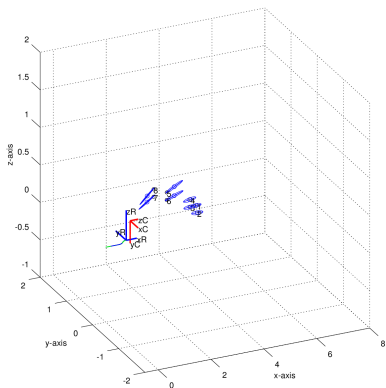


Filtro de Kalman Unscented

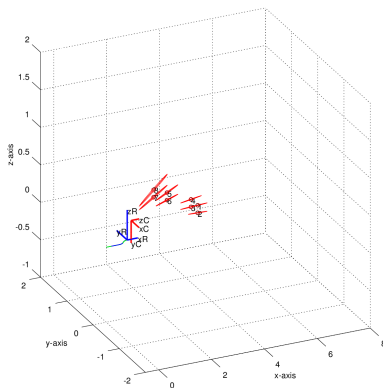
Paso temporal $k = 400$ (6/8)

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

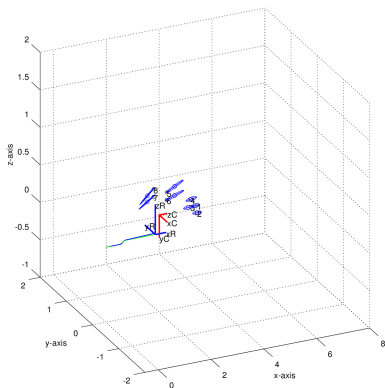


Filtro de Kalman Unscented

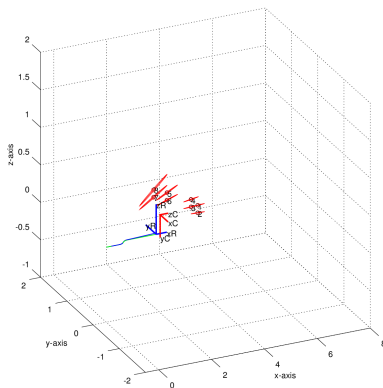
Paso temporal $k = 500$ (7/8)

Resultados de simulación - EKF vs. UKF (1)

Filtro de Kalman Extendido

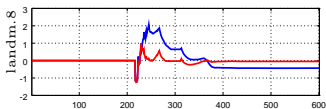
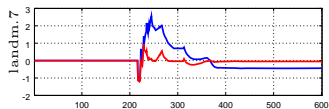
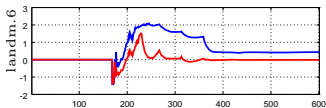
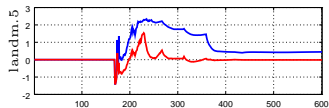
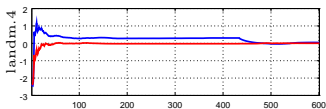
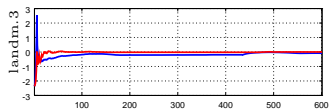
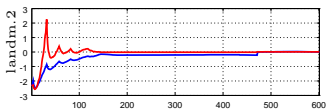
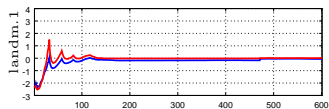


Filtro de Kalman Unscented

Paso temporal $k = 600$ (8/8)

Resultados de simulación - EKF vs. UKF (1)

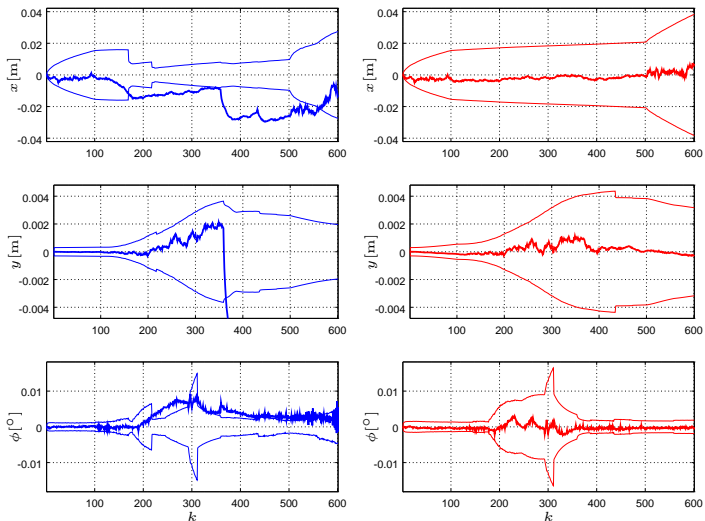
Error en la estimación de profundidad de cada landmarks
(EKF - UKF)



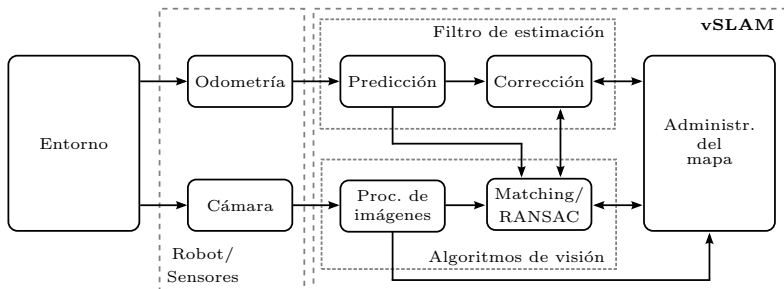
Resultados de simulación - EKF vs. UKF (1)

Estimación de la pose del robot

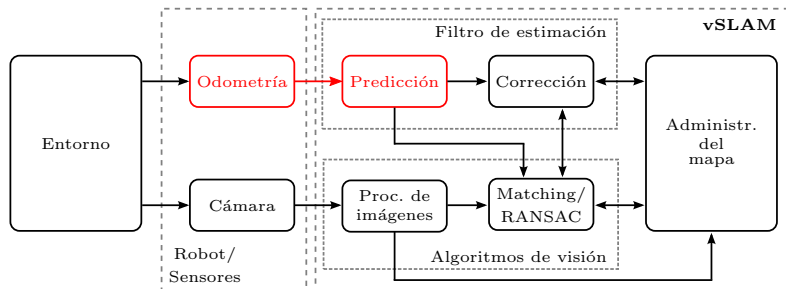
(EKF - UKF)



SLAM monocular en robot móvil

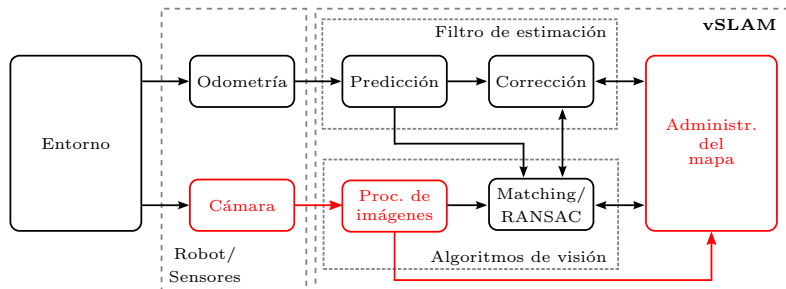


SLAM monocular en robot móvil



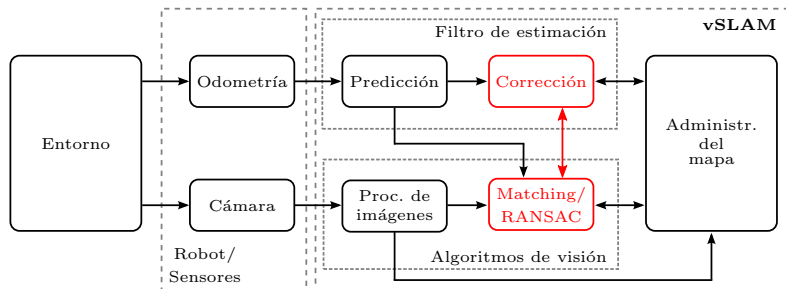
- Odometría como sensor propioceptivos

SLAM monocular en robot móvil



- Odometría como sensor propioceptivos
- Cámara monocular:
 - ▶ Extracción de puntos de interés Harris
 - ▶ Parametrización en profundidad inversa y euclídea de puntos 3D

SLAM monocular en robot móvil



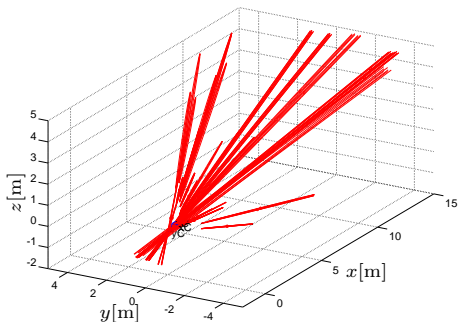
- Odometría como sensor propioceptivos
- Cámara monocular:
 - ▶ Extracción de puntos de interés Harris
 - ▶ Parametrización en profundidad inversa y euclídea de puntos 3D
- Asociación de datos (matching activo):
 - ▶ Correlación cruzada normalizada (NCC)
 - ▶ RANSAC de 1-punto

Resultados experimentales de UKF-SLAM. Plataforma



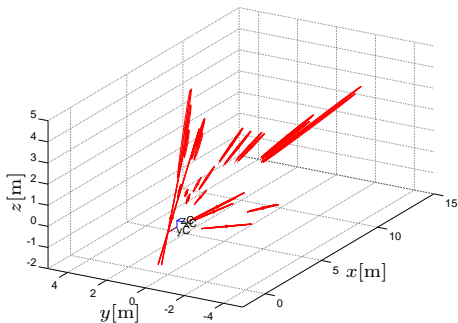
- Robot móvil RoMAA-II
- Resolución de imágenes 640x480 píxeles
- Almacenamiento de datos con Player, operación con joystick
- Sistema calibrado: \mathbf{K} , \mathbf{t}_r^c , \mathbf{R}_r^c

Resultados experimentales de UKF-SLAM (1)



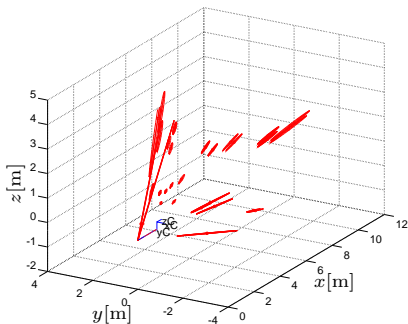
Paso temporal $k = 50$, avance $\sim 0,30m$ (1/4)

Resultados experimentales de UKF-SLAM (1)



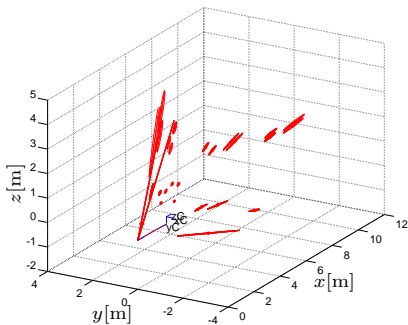
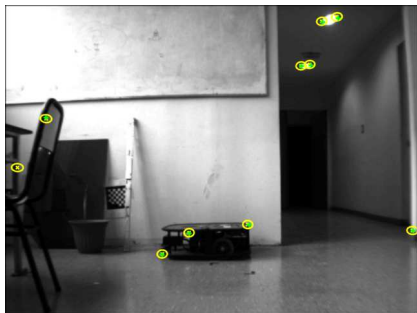
Paso temporal $k = 100$, avance $\sim 0,80m$ (2/4)

Resultados experimentales de UKF-SLAM (1)



Paso temporal $k = 150$, avance $\sim 1,40m$ (3/4)

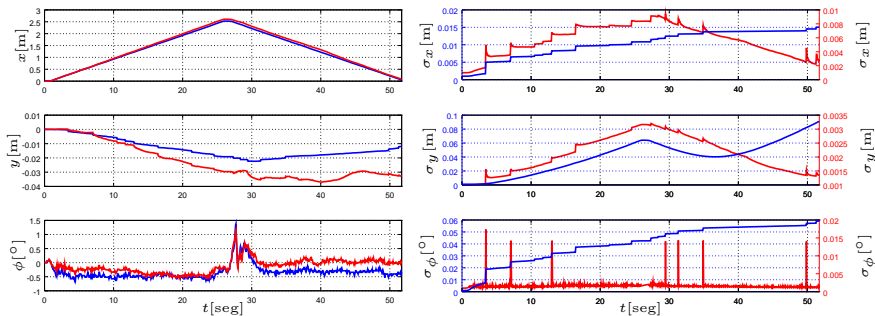
Resultados experimentales de UKF-SLAM (1)



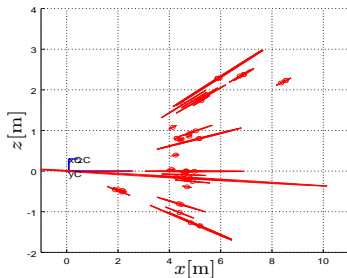
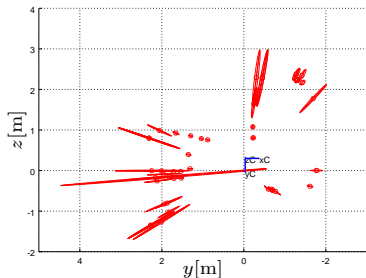
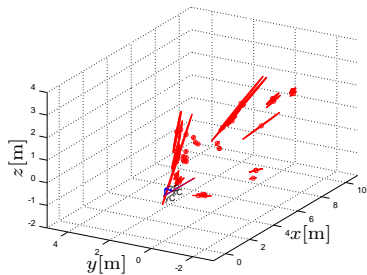
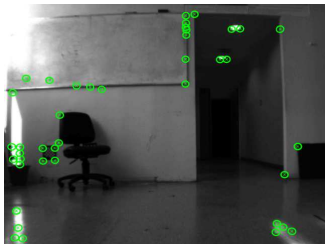
Paso temporal $k = 170$, avance $\sim 2,10m$ (4/4)

Resultados experimentales de UKF-SLAM (2)

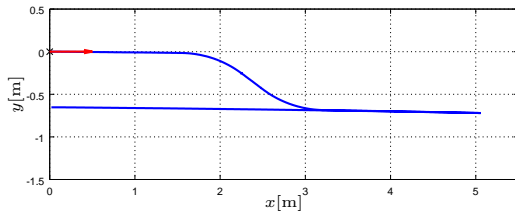
Estimación de la localización del robot
(Odometría - Estimación)



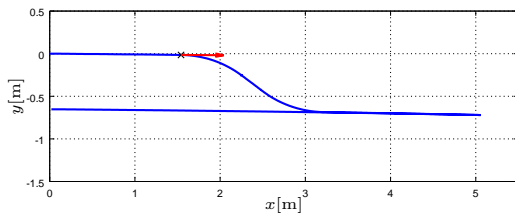
Resultados experimentales de UKF-SLAM (2)



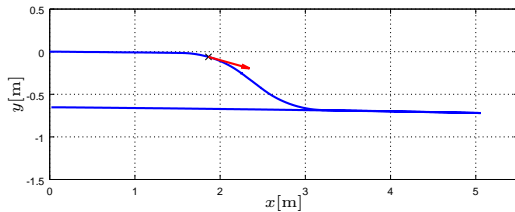
Resultados experimentales de UKF-SLAM para localización (1)



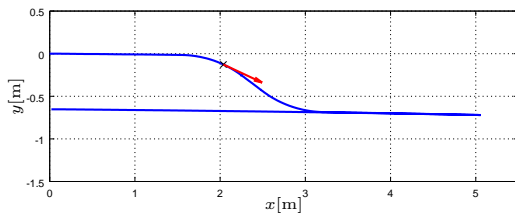
Resultados experimentales de UKF-SLAM para localización (1)



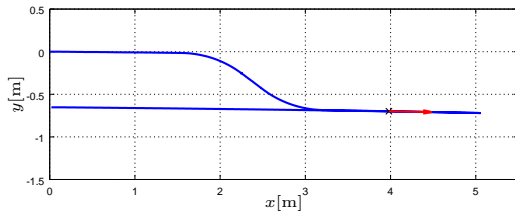
Resultados experimentales de UKF-SLAM para localización (1)



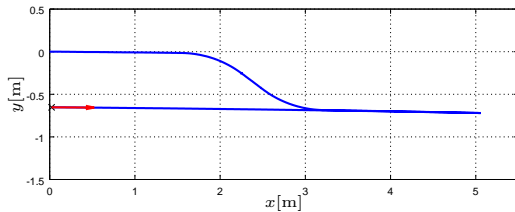
Resultados experimentales de UKF-SLAM para localización (1)



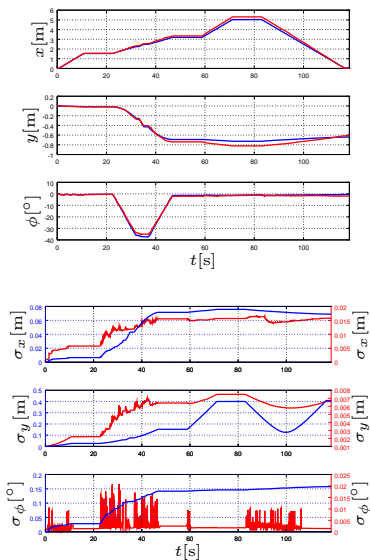
Resultados experimentales de UKF-SLAM para localización (1)



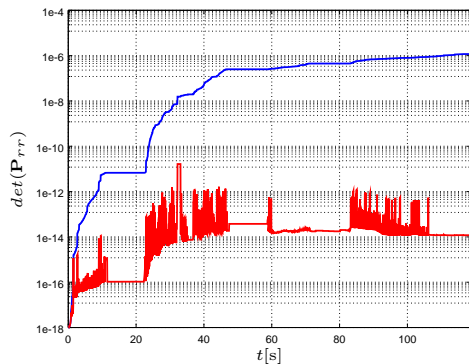
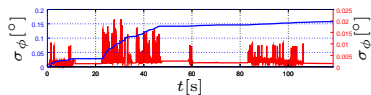
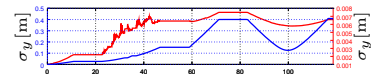
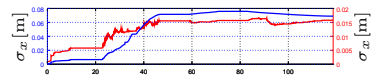
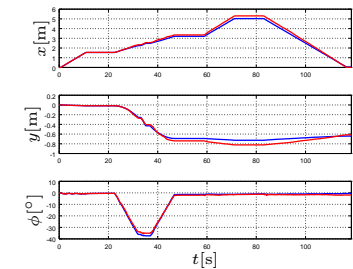
Resultados experimentales de UKF-SLAM para localización (1)



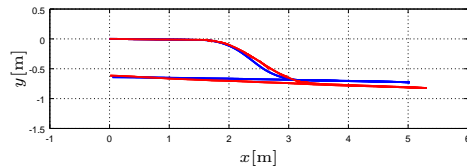
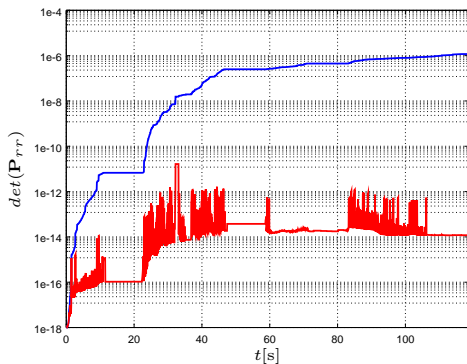
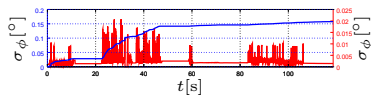
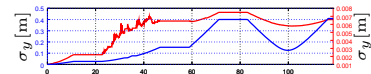
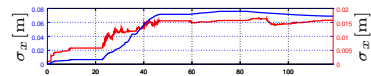
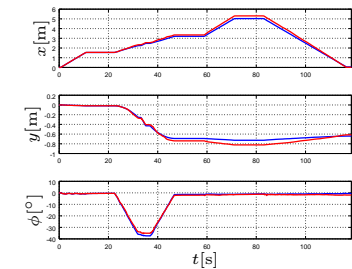
Resultados experimentales de UKF-SLAM para localización (1)



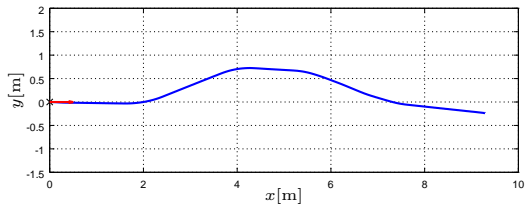
Resultados experimentales de UKF-SLAM para localización (1)



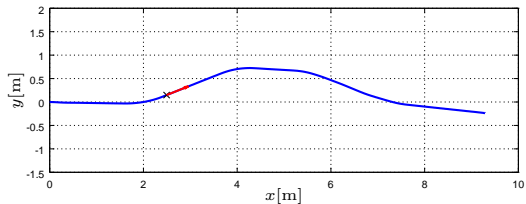
Resultados experimentales de UKF-SLAM para localización (1)



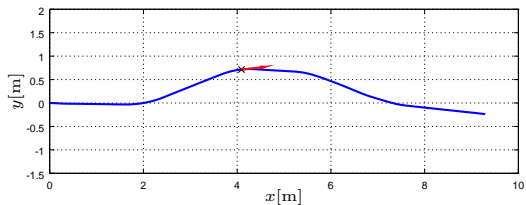
Resultados experimentales de UKF-SLAM para localización (2)



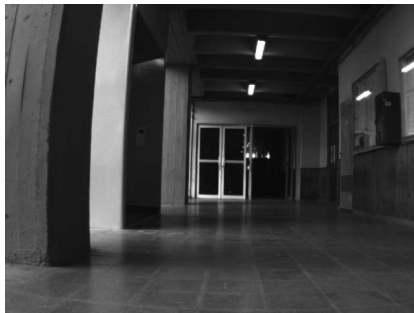
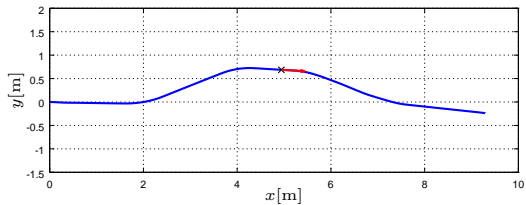
Resultados experimentales de UKF-SLAM para localización (2)



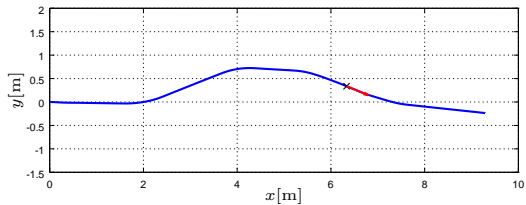
Resultados experimentales de UKF-SLAM para localización (2)



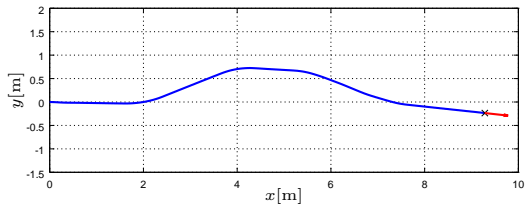
Resultados experimentales de UKF-SLAM para localización (2)



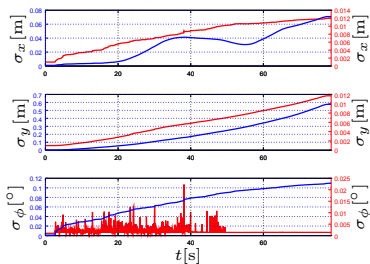
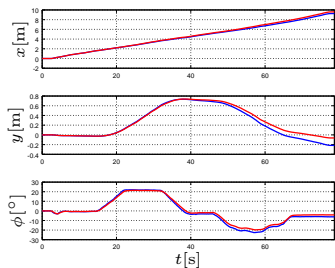
Resultados experimentales de UKF-SLAM para localización (2)



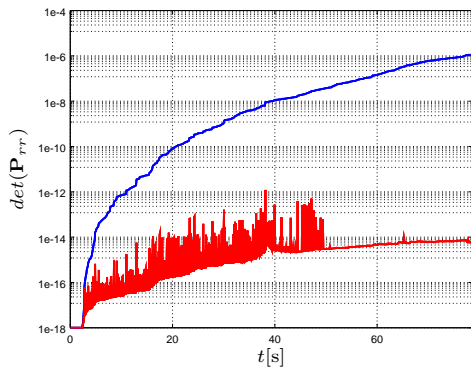
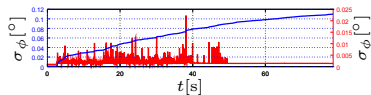
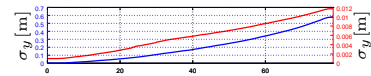
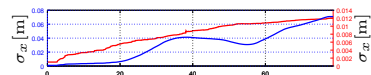
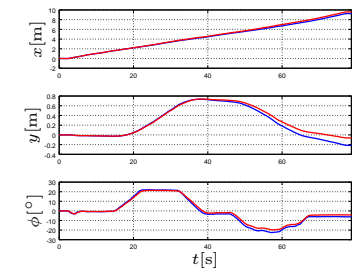
Resultados experimentales de UKF-SLAM para localización (2)



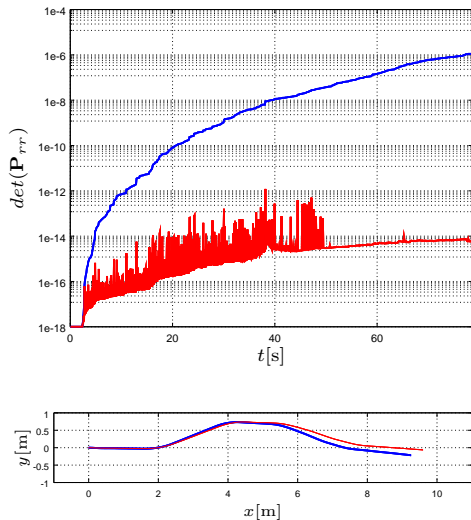
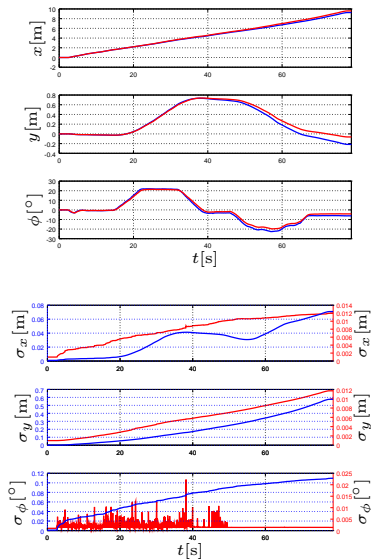
Resultados experimentales de UKF-SLAM para localización (2)



Resultados experimentales de UKF-SLAM para localización (2)



Resultados experimentales de UKF-SLAM para localización (2)



- 1 Introducción
- 2 Localización y Mapeo Simultáneos - SLAM
- 3 Robot Móvil RoMAA-II y visión robótica
- 4 SLAM con mapeo bidimensional
- 5 SLAM con mapeo tridimensional
- 6 Conclusiones y trabajos futuros

Conclusiones

Robot móvil RoMAA-II

Prestaciones similares a los robots disponibles comercialmente

Conclusiones

Robot móvil RoMAA-II

Prestaciones similares a los robots disponibles comercialmente

Calibración robot/cámara

Procedimiento simple y de fácil aplicación para obtener los parámetros extrínsecos

Conclusiones

Robot móvil RoMAA-II

Prestaciones similares a los robots disponibles comercialmente

Calibración robot/cámara

Procedimiento simple y de fácil aplicación para obtener los parámetros extrínsecos

SLAM con mapeo bidimensional

- Comportamiento similar entre filtro EKF y UKF
- Poca información sobre la estructura del entorno
- Los landmarks 2D pueden representar entidades complejas (troncos de árboles)
- En entornos de alta complejidad se necesitan algoritmos de percepción robustos

Conclusiones

Robot móvil RoMAA-II

Prestaciones similares a los robots disponibles comercialmente

Calibración robot/cámara

Procedimiento simple y de fácil aplicación para obtener los parámetros extrínsecos

SLAM con mapeo bidimensional

- Comportamiento similar entre filtro EKF y UKF
- Poca información sobre la estructura del entorno
- Los landmarks 2D pueden representar entidades complejas (troncos de árboles)
- En entornos de alta complejidad se necesitan algoritmos de percepción robustos

SLAM con mapeo tridimensional

- Mejor comportamiento del UKF comparado con el EKF
- Método sencillo y económico para localización de robot móvil
- La odometría permite ajustar la escala y mejorar la asociación de datos

Trabajos futuros

Líneas de investigación

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF
- Fusión con otros sensores como unidades inerciales

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF
- Fusión con otros sensores como unidades inerciales
- Calibración online de los parámetros extrínsecos

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF
- Fusión con otros sensores como unidades inerciales
- Calibración online de los parámetros extrínsecos
- Reconstrucción densa del entorno (métodos directos o semi-directos)

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF
- Fusión con otros sensores como unidades inerciales
- Calibración online de los parámetros extrínsecos
- Reconstrucción densa del entorno (métodos directos o semi-directos)
- Incorporación de información semántica en SLAM visual

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF
- Fusión con otros sensores como unidades inerciales
- Calibración online de los parámetros extrínsecos
- Reconstrucción densa del entorno (métodos directos o semi-directos)
- Incorporación de información semántica en SLAM visual

Aplicaciones

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF
- Fusión con otros sensores como unidades inerciales
- Calibración online de los parámetros extrínsecos
- Reconstrucción densa del entorno (métodos directos o semi-directos)
- Incorporación de información semántica en SLAM visual

Aplicaciones

- Robótica en agricultura (AGV, UAV)

Trabajos futuros

Líneas de investigación

- Extensión de los esquemas de SLAM visual a 6 DOF
- Fusión con otros sensores como unidades inerciales
- Calibración online de los parámetros extrínsecos
- Reconstrucción densa del entorno (métodos directos o semi-directos)
- Incorporación de información semántica en SLAM visual

Aplicaciones

- Robótica en agricultura (AGV, UAV)
- Robótica de asistencia a personas con discapacidades

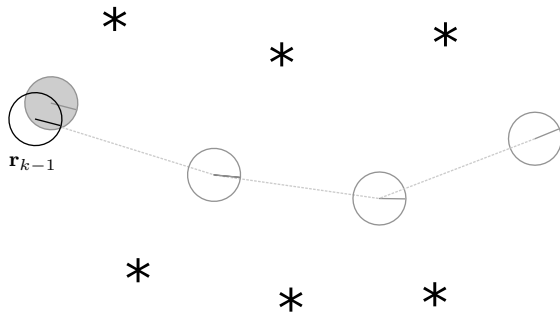
Navegación Autónoma de Robot Móvil en Ambientes Parcialmente Estructurados utilizando Visión Artificial

Gonzalo Fernando Perez Paina

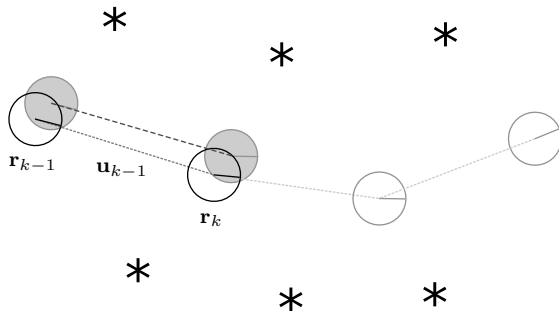
Muchas gracias!



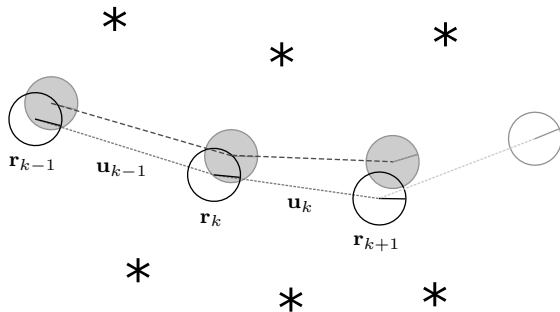
Ejemplo de SLAM de landmarks - Modelo de movimiento



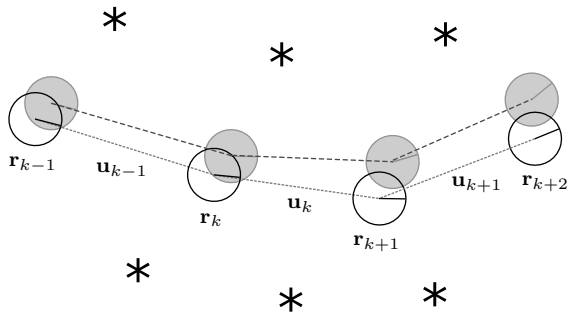
Ejemplo de SLAM de landmarks - Modelo de movimiento



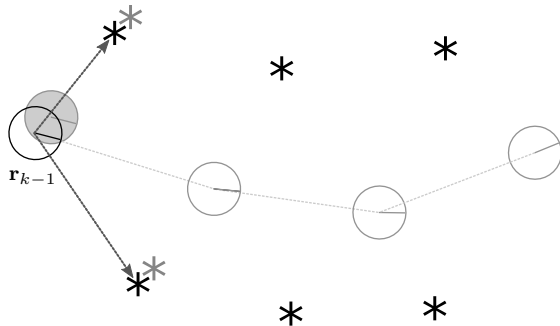
Ejemplo de SLAM de landmarks - Modelo de movimiento



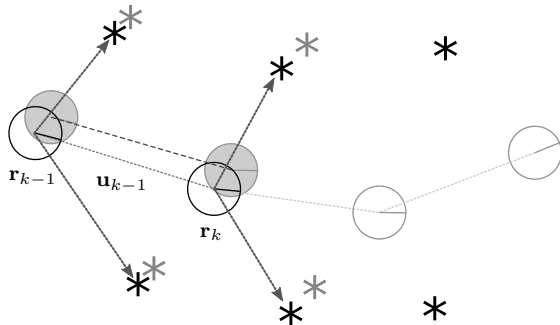
Ejemplo de SLAM de landmarks - Modelo de movimiento



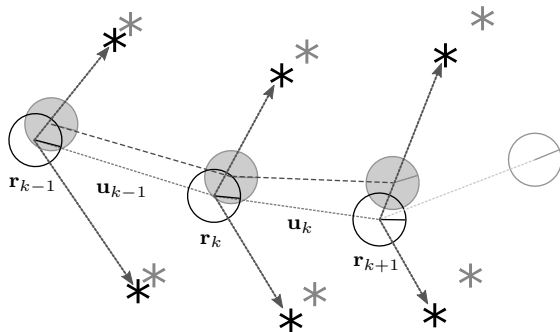
Ejemplo de SLAM de landmarks - Agregando mediciones



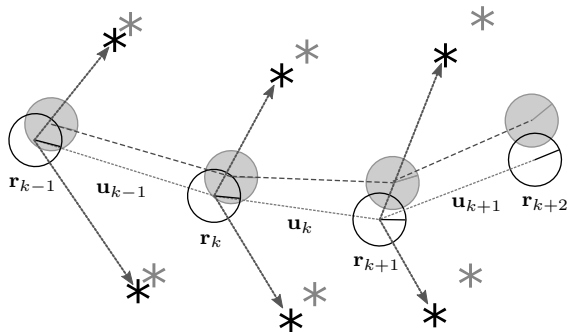
Ejemplo de SLAM de landmarks - Agregando mediciones



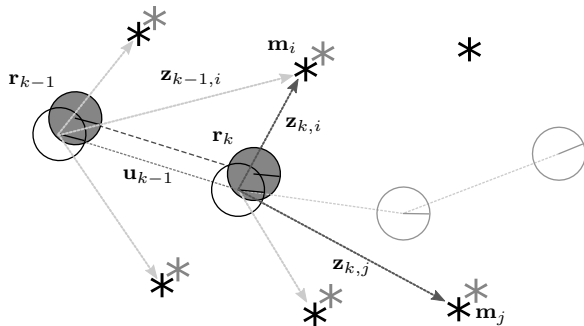
Ejemplo de SLAM de landmarks - Agregando mediciones



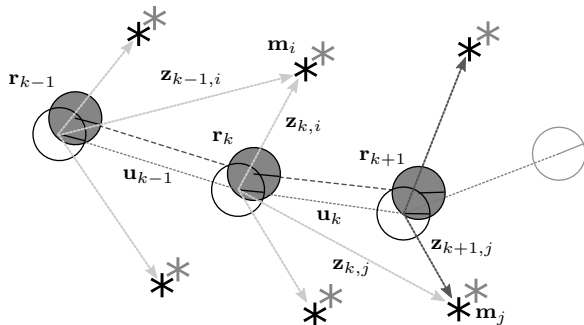
Ejemplo de SLAM de landmarks - Agregando mediciones



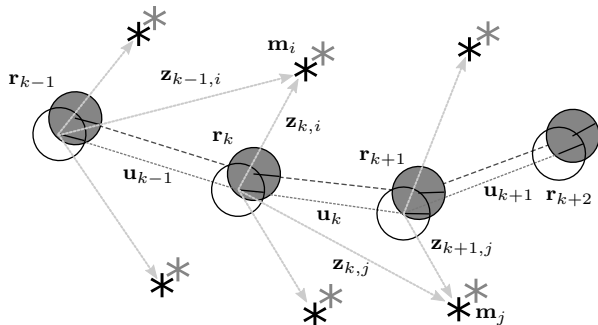
Ejemplo de SLAM de landmarks - Correlación



Ejemplo de SLAM de landmarks - Correlación



Ejemplo de SLAM de landmarks - Correlación



KF-SLAM - Proceso de estimación

1) Predicción - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k^-, \mathbf{P}_k^-$ - Se utiliza el **MODELO DE MOVIMIENTO** $\mathbf{f}_r(\cdot)$

5) Corrección - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k, \mathbf{P}_k$

KF-SLAM - Proceso de estimación

1) Predicción - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k^-, \mathbf{P}_k^-$ - Se utiliza el **MODELO DE MOVIMIENTO** $\mathbf{f}_r(\cdot)$

2) Predicción de la observación

Se utilizando el **MODELO DE OBSERVACIÓN** $\mathbf{h}(\cdot)$

5) Corrección - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k, \mathbf{P}_k$

KF-SLAM - Proceso de estimación

1) Predicción - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k^-, \mathbf{P}_k^-$ - Se utiliza el **MODELO DE MOVIMIENTO** $\mathbf{f}_r(\cdot)$

2) Predicción de la observación

Se utilizando el **MODELO DE OBSERVACIÓN** $\mathbf{h}(\cdot)$

3) Observación

- Utiliza los sensores a bordo del robot
- Proceso de medición y extracción de características

5) Corrección - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k, \mathbf{P}_k$

KF-SLAM - Proceso de estimación

1) Predicción - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k^-, \mathbf{P}_k^-$ - Se utiliza el **MODELO DE MOVIMIENTO** $\mathbf{f}_r(\cdot)$

2) Predicción de la observación

Se utilizando el **MODELO DE OBSERVACIÓN** $\mathbf{h}(\cdot)$

3) Observación

- Utiliza los sensores a bordo del robot
- Proceso de medición y extracción de características

4) Asociación de datos

- Correspondencia entre observación y predicción
- Depende del tipo de sensor utilizado

5) Corrección - Filtro de Kalman

Se tiene $\hat{\mathbf{x}}_k, \mathbf{P}_k$

Transformada Unscented, UT

Dada una v.a. $\mathbf{x} \in \mathbb{R}^L$, con $\mathbf{x} \sim \mathcal{N}(\bar{\mathbf{x}}, \mathbf{P}_x)$, y la v.a. \mathbf{y} dada por $\mathbf{y} = g(\mathbf{x})$.

- 1 Conjunto de puntos sigma
- 2 Propagación de puntos sigma
- 3 Estimación de media y covarianza

Transformada Unscented, UT

Dada una v.a. $\mathbf{x} \in \mathbb{R}^L$, con $\mathbf{x} \sim \mathcal{N}(\bar{\mathbf{x}}, \mathbf{P}_x)$, y la v.a. \mathbf{y} dada por $\mathbf{y} = g(\mathbf{x})$.

- 1 Conjunto de puntos sigma
- 2 Propagación de puntos sigma
- 3 Estimación de media y covarianza

Formar un conjunto con $2L + 1$ puntos sigma

$$\mathcal{X}_0 = \bar{\mathbf{x}}$$

$$\mathcal{X}_i = \bar{\mathbf{x}} + \left(\sqrt{(L + \lambda)\mathbf{P}_x} \right)_i \quad i = 1, \dots, L$$

$$\mathcal{X}_i = \bar{\mathbf{x}} - \left(\sqrt{(L + \lambda)\mathbf{P}_x} \right)_{i-L} \quad i = L + 1, \dots, 2L$$

donde $\lambda = \sigma^2(L + \kappa) - L$ es un parámetro de escalado

Transformada Unscented, UT

Dada una v.a. $\mathbf{x} \in \mathbb{R}^L$, con $\mathbf{x} \sim \mathcal{N}(\bar{\mathbf{x}}, \mathbf{P}_x)$, y la v.a. \mathbf{y} dada por $\mathbf{y} = g(\mathbf{x})$.

- 1 Conjunto de puntos sigma
- 2 Propagación de puntos sigma
- 3 Estimación de media y covarianza

$$\mathcal{Y}_i = \mathbf{g}(\mathcal{X}_i), \quad i = 0, \dots, 2L.$$

Transformada Unscented, UT

Dada una v.a. $\mathbf{x} \in \mathbb{R}^L$, con $\mathbf{x} \sim \mathcal{N}(\bar{\mathbf{x}}, \mathbf{P}_x)$, y la v.a. \mathbf{y} dada por $\mathbf{y} = g(\mathbf{x})$.

- 1 Conjunto de puntos sigma
- 2 Propagación de puntos sigma
- 3 **Estimación de media y covarianza**

$$\bar{\mathbf{y}} = \mathbb{E}[g(\mathbf{x})] \approx \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}^{(i)}$$

$$\mathbf{P}_y = Cov[g(\mathbf{x})] \approx \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{Y}^{(i)} - \bar{\mathbf{y}})(\mathcal{Y}^{(i)} - \bar{\mathbf{y}})^T$$

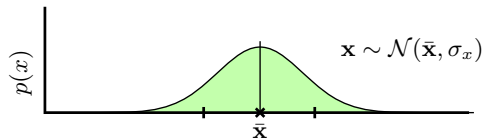
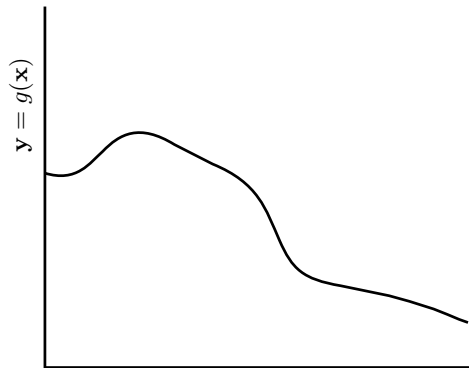
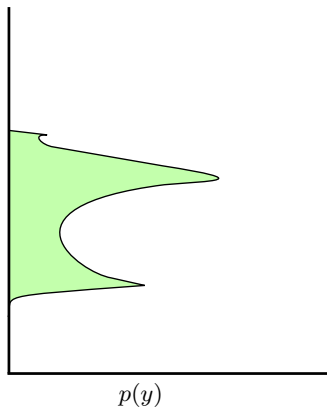
$$\mathbf{P}_{xy} \approx \sum_{i=0}^{2L} W_i^{(c)} (\mathcal{X}^{(i)} - \bar{\mathbf{x}})(\mathcal{Y}^{(i)} - \bar{\mathbf{y}})^T$$

donde las constantes de peso $W_i^{(m)}$ y $W_i^{(c)}$ están dadas por

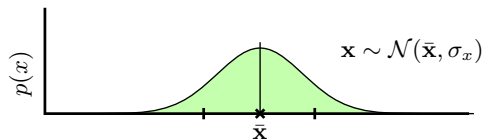
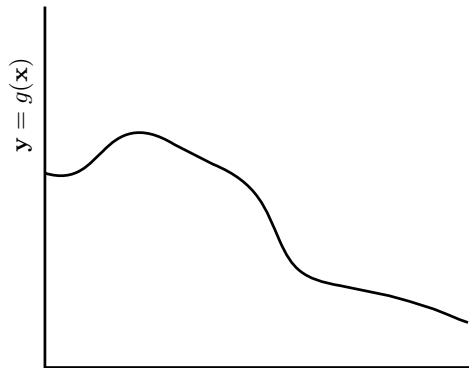
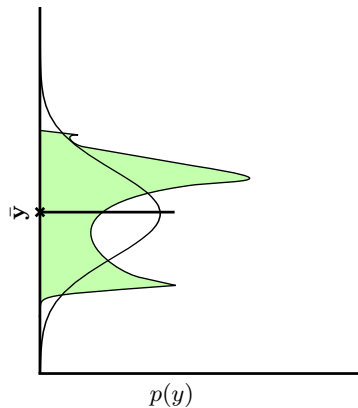
$$W_0^{(m)} = \lambda / (L + \lambda), \quad W_0^{(c)} = \lambda / (L + \lambda) + (1 - \alpha^2 + \beta)$$

$$W_i^{(m)} = W_i^{(c)} = 1 / (2(L + \lambda)), \quad i = 1, \dots, 2L$$

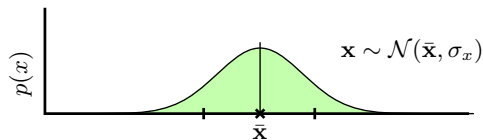
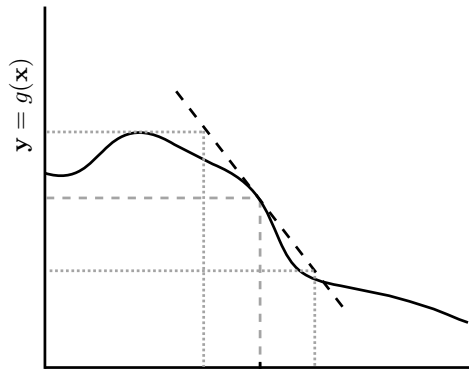
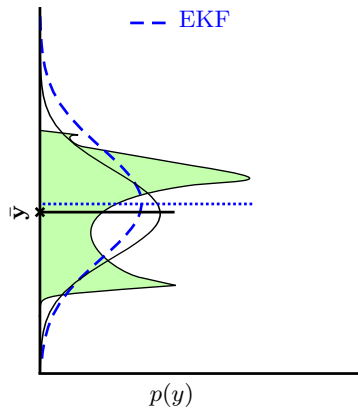
Filtro de Kalman. EKF y UKF



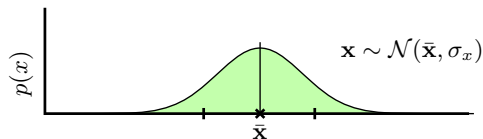
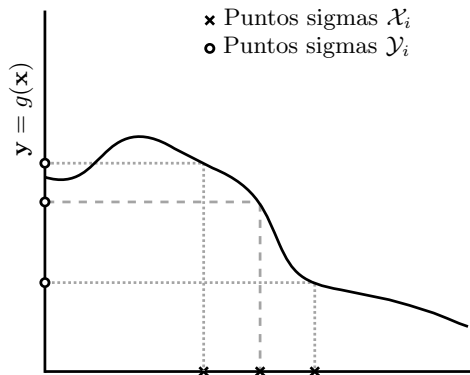
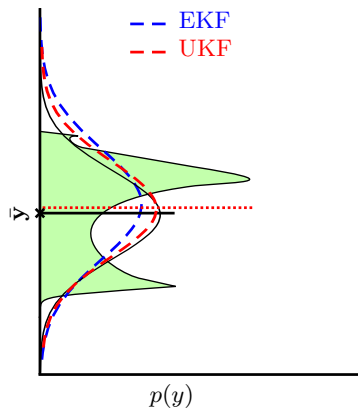
Filtro de Kalman. EKF y UKF



Filtro de Kalman. EKF y UKF



Filtro de Kalman. EKF y UKF



Filtro de Kalman Extendido, EKF

Inicialización

$$\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0], \quad \mathbf{P}_0 = \mathbb{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T]$$

Filtro de Kalman Extendido, EKF

Inicialización

$$\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0], \quad \mathbf{P}_0 = \mathbb{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T]$$

Predicción

$$\hat{\mathbf{x}}_k^- = \mathbf{f}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1})$$

$$\mathbf{P}_k^- = \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k$$

$$\mathbf{F}_k = \left. \frac{\partial f(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k-1}}, \quad \mathbf{H}_k = \left. \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_k^-}$$

Corrección

$$\mathbf{z}_k^- = \mathbf{h}(\hat{\mathbf{x}}_k^-)$$

$$\mathbf{R}_k^- = \mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T$$

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{R}_k^- + \mathbf{R}_k)^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \hat{\mathbf{z}}_k^-)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-$$

Filtro de Kalman Unscented, UKF

Inicialización

$$\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0], \quad \mathbf{P}_0 = \mathbb{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T]$$

Predicción

$$\hat{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1} \xrightarrow{\text{UT}[\mathbf{f}(\cdot)]} \hat{\mathbf{x}}_k^-, \mathbf{P}_k^-$$

Corrección

$$\hat{\mathbf{x}}_k^-, \mathbf{P}_k^- \xrightarrow{\text{UT}[\mathbf{h}(\cdot)]} \hat{\mathbf{z}}_k^-, \mathbf{R}_k^-$$

$$\mathbf{K}_k = \mathbf{P}_{xz,k} (\mathbf{R}_k + \mathbf{R}_k^-)^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \hat{\mathbf{z}}_k^-)$$

$$\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{K}_k (\mathbf{R}_k + \mathbf{R}_k^-) \mathbf{K}_k^T$$

EKF-SLAM–Predicción

Dado el vector de estado del SLAM estimado a paso temporal $k - 1$

$$\hat{\mathbf{x}}_k^- = \begin{bmatrix} \hat{\mathbf{r}}_{k-1} \\ \hat{\mathbf{m}}_{k-1} \end{bmatrix}$$

la **PREDICCIÓN** del estado actual (asumiendo el mapa estático) de viene dada por

$$\hat{\mathbf{x}}_k^- = \begin{bmatrix} \hat{\mathbf{r}}_k^- \\ \hat{\mathbf{m}}_k^- \end{bmatrix} = \begin{bmatrix} \mathbf{f}_r(\hat{\mathbf{r}}_{k-1}, \mathbf{u}_{k-1}) \\ \hat{\mathbf{m}}_{k-1} \end{bmatrix}.$$

La etapa de predicción del EKF se aplica en base a las siguiente matrices

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{F}_{r,k} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \mathbf{Q}_k = \begin{bmatrix} \mathbf{F}_{u,k} \mathbf{P}_u \mathbf{F}_{u,k} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

donde

- \mathbf{P}_u es la incertidumbre de la acción de control, y
- \mathbf{F}_u y \mathbf{F}_r son los jacobianos de la función de movimiento.

$$\mathbf{F}_r = \frac{\partial \mathbf{f}_r(\mathbf{x}, \mathbf{u})}{\partial \mathbf{r}}, \quad \mathbf{F}_u = \frac{\partial \mathbf{f}_r(\mathbf{x}, \mathbf{u})}{\partial \mathbf{u}}$$

EKF-SLAM-Corrección

En la etapa de **CORRECCIÓN** se utiliza

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}_{r,k}^1 & \mathbf{H}_{m,k}^1 & \cdots & \mathbf{0} \\ \mathbf{H}_{r,k}^2 & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_{r,k}^N & \mathbf{0} & \cdots & \mathbf{H}_{m,k}^N \end{bmatrix}, \quad \mathbf{R}_k = \mathbf{J}_s \mathbf{P}_m \mathbf{J}_s$$

donde

- \mathbf{P}_m es el ruido de medición, y
- \mathbf{J}_s el jacobiano de sensado que proyecta la incertidumbre de las mediciones al espacio de características utilizado.

$$\mathbf{H}_r = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{r}}, \quad \mathbf{H}_m^i = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{m}^i}$$

EKF-SLAM–Agregado de landmarks

La matriz de covarianza que representa la incertidumbre del nuevo landmark viene dado por

$$\mathbf{P}_{yy} = \mathbf{F}_r \mathbf{P}_{rr} \mathbf{F}_r^T + \mathbf{F}_z \mathbf{R} \mathbf{F}_z^T,$$

donde

- \mathbf{R} es la incertidumbre de la nueva características (en el espacio de observación), y
- \mathbf{F}_z es el jacobiano de la función de inicialización utilizada que proyecta dicha incertidumbre al sistema de referencia global del SLAM.

La covarianza cruzada entre el nuevo landmark y las demás entidades existentes en el vector del SLAM es

$$\mathbf{P}_{yx} = \mathbf{F}_k \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm} \end{bmatrix} = \mathbf{F}_k \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \cdots & \mathbf{P}_{rm_N} \end{bmatrix}.$$

Inicialización de landmarks

Vector de estado y matriz de covarianza previo a agregar el nuevo landmark

$$\hat{\mathbf{x}} = [\hat{\mathbf{r}}^T \quad \hat{\mathbf{m}}_1^T \quad \dots \quad \hat{\mathbf{m}}_N^T]^T$$
$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \dots & \mathbf{P}_{rm_N} \\ \mathbf{P}_{m_1r} & \mathbf{P}_{m_1m_1} & \dots & \mathbf{P}_{m_1m_N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m_Nr} & \mathbf{P}_{m_Nm_1} & \dots & \mathbf{P}_{m_Nm_N} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm} \\ \mathbf{P}_{mr} & \mathbf{P}_{mm} \end{bmatrix}$$

Inicialización de landmarks

Vector de estado y matriz de covarianza previo a agregar el nuevo landmark

$$\hat{\mathbf{x}} = [\hat{\mathbf{r}}^T \quad \hat{\mathbf{m}}_1^T \quad \dots \quad \hat{\mathbf{m}}_N^T]^T$$

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \dots & \mathbf{P}_{rm_N} \\ \mathbf{P}_{m_1r} & \mathbf{P}_{m_1m_1} & \dots & \mathbf{P}_{m_1m_N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m_Nr} & \mathbf{P}_{m_Nm_1} & \dots & \mathbf{P}_{m_Nm_N} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm} \\ \mathbf{P}_{mr} & \mathbf{P}_{mm} \end{bmatrix}$$

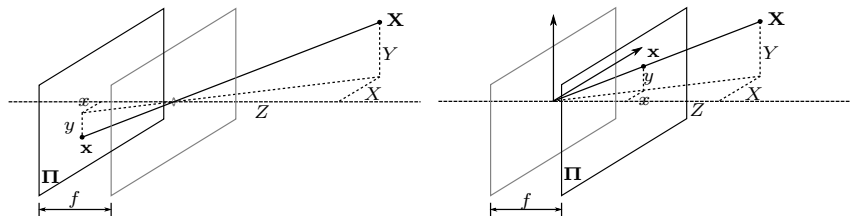
Luego de la inicialización utilizando $\mathbf{y} = \mathbf{y}(\mathbf{x}, \mathbf{z})$, se tiene

$$\hat{\mathbf{x}} = [\hat{\mathbf{r}}^T \quad \hat{\mathbf{m}}_1^T \quad \dots \quad \hat{\mathbf{m}}_N^T \quad \hat{\mathbf{y}}^T]^T$$

$$\mathbf{P}_{\text{NEW}} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rm_1} & \dots & \mathbf{P}_{rm_N} & \mathbf{P}_{ry} \\ \mathbf{P}_{m_1r} & \mathbf{P}_{m_1m_1} & \dots & \mathbf{P}_{m_1m_N} & \mathbf{P}_{m_1y} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{P}_{m_Nr} & \mathbf{P}_{m_Nm_1} & \dots & \mathbf{P}_{m_Nm_N} & \mathbf{P}_{m_Ny} \\ \mathbf{P}_{yr} & \mathbf{P}_{ym_1} & \dots & \mathbf{P}_{ym_N} & \mathbf{P}_{yy} \end{bmatrix} = \begin{bmatrix} \mathbf{P} & \mathbf{P}_{yx}^T \\ \mathbf{P}_{yx} & \mathbf{P}_{yy} \end{bmatrix}$$

donde $\mathbf{P}_{yx} = [\mathbf{P}_{yr} \quad \mathbf{P}_{ym_1} \quad \dots \quad \mathbf{P}_{ym_N}]$.

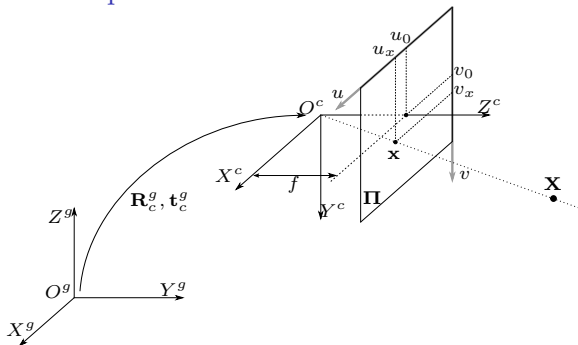
Modelo pinhole de cámaras



Por triángulos semejantes

$$x = f \frac{X}{Z}, \quad y = f \frac{Y}{Z}, \quad z = f,$$

Modelo pinhole de cámaras - Parámetros intrínsecos y extrínsecos



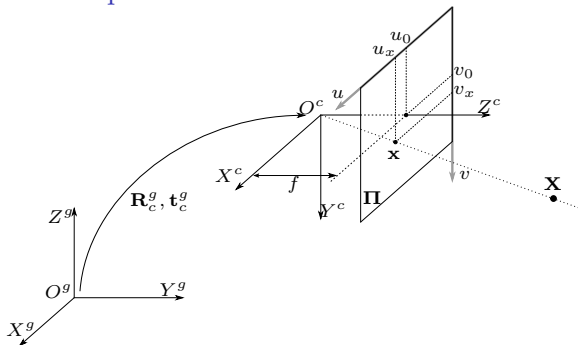
- SCG: $\langle O^g, X^g, Y^g, Z^g \rangle$

- SCC: $\langle O^c, X^c, Y^c, Z^c \rangle$

$$P = KR [I \mid -C]$$

$$K = \begin{bmatrix} f/h_u & s & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

Modelo pinhole de cámaras - Parámetros intrínsecos y extrínsecos



- SCG: $\langle O^g, X^g, Y^g, Z^g \rangle$

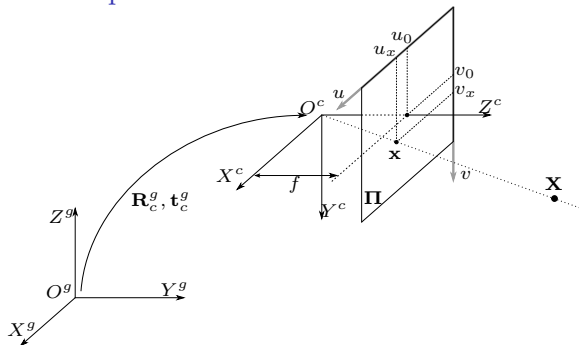
- SCC: $\langle O^c, X^c, Y^c, Z^c \rangle$

$$\mathbf{P} = \mathbf{KR} [\mathbf{I} \mid -\mathbf{C}]$$

$$\mathbf{K} = \begin{bmatrix} f/h_u & s & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\lambda \mathbf{x} = \mathbf{PX} = \mathbf{K} [\mathbf{R} \mid -\mathbf{RC}] \begin{bmatrix} \mathbf{X}^g \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} \mathbf{X}^c \\ 1 \end{bmatrix}, \quad \mathbf{X} \in \mathbb{P}^3, \quad \mathbf{x} \in \mathbb{P}^2$$

Modelo pinhole de cámaras - Parámetros intrínsecos y extrínsecos



- SCG: $\langle O^g, X^g, Y^g, Z^g \rangle$
- SCC: $\langle O^c, X^c, Y^c, Z^c \rangle$

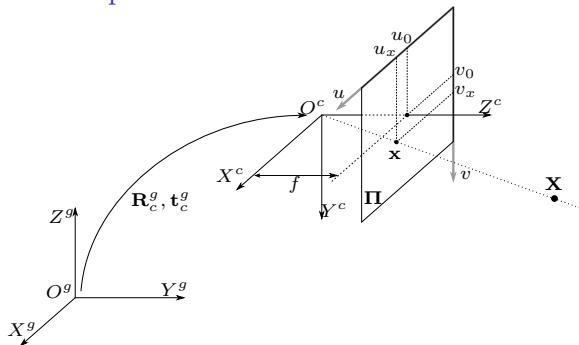
$$\mathbf{P} = \mathbf{K}\mathbf{R} \left[\mathbf{I} \mid -\mathbf{C} \right]$$

$$\mathbf{K} = \begin{bmatrix} f/h_u & s & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\lambda \mathbf{x} = \mathbf{P}\mathbf{X} = \mathbf{K} \left[\mathbf{R} \mid -\mathbf{RC} \right] \begin{bmatrix} \mathbf{X}^g \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} \mathbf{X}^c \\ 1 \end{bmatrix}, \quad \mathbf{X} \in \mathbb{P}^3, \quad \mathbf{x} \in \mathbb{P}^2$$

$$\lambda \mathbf{x} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^c \\ y^c \\ z^c \end{bmatrix} = \begin{bmatrix} f_u x^c + u_0 z^c \\ f_v y^c + v_0 z^c \\ z^c \end{bmatrix},$$

Modelo pinhole de cámaras - Parámetros intrínsecos y extrínsecos



- SCG: $\langle O^g, X^g, Y^g, Z^g \rangle$
- SCC: $\langle O^c, X^c, Y^c, Z^c \rangle$

$$\mathbf{P} = \mathbf{K} \mathbf{R} [\mathbf{I} \mid -\mathbf{C}]$$

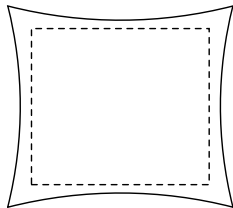
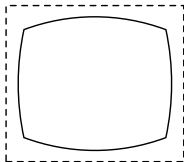
$$\mathbf{K} = \begin{bmatrix} f/h_u & s & u_0 \\ 0 & f/h_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\lambda \mathbf{x} = \mathbf{P} \mathbf{X} = \mathbf{K} [\mathbf{R} \mid -\mathbf{R}\mathbf{C}] \begin{bmatrix} \mathbf{X}^g \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} \mathbf{X}^c \\ 1 \end{bmatrix}, \quad \mathbf{X} \in \mathbb{P}^3, \quad \mathbf{x} \in \mathbb{P}^2$$

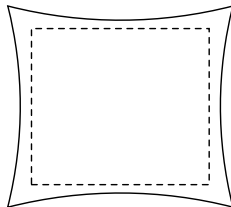
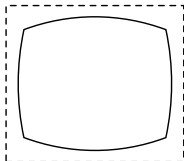
$$\lambda \mathbf{x} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^c \\ y^c \\ z^c \end{bmatrix} = \begin{bmatrix} f_u x^c + u_0 z^c \\ f_v y^c + v_0 z^c \\ z^c \end{bmatrix}, \quad \begin{aligned} u_x &= \frac{f_u x^c + u_0 z^c}{z^c} = f_u \frac{x^c}{z^c} + u_0 \\ v_x &= \frac{f_v y^c + v_0 z^c}{z^c} = f_v \frac{y^c}{z^c} + v_0 \end{aligned}$$

donde $\mathbf{m}_x = [u_x \quad v_x]^T \in \mathcal{I} \subset \mathbb{R}^2$.

Lentes - Distorsión



Lentes - Distorsión



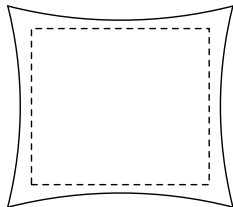
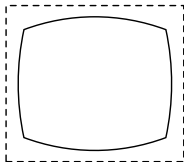
- (x_d^c, y_d^c) : punto distorsionado (SCC)
- (x^c, y^c) : punto sin distorsión (SCC)

$$x^c = x_d^c (1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + \kappa_3 r_d^6)$$

$$y^c = y_d^c (1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + \kappa_3 r_d^6)$$

$$\text{con } r_d = \sqrt{x_d^{c2} + y_d^{c2}}.$$

Lentes - Distorsión



- (x_d^c, y_d^c) : punto distorsionado (SCC)
- (x^c, y^c) : punto sin distorsión (SCC)

$$x^c = x_d^c (1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + \kappa_3 r_d^6)$$

$$y^c = y_d^c (1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + \kappa_3 r_d^6)$$

$$\text{con } r_d = \sqrt{x_d^{c2} + y_d^{c2}}.$$

En coordenadas de la imagen

$$u = u_0 + (u_d - u_0) (1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + \kappa_3 r_d^6)$$

$$v = v_0 + (v_d - v_0) (1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + \kappa_3 r_d^6)$$

donde

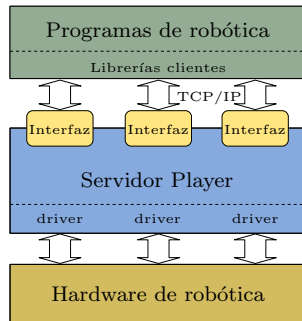
$$r_d = \sqrt{(h_u(u_d - u_0))^2 + (h_v(v_d - v_0))^2}.$$

κ_1 , κ_2 y κ_3 son los parámetros de distorsión

Entorno de desarrollo de robótica Player/Stage

Software de la PC de a bordo

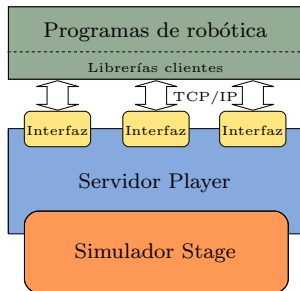
- Clase de comunicación (C++)
 - ▶ Implementa métodos análogos a c/u de los comandos de bajo nivel
- Driver para Player
 - ▶ Permite utilizar el robot de forma abstracta en redes TCP/IP



Entorno de desarrollo de robótica Player/Stage

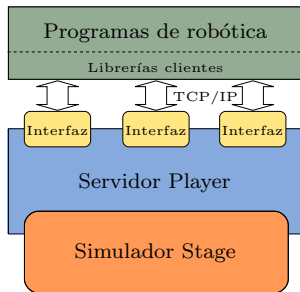
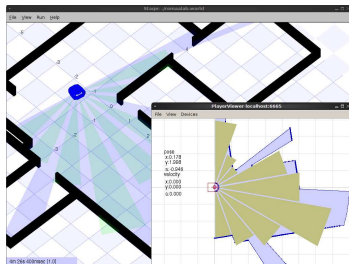
Software de la PC de a bordo

- Clase de comunicación (C++)
 - ▶ Implementa métodos análogos a c/u de los comandos de bajo nivel
- Driver para Player
 - ▶ Permite utilizar el robot de forma abstracta en redes TCP/IP



Entorno de desarrollo de robótica Player/Stage

GUI del simulador Stage



SLAM2D–Sensor de barrido láser

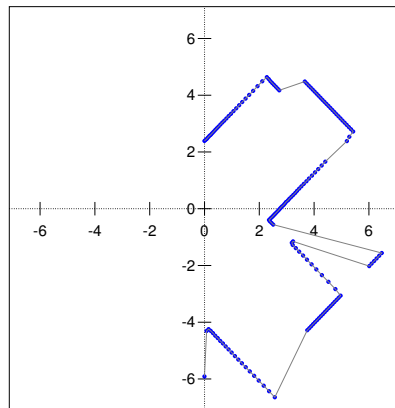
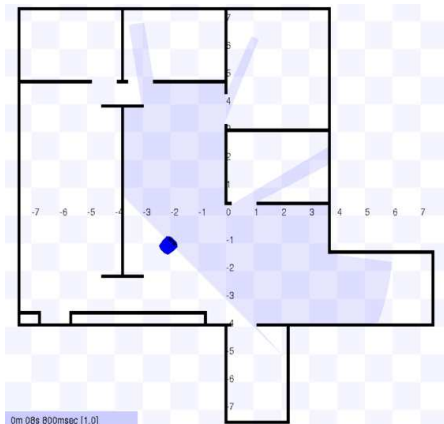
Conjunto de puntos $\mathcal{L} = \{(\rho_i, \zeta_i) | i = 1, \dots, N_l\}$

- Resolución angular fija $\zeta_{i-1} - \zeta_i = \Delta\xi$
- Secuencia ordenada: desde $\zeta_{min} = \zeta_1$ hasta $\zeta_{max} = \zeta_{N_l}$

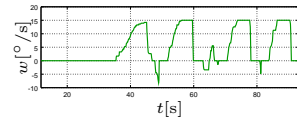
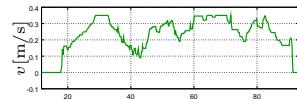
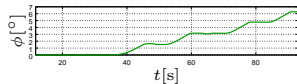
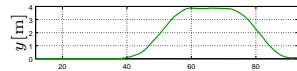
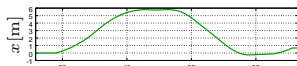
SLAM2D–Sensor de barrido láser

Conjunto de puntos $\mathcal{L} = \{(\rho_i, \zeta_i) | i = 1, \dots, N_l\}$

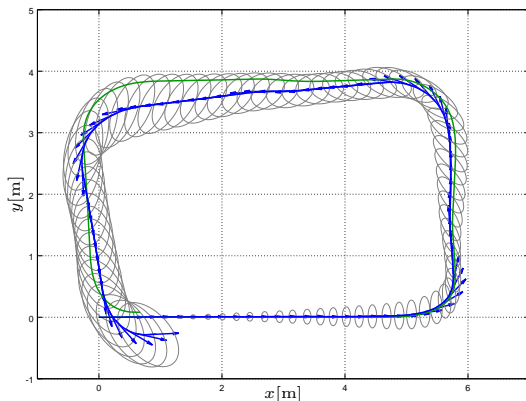
- Resolución angular fija $\zeta_{i-1} - \zeta_i = \Delta\xi$
- Secuencia ordenada: desde $\zeta_{min} = \zeta_1$ hasta $\zeta_{max} = \zeta_{N_l}$



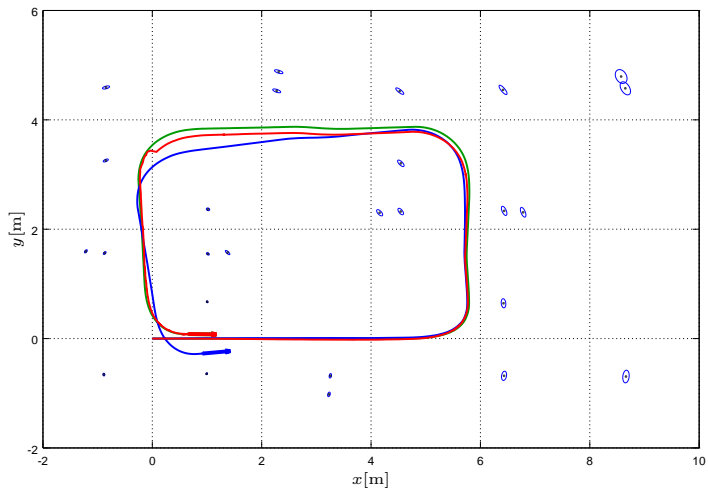
Resultados de simulación de SLAM2D (1)



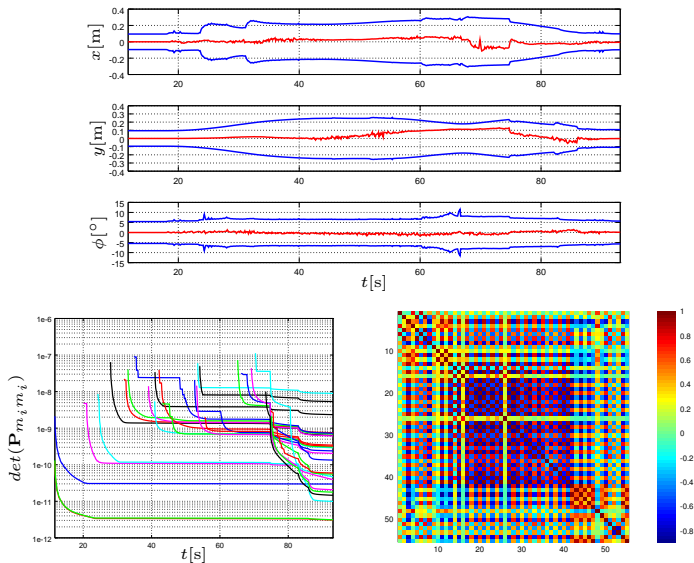
- $\alpha_1 = 0,02^\circ/\text{s}$, $\alpha_2 = 0,04^\circ/\text{m}$, $\alpha_3 = 0,05\text{m}/\text{m}$ y $\alpha_4 = 0,05\text{m}/\text{s}$
- $\sigma_{\rho_i} = 5\%(\rho_i/\rho_{max})$ y $\sigma_\zeta = 10\%\Delta\zeta$, con $\rho_{max} = 8,0\text{m}$



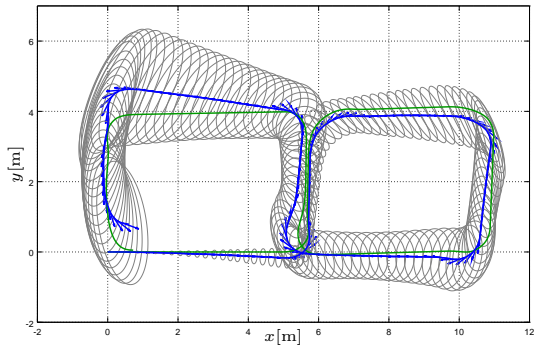
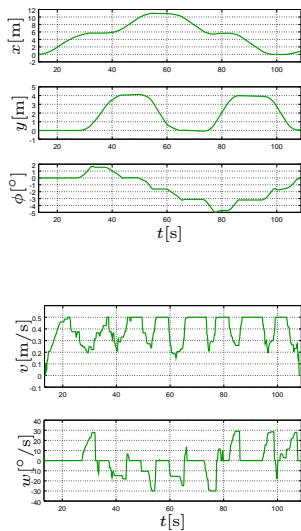
Resultados de simulación de SLAM2D (1)



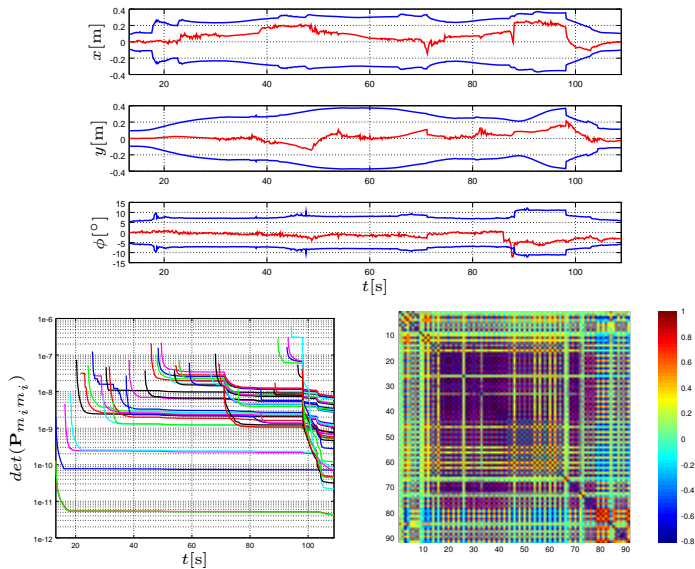
Resultados de simulación de SLAM2D (1)



Resultados de simulación de SLAM2D (1)



Resultados de simulación de SLAM2D (1)



SLAM2D–Filtro de Información Extendido (EIF)

Parámetros canónicos de una distribución gaussiana multivariada

- $\psi = \Omega \mathbf{x}$: vector de información
- $\Omega = \mathbf{P}^{-1}$: matriz de información

Predicción

$$\begin{aligned}\hat{\mathbf{x}}_{k-1} &= \Omega_{k-1}^{-1} \xi_{k-1} \\ \Omega_k^- &= \left(\mathbf{F}_k \Omega_{k-1}^{-1} \mathbf{F}_k^T + \mathbf{Q}_k \right)^{-1} \\ \hat{\mathbf{x}}_k^- &= \mathbf{f}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}) \\ \xi_k^- &= \Omega_k^- \hat{\mathbf{x}}_k^-, \end{aligned}$$

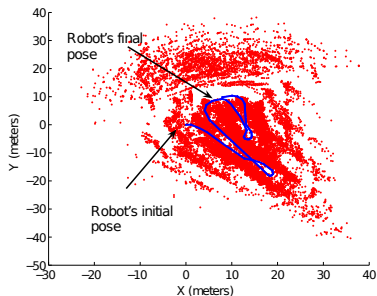
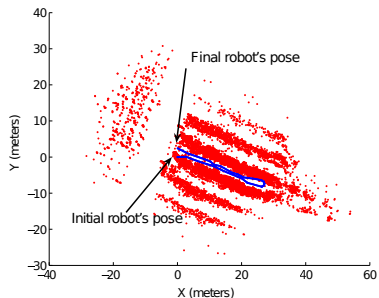
Corrección

$$\begin{aligned}\Omega_k &= \Omega_k^- + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k \\ \xi_k &= \xi_k^- + \mathbf{H}_k^T \mathbf{R}_k^{-1} \left(\mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-) + \mathbf{H}_k \hat{\mathbf{x}}_k^- \right).\end{aligned}$$

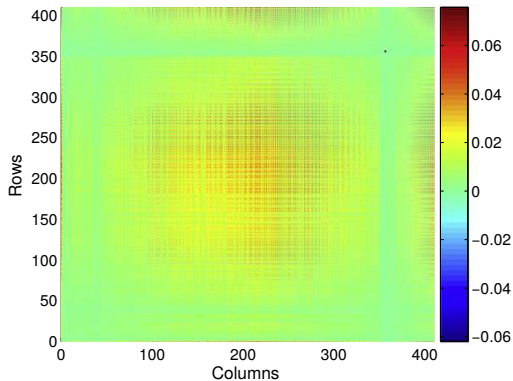
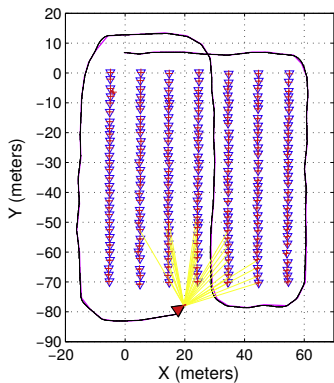
SLAM2D–Criterio de selección de características

- 1: Sea \mathcal{C}_k el conjunto de características detectadas al paso temporal k
- 2: Sea $\mathcal{C}_k^a \subseteq \mathcal{C}_k$ el conjunto de caract. con asociación correcta a paso temporal k
- 3: Sea LIM el número de características a ser utilizadas en la etapa de corrección
- 4: $\bar{\Omega}_k \leftarrow \Omega_k^-$
- 5: $\bar{\xi}_k \leftarrow \xi_k^-$
- 6: **for** $j \leftarrow 1, \dots, \min(LIM, \#\mathcal{C}_k^a)$ **do**
- 7: $\mathbf{z}_j^{opt} \leftarrow \arg_z \max(|\bar{\Omega}_k|)$
- 8: $\Omega_k \leftarrow \bar{\Omega}_k + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k$
- 9: $\xi_k \leftarrow \bar{\xi}_k + \mathbf{H}_k^T \mathbf{R}_k^{-1} (\mathbf{z}_j^{opt} - \mathbf{h}(\hat{\mathbf{x}}_k^-) + \mathbf{H}_k \hat{\mathbf{x}}_k^-)$
- 10: $\bar{\xi}_k \leftarrow \xi_k$
- 11: $\bar{\Omega}_k \leftarrow \Omega_k$
- 12: $\mathcal{C}_k^a \leftarrow \mathcal{C}_k^a - \{\mathbf{z}_j^{opt}\}$
- 13: **end for**

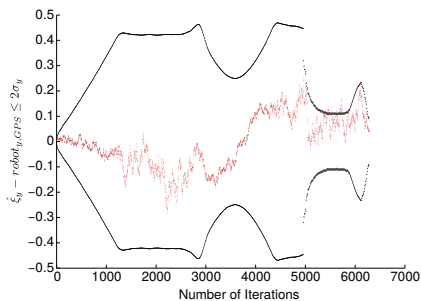
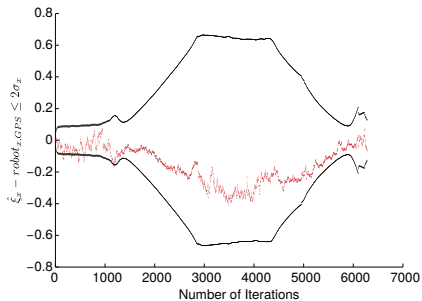
SLAM2D en entornos de agricultura



SLAM2D en entornos de agricultura—Resultados



Error de estimación de posición



SLAM3D–Inicialización de landmarks

Dado

- \mathbf{z} : punto imagen con distorsión ($\mathbf{z}_d \implies \mathbf{z}$)
- \mathbf{x} : vector de estado actual del SLAM
- ρ_0, σ_{ρ_0} : información a priori de la profundidad

SLAM3D–Inicialización de landmarks

Dado

- \mathbf{z} : punto imagen con distorsión ($\mathbf{z}_d \implies \mathbf{z}$)
- \mathbf{x} : vector de estado actual del SLAM
- ρ_0, σ_{ρ_0} : información a priori de la profundidad

$$\mathbf{y} = \mathbf{y}(\mathbf{x}, \mathbf{z}, \rho_0) \equiv \mathbf{m}_{\text{ID}} = \begin{bmatrix} \mathbf{t}^0 \\ (\gamma, \psi) \\ \rho \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \\ z^0 \\ \mathbf{g}(\mathbf{R}_c^g \mathbf{K}^{-1} \mathbf{z}) \\ \rho_0 \end{bmatrix}$$

SLAM3D–Modelo de velocidad constante

Estado del robot

$$\mathbf{r} = [x \quad y \quad \phi \quad v \quad \omega]^T$$

Modelo de movimiento de velocidad constante en tiempo discreto

$$\mathbf{r}_k = \mathbf{f}_r(\mathbf{r}_{k-1}, \mathbf{w}_{k-1}) = \begin{bmatrix} x_{k-1} + \Delta t(v_{k-1} + V_{k-1}) \cos(\phi_{k-1}) \\ y_{k-1} + \Delta t(v_{k-1} + V_{k-1}) \sin(\phi_{k-1}) \\ \phi_{k-1} + \Delta t(\omega_{k-1} + \Omega_{k-1}) \\ v_{k-1} + V_{k-1} \\ \omega_{k-1} + \Omega_{k-1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \phi_k \\ v_k \\ \omega_k \end{bmatrix}$$

SLAM3D–Modelo de velocidad constante

Estado del robot

$$\mathbf{r} = [x \quad y \quad \phi \quad v \quad \omega]^T$$

Modelo de movimiento de velocidad constante en tiempo discreto

$$\mathbf{r}_k = \mathbf{f}_r(\mathbf{r}_{k-1}, \mathbf{w}_{k-1}) = \begin{bmatrix} x_{k-1} + \Delta t(v_{k-1} + V_{k-1}) \cos(\phi_{k-1}) \\ y_{k-1} + \Delta t(v_{k-1} + V_{k-1}) \sin(\phi_{k-1}) \\ \phi_{k-1} + \Delta t(\omega_{k-1} + \Omega_{k-1}) \\ v_{k-1} + V_{k-1} \\ \omega_{k-1} + \Omega_{k-1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \phi_k \\ v_k \\ \omega_k \end{bmatrix}$$

donde

$$\mathbf{w}_r = \begin{bmatrix} a \\ \alpha \end{bmatrix}, \quad \mathbf{Q}_r = \begin{bmatrix} \sigma_a^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$$

SLAM3D–Modelo de observación

$$\mathbf{z}_k^i = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \quad i = 1, \dots, N_z$$

$$\mathbf{x} = \left[\mathbf{r} \quad \mathbf{m}_{\text{ID}}^{1T} \quad \dots \quad \mathbf{m}_{\text{EU}}^i \quad \dots \quad \mathbf{m}_{\text{ID}}^{N^T} \right]^T$$

Euclídea

$$\mathbf{m}_{\text{EU}}^i = [x_i \quad y_i \quad z_i]^T \in \mathbb{R}^3$$

Profundidad inversa

$$\mathbf{m}_{\text{ID}}^i = [\mathbf{p}_i^0 \quad \gamma_i \quad \psi_i \quad \rho_i]^T \in \mathbb{R}^5$$

SLAM3D–Modelo de observación

$$\mathbf{z}_k^i = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \quad i = 1, \dots, N_z$$

$$\mathbf{x} = \left[\mathbf{r} \quad \mathbf{m}_{\text{ID}}^{1T} \quad \dots \quad \mathbf{m}_{\text{EU}}^{iT} \quad \dots \quad \mathbf{m}_{\text{ID}}^{N^T} \right]^T$$

Euclídea

$$\mathbf{m}_{\text{EU}}^i = [x_i \quad y_i \quad z_i]^T \in \mathbb{R}^3$$

Profundidad inversa

$$\mathbf{m}_{\text{ID}}^i = [\mathbf{p}_i^0 \quad \gamma_i \quad \psi_i \quad \rho_i]^T \in \mathbb{R}^5$$

$$\mathbf{z}_i = \mathbf{K} \mathbf{R}_g^c \underbrace{(\mathbf{m}_{\text{EU}}^i - \mathbf{t}_c^g)}_{\mathbf{m}_i^c} \in \mathbb{P}^2$$

$$\mathbf{z}_i = \mathbf{K} \mathbf{R}_g^c \underbrace{(\rho_i(\mathbf{t}_i^0 - \mathbf{t}_c^g) + \mathbf{d}(\gamma_i, \psi_i))}_{\mathbf{m}_i^c} \in \mathbb{P}^2$$

SLAM3D–Modelo de observación

$$\mathbf{z}_k^i = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \quad i = 1, \dots, N_z$$

$$\mathbf{x} = \left[\mathbf{r} \quad \mathbf{m}_{\text{ID}}^{1T} \quad \dots \quad \mathbf{m}_{\text{EU}}^{iT} \quad \dots \quad \mathbf{m}_{\text{ID}}^{N^T} \right]^T$$

Euclídea

$$\mathbf{m}_{\text{EU}}^i = [x_i \quad y_i \quad z_i]^T \in \mathbb{R}^3$$

Profundidad inversa

$$\mathbf{m}_{\text{ID}}^i = [\mathbf{p}_i^0 \quad \gamma_i \quad \psi_i \quad \rho_i]^T \in \mathbb{R}^5$$

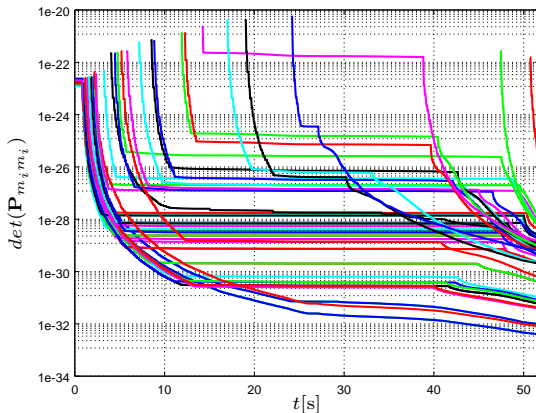
$$\mathbf{z}_i = \mathbf{K} \underbrace{\mathbf{R}_g^c (\mathbf{m}_{\text{EU}}^i - \mathbf{t}_c^g)}_{\mathbf{m}_i^c} \in \mathbb{P}^2$$

$$\mathbf{z}_i = \mathbf{K} \underbrace{\mathbf{R}_g^c (\rho_i (\mathbf{t}_i^0 - \mathbf{t}_c^g) + \mathbf{d}(\gamma_i, \psi_i))}_{\mathbf{m}_i^c} \in \mathbb{P}^2$$

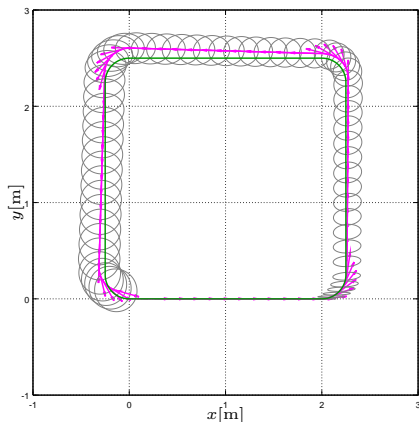
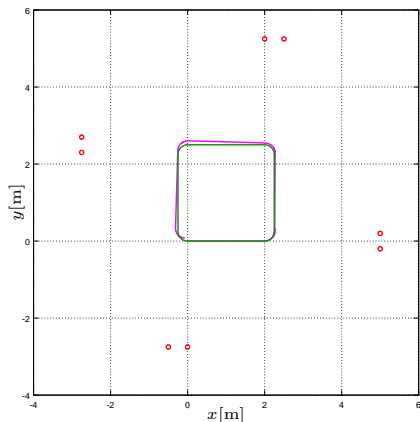
Luego $\mathbf{z}_d^i \implies \mathbf{z}^i$ (modelo de distorsión radial)

SLAM3D–Resultados experimentales de UKF-SLAM (2)

Incertidumbre de landmarks

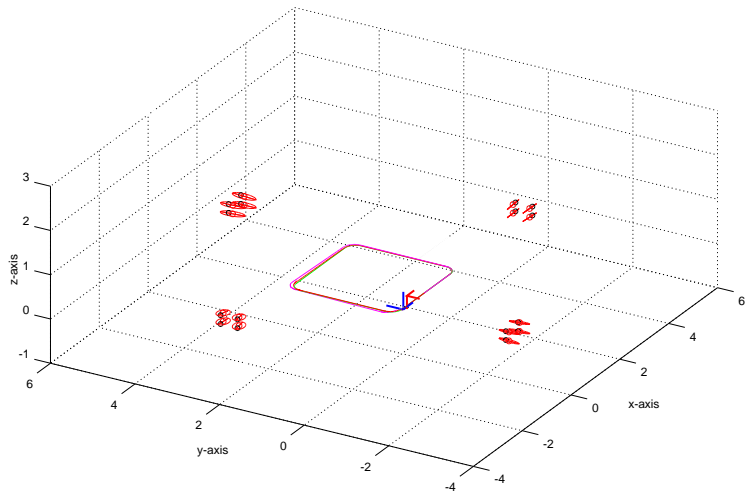


SLAM3D–UKF–SLAM para localización



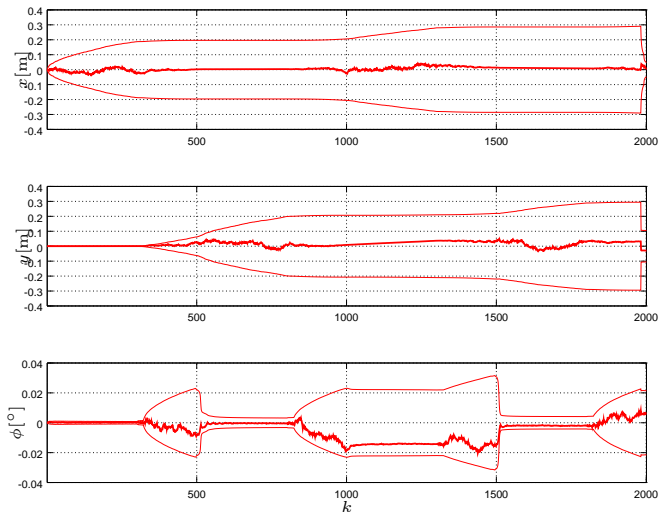
- $\alpha_1 = 0,10^\circ/^\circ$, $\alpha_2 = 0,50^\circ/m$, $\alpha_3 = 0,50m/m$, $\alpha_4 = 0,10m/^\circ$
- $\sigma_z = 1$ píxel
- Altura de landmarks: 0.15m y 0.35m

SLAM3D–Resultados de simulación de UKF-SLAM para localización



SLAM3D–Resultados de simulación de UKF-SLAM para localización

Estimación de la pose del robot



Resultados experimentales de UKF-SLAM - MMO vs. MMVC

- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 15$ (1/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 20$ (2/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

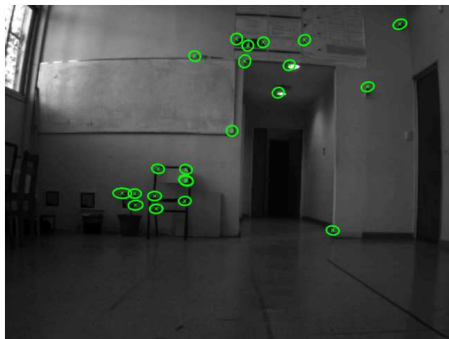
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 30$ (3/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

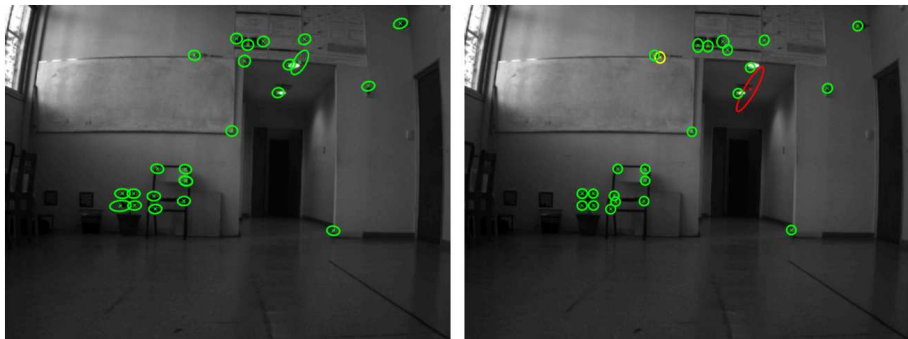
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 33$ (4/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

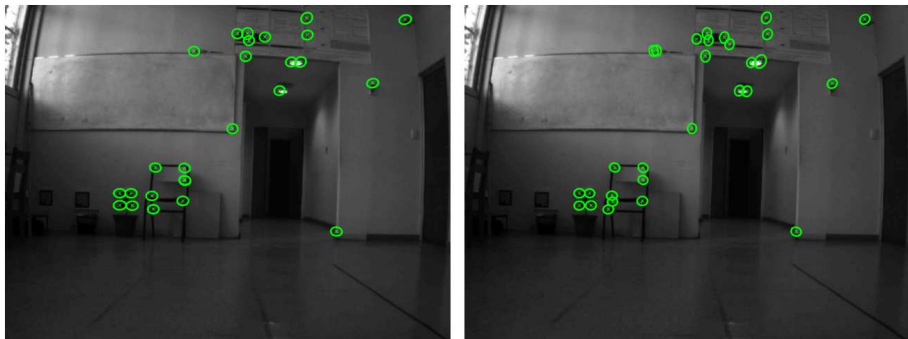
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 38$ (5/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

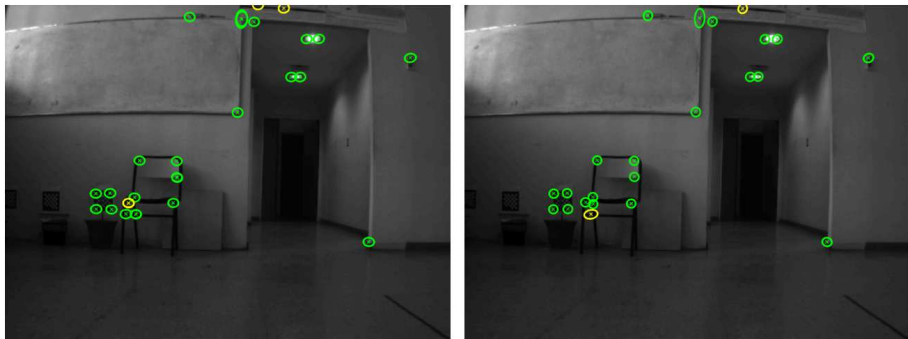
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 55$ (6/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

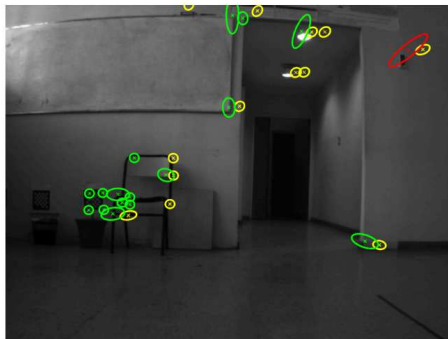
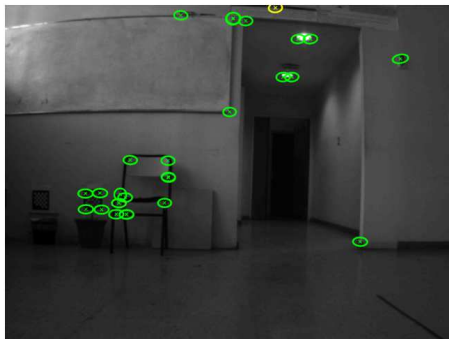
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 185$ (7/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

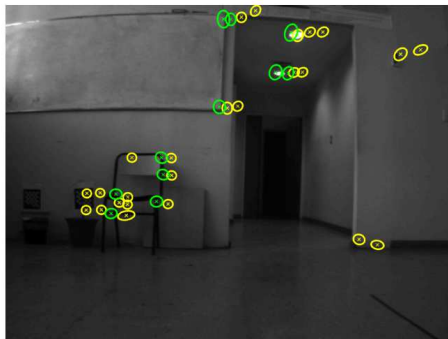
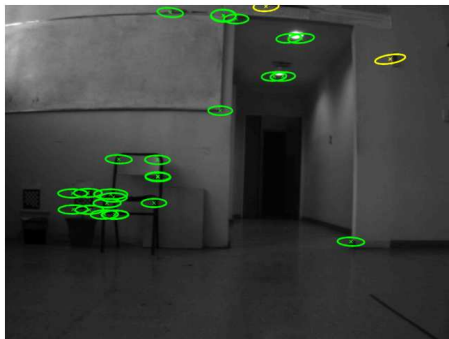
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 207$ (8/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

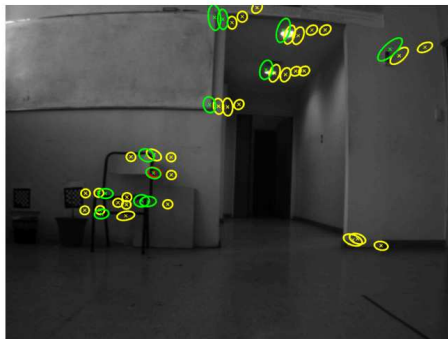
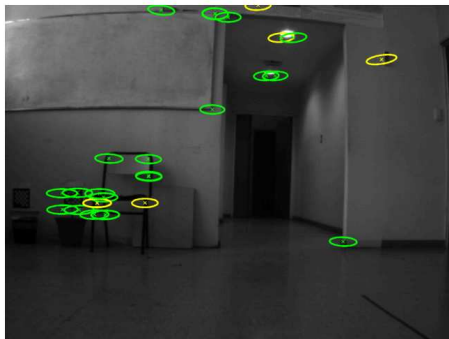
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 213$ (9/10)

Resultados experimentales de UKF-SLAM - MMO vs. MMVC

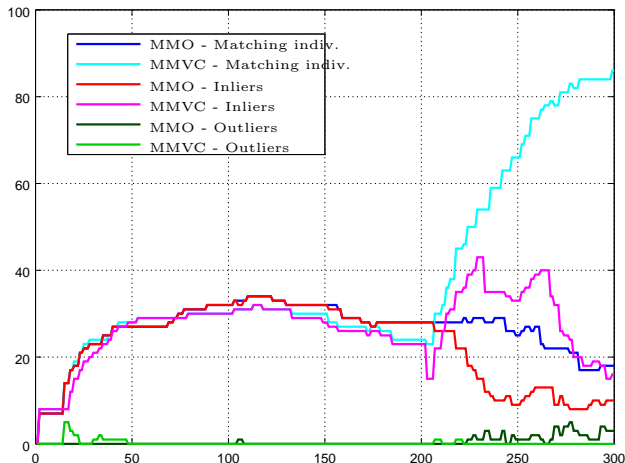
- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



Paso temporal $k = 218$ (10/10)

SLAM3D—Resultados experimentales de UKF-SLAM - MMO vs. MMVC

- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



SLAM3D–Asociación de datos

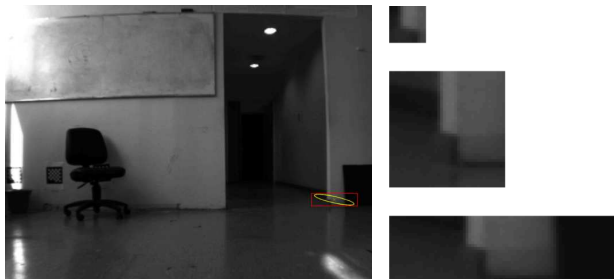
- *Imagen a imagen* usado comúnmente en procesamiento offline (exhaustivo)

SLAM3D–Asociación de datos

- *Imagen a imagen* usado comúnmente en procesamiento offline (exhaustivo)
- *Mapa a imagen* aprovechando la estimación a priori del filtro del SLAM
 - ▶ Permite **incrementar la robustez y reducir el costo computacional**
 - ▶ Waring de patch (descriptor), zona de interés, NCC

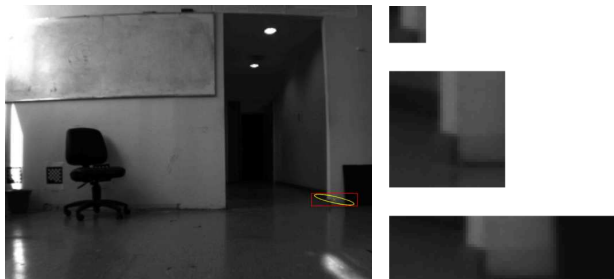
SLAM3D–Asociación de datos

- *Imagen a imagen* usado comúnmente en procesamiento offline (exhaustivo)
- *Mapa a imagen* aprovechando la estimación a priori del filtro del SLAM
 - ▶ Permite **incrementar la robustez y reducir el costo computacional**
 - ▶ Waring de patch (descriptor), zona de interés, NCC



SLAM3D–Asociación de datos

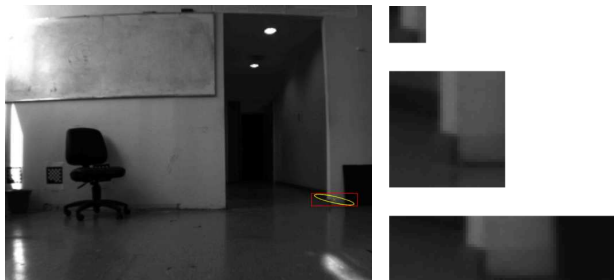
- *Imagen a imagen* usado comúnmente en procesamiento offline (exhaustivo)
- *Mapa a imagen* aprovechando la estimación a priori del filtro del SLAM
 - ▶ Permite **incrementar la robustez y reducir el costo computacional**
 - ▶ Waring de patch (descriptor), zona de interés, NCC



$$(\mathbf{z} - \mathbf{z}_i^-)^T (\mathbf{R}_i^-)^{-1} (\mathbf{z} - \mathbf{z}_i^-) = k, \implies$$

SLAM3D–Asociación de datos

- *Imagen a imagen* usado comúnmente en procesamiento offline (exhaustivo)
- *Mapa a imagen* aprovechando la estimación a priori del filtro del SLAM
 - ▶ Permite **incrementar la robustez y reducir el costo computacional**
 - ▶ Waring de patch (descriptor), zona de interés, NCC



$$(\mathbf{z} - \mathbf{z}_i^-)^T (\mathbf{R}_i^-)^{-1} (\mathbf{z} - \mathbf{z}_i^-) = k, \implies \mathbf{u}_i^T \mathbf{C}_i \mathbf{u}_i = 0$$

SLAM3D–Asociación de datos

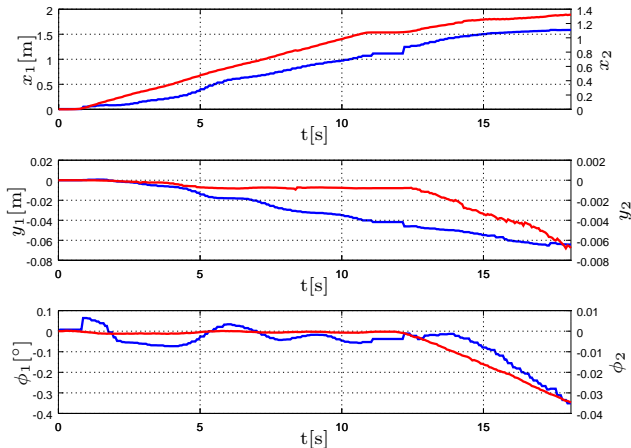
- *Imagen a imagen* usado comúnmente en procesamiento offline (exhaustivo)
- *Mapa a imagen* aprovechando la estimación a priori del filtro del SLAM
 - ▶ Permite **incrementar la robustez y reducir el costo computacional**
 - ▶ Waring de patch (descriptor), zona de interés, NCC



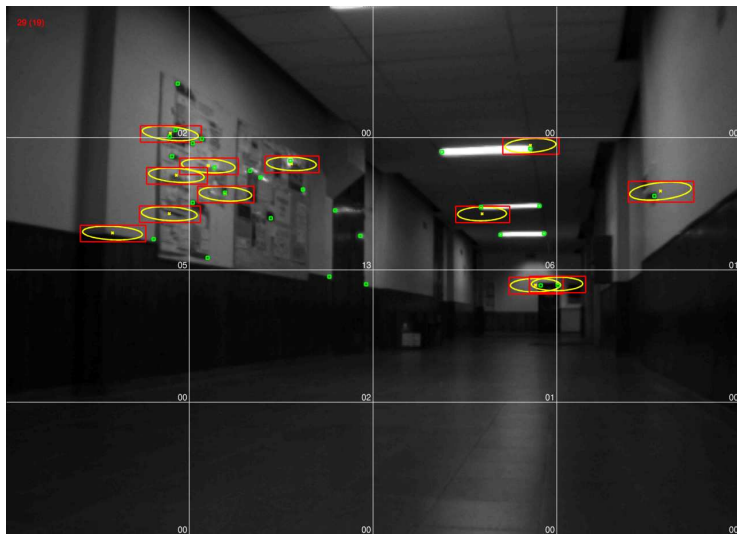
$$(\mathbf{z} - \mathbf{z}_i^-)^T (\mathbf{R}_i^-)^{-1} (\mathbf{z} - \mathbf{z}_i^-) = k, \implies \mathbf{u}_i^T \mathbf{C}_i \mathbf{u}_i = 0 \quad \mathbf{l} = \mathbf{C} \mathbf{u}, \quad \mathbf{l} = [a \quad b \quad c]^T \in \mathbb{P}^2$$

SLAM3D—Resultados experimentales de UKF-SLAM - MMO vs. MMVC

- Modelo de movimiento de odometría (MMO)
- Modelo de movimiento de velocidad constante (MMVC)



SLAM3D-Extracción de características



SLAM3D-Extracción de características

