Universitat Oberta de Catalunya

uoc.edu

Contribución al desarrollo de una aplicación móvil de posicionamiento en interiores basado en luz (VLP) usando los sensores inerciales del dispositivo

Trabajo Final de Máster – Smart Cities

Autor: **Aitor Alcázar Fernández**

Tutores: Antoni Pérez Navarro (UOC)
Álvaro De La Llana Calvo (UAH)

Profesor: Carlos Monzo Sánchez (UOC)

Contenido

Introducción
Marco teórico
Desarrollo
Resultados
Conclusiones

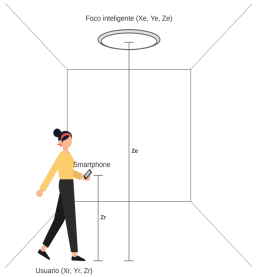
Introducción

Contexto y objetivos del TFM

Contexto

geintra
Universidad de Alcalá
Castilla-La Mancha

Proyecto GUIA



Objetivos del TFM

Objetivo #1:
Algoritmo de corrección de la posición según sus ángulos de posición

Objetivo #2:
Algoritmo de estimación de la posición cuando no se capta un foco

Marco teórico

Medidas y estrategias de posicionamiento

Medidas

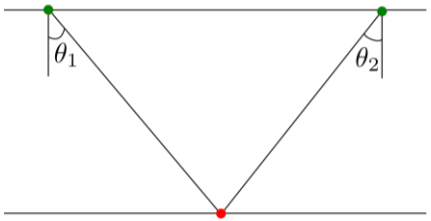
Ángulo de llegada (AoA)
Consiste en medir el ángulo de recepción de una señal para estimar su dirección de llegada

Tiempo de llegada (ToA)
Consiste en obtener la distancia entre el emisor y el receptor mediante el tiempo de propagación

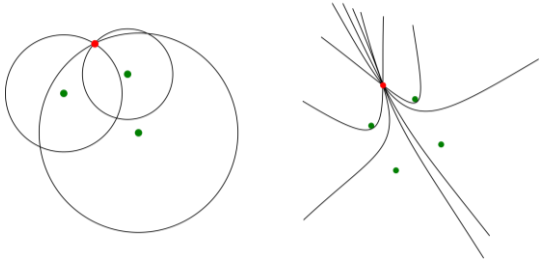
Otros
Medidas basadas en RSS, fingerprinting o navegación por estima (*dead reckoning*)

Estrategias

Triangulación



Lateralización esférica e hiperbólica



Sensores y dispositivos móviles

Sensores en dispositivos móviles

Inerciales
- Giroscopio
- Acelerómetro
- Magnetómetro

Cámara
- Frontal
- Trasera

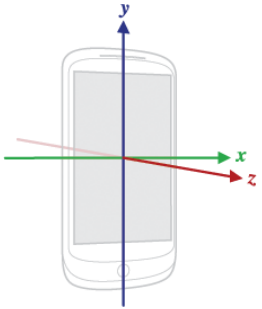
Otros
- Capacitivos
- Proximidad
- Barómetro
- GNSS

IMU en dispositivos móviles

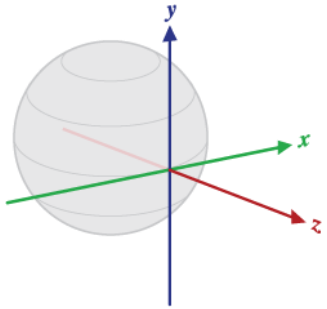
IMU similar en dispositivos Android
- Sensores inerciales básicos
- Tipo *strapdown*
- Integración hardware-software

Sistema de referencia único

Body frame

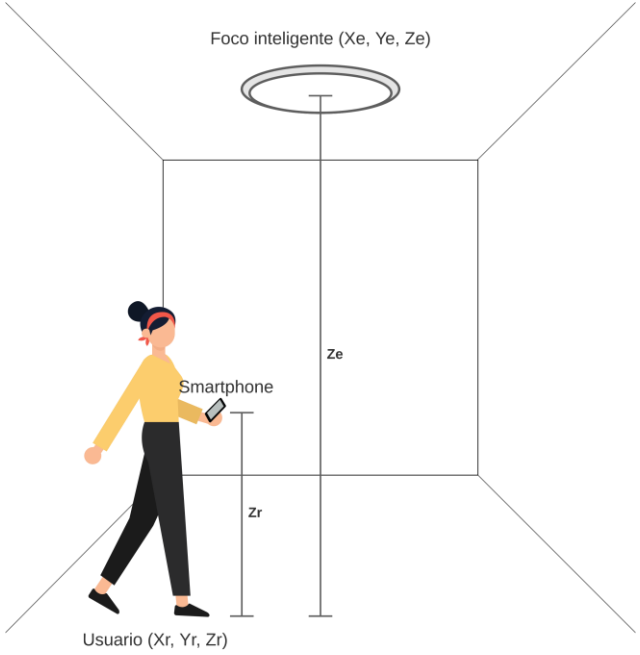


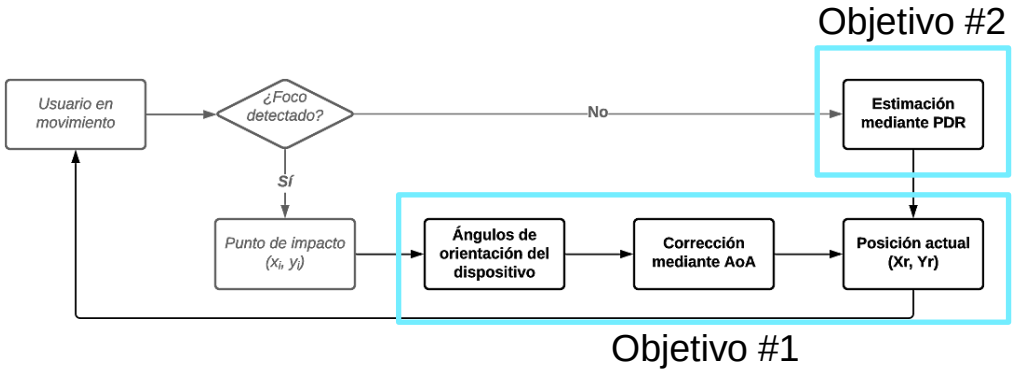
Global frame



Desarrollo

Topología de la aplicación





Implementación

android studio

Kotlin

Algoritmo de corrección basado en AoA

Modelo matemático: *Pinhole*

Modelo de *pinhole*

$$\begin{pmatrix} sx_i' \\ sy_i' \\ s \end{pmatrix} = \underbrace{\begin{pmatrix} f & 0 & C_x \\ 0 & f & C_y \\ 0 & 0 & 1 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}}_{R} \begin{pmatrix} X_e - X_r \\ Y_e - Y_r \\ Z_e - Z_r \end{pmatrix}$$

Posicionamiento de un único sensor

$$\begin{pmatrix} X_r \\ Y_r \\ Z_r \end{pmatrix} = -\mathbf{R}^{-1}\mathbf{A}^{-1} \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} s + \begin{pmatrix} X_e \\ Y_e \\ Z_e \end{pmatrix}$$

$$\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = -\mathbf{R}^{-1}\mathbf{A}^{-1} \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} \qquad s = \frac{Z_r - Z_e}{q_3}$$

Dispositivo móvil

Samsung Galaxy Tab S3 (SM-T820)

Parámetros

Parámetro	Valor
Longitud focal (f)	3,31 mm
Tamaño del píxel	1,20 μm
Nº filas por imagen	2560 px
Nº columnas por imagen	1920 px

Algoritmo de corrección basado en AoA

Cálculo de los parámetros del modelo matemático

Matriz intrínseca A

Dispositivo móvil

$$C_x = d\frac{w}{2}, \quad C_y = d\frac{h}{2} \longrightarrow \mathbf{A} = \begin{pmatrix} f & 0 & d\frac{w}{2} \\ 0 & f & d\frac{h}{2} \\ 0 & 0 & 1 \end{pmatrix}$$

Matriz de rotación R

$$\mathbf{R}_X(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix}$$

$$\mathbf{R}_Y(\theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix} \longrightarrow \mathbf{R} = \mathbf{R}_Z(\psi)\mathbf{R}_Y(\theta)\mathbf{R}_X(\phi)$$

$$\mathbf{R}_Z(\psi) = \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Punto de impacto

Punto medido — Punto calculado

$$(x_i', y_i') \longrightarrow \begin{aligned} x_i &= d \cdot (w - x_i') \\ y_i &= d \cdot (h - y_i') \end{aligned}$$

Altura del receptor y posición del emisor

La altura del receptor es fija

$$Z_r = 120 \text{ cm}$$

La posición del emisor se obtiene mediante el foco inteligente

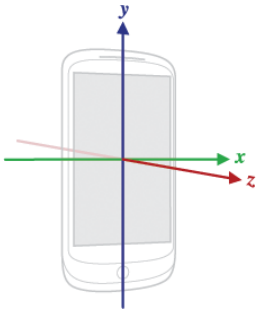
$$(X_e, Y_e, Z_e)$$

Algoritmo de estimación basado en PDR

Fases de implementación

Ecuación general de PDR

$$\begin{cases} X_k = X_{k-1} + L_{k-1,k} \cdot \sin(\psi_{k-1,k}) \\ Y_k = Y_{k-1} + L_{k-1,k} \cdot \cos(\psi_{k-1,k}) \end{cases}$$



Y hacia el Norte
X hacia el Este

Detección de pasos

Gracias al acelerómetro del dispositivo se obtienen las aceleraciones en los tres ejes coordenados

Longitud del paso

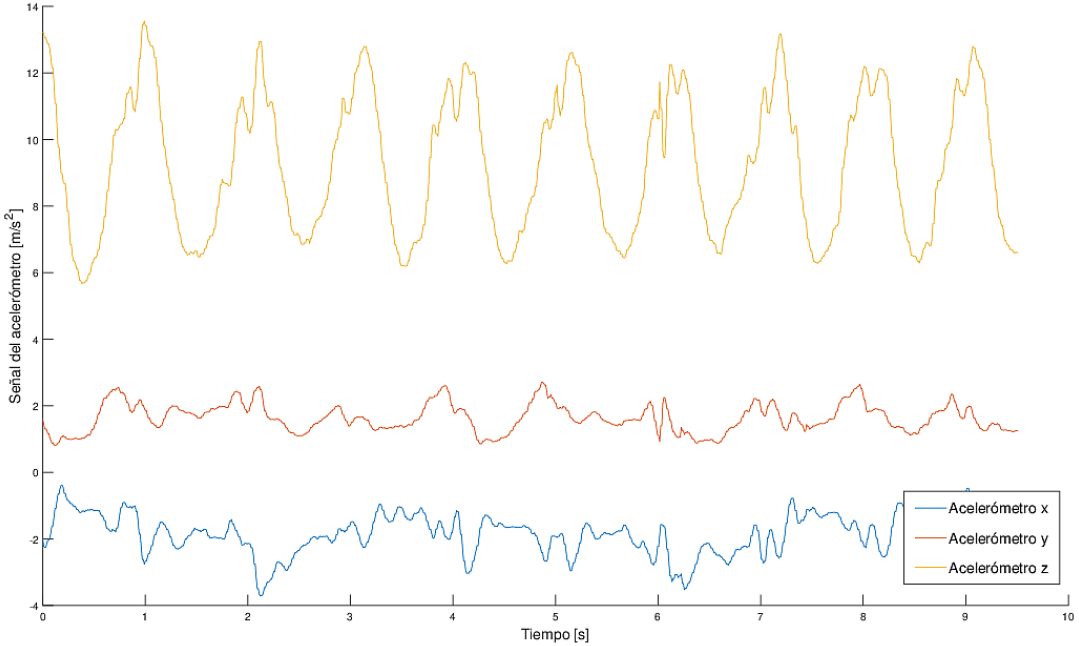
Se determina la distancia entre cada uno de los pasos del usuario

Rumbo del usuario

Gracias al magnetómetro y al giroscopio se obtiene el rumbo estimado del usuario

Algoritmo de estimación basado en PDR

Detección de pasos: señales



Algoritmo de estimación basado en PDR

Detección de pasos: señal filtrada, umbral y picos

Filtro digital (LPF)

$$y[i] = y[i-1] + \alpha(x[i] - y[i-1])$$

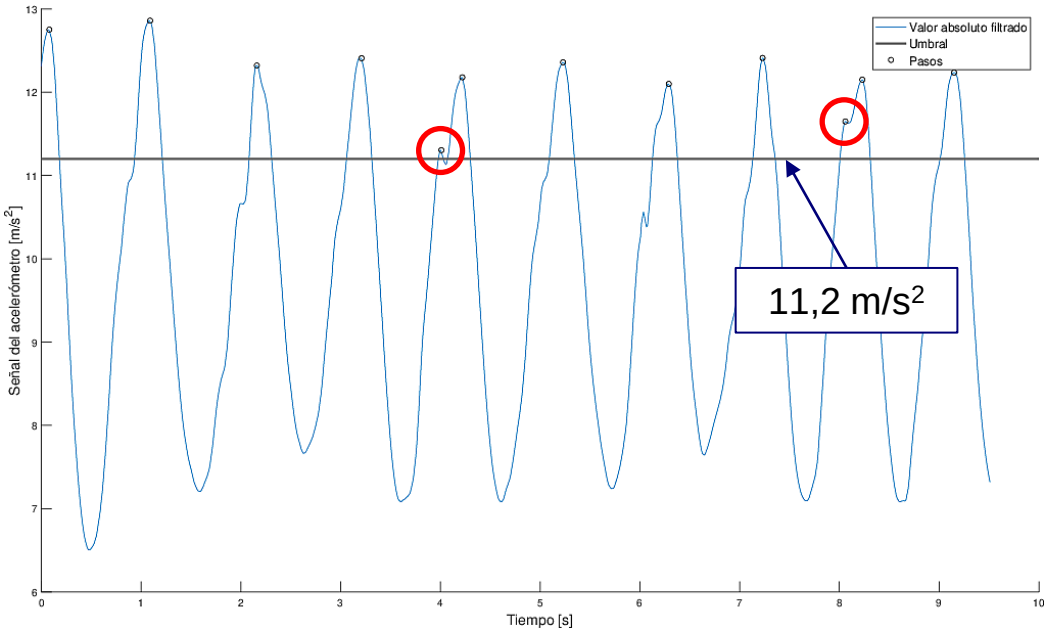
Factor de suavizado

$$\tau = \frac{1}{2\pi f_c} \rightarrow \alpha = \frac{T_s}{\tau + T_s}$$

$$f_c = 15\,\mathrm{Hz} \qquad \alpha = 0{,}4851$$

Umbral y picos

- Umbral obtenido como solución de compromiso
- Picos con falsos positivos



Algoritmo de estimación basado en PDR

Longitud del paso

Estudios de longitudes generales

	Hombres	Mujeres	Media general
Longitud del paso (cm)	79	66	72,5
Longitud de la zancada (cm)	158	132	145
Cadencia (pasos/min)	117	117	117
Velocidad (m/s)	1,54	1,31	1,425

Ecuación general con L constante

$$\begin{cases} X_k = X_{k-1} + L \cdot \sin(\psi_{k-1,k}) \\ Y_k = Y_{k-1} + L \cdot \cos(\psi_{k-1,k}) \end{cases}$$

Particularización en el proyecto

En los museos se camina más lento, lo que da lugar a pasos de L=58 cm de media

Rumbo del usuario

Ángulo de dirección

Se actualiza el ángulo de posición cada vez que se detecta un paso. Ese ángulo representa el rumbo del usuario

Ecuación general dependiente del rumbo actual

$$\begin{cases} X_k = X_{k-1} + L \cdot \sin \psi_k \\ Y_k = Y_{k-1} + L \cdot \cos \psi_k \end{cases}$$

Resultados

Algoritmo de corrección basado en AoA

Medidas estáticas para determinar errores

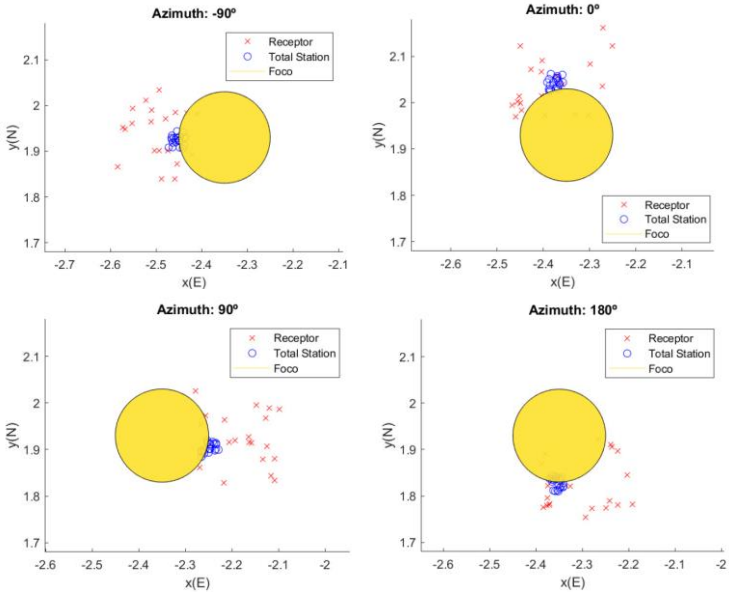
Configuración y *ground truth*

Estación total y prisma de Leica

Errores de 1 mm en medidas de 10 km

Distancia

$$d_{ts,rx} = \sqrt{(x_{ts} - x_{rx})^2 + (y_{ts} - y_{rx})^2}$$



Resultados

Algoritmo de corrección basado en AoA

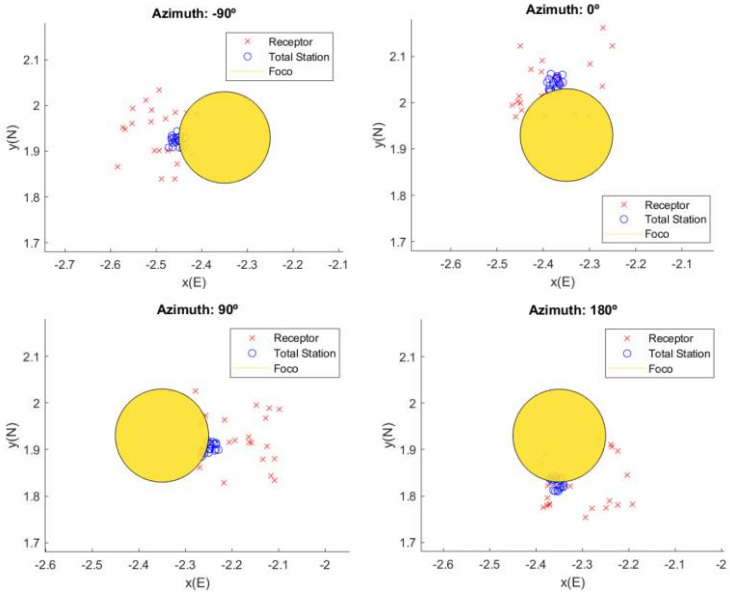
Medidas estáticas para determinar errores

Errores

Azimut (°)	max(E_d) (m)	$\bar{E}_d$ (m)	$\sigma^2(E_d)$ (m)	$\sigma(E_d)$ (m)
-90	0,1389	0,0522	0,0895	0,0299
0	0,1798	0,0854	0,1625	0,0403
90	0,1727	0,1064	0,1899	0,0436
180	0,1830	0,0935	0,2567	0,0506

Roll (°)	max(E_d) (m)	$\bar{E}_d$ (m)	$\sigma^2(E_d)$ (m)	$\sigma(E_d)$ (m)
-8	0,1407	0,0896	0,1388	0,0372
-4	0,1726	0,0859	0,1343	0,0366
0	0,1727	0,1020	0,1701	0,0412
4	0,1566	0,0828	0,1926	0,0439
8	0,1830	0,1030	0,2522	0,0502

Pitch (°)	max(E_d) (m)	$\bar{E}_d$ (m)	$\sigma^2(E_d)$ (m)	$\sigma(E_d)$ (m)
0	0,1727	0,0801	0,1486	0,0385
4	0,1608	0,1025	0,1613	0,0402
8	0,1830	0,0953	0,2217	0,0471
12	0,1798	0,0927	0,1709	0,0413



Algoritmo de corrección basado en AoA

Medidas para determinar el área de cobertura

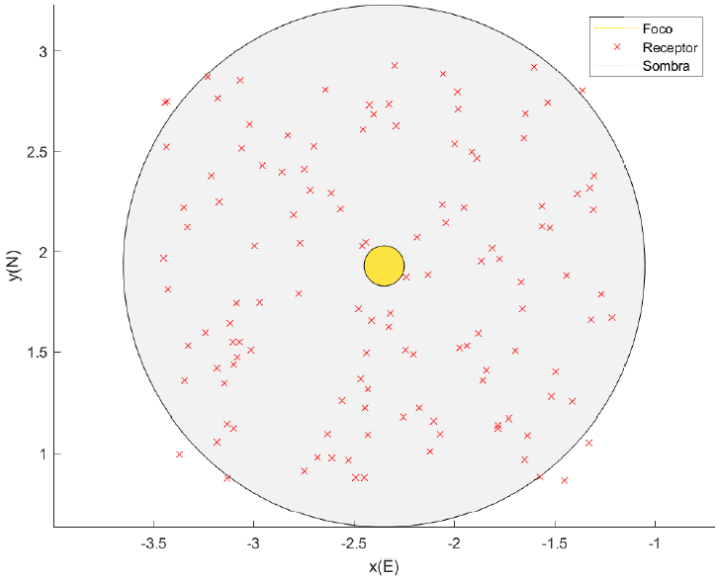
Configuración

Se utiliza la aplicación bajo el foco hasta determinar cuando no se detecta

Interviene la altura del receptor y el emisor, los ángulos de posición del dispositivo, el sistema de lentes, etc.

Resultados

Se traza una circunferencia para establecer el área de cobertura en 1,3 m



Algoritmo de estimación basado en PDR

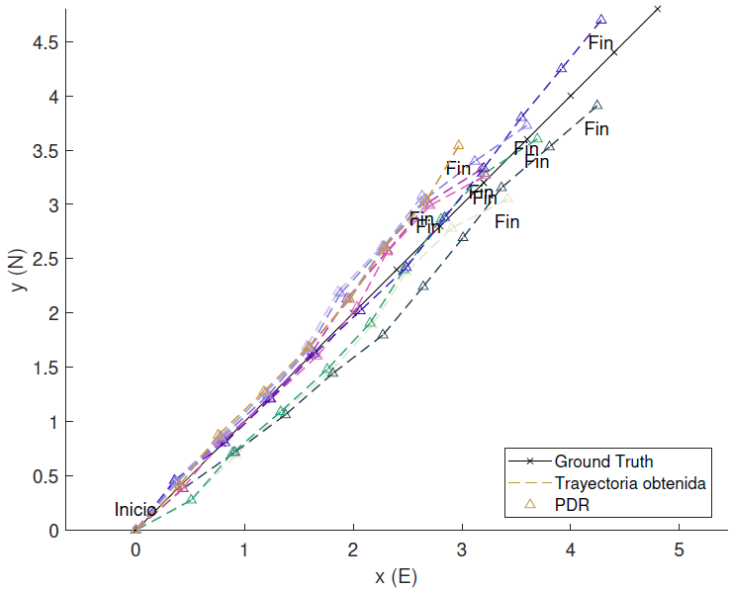
Medidas sobre una recta

Configuración

Se establece una recta como trayectoria a seguir

Se marca en el suelo una distancia conocida entre pasos para obtener el error

Se repite 10 veces

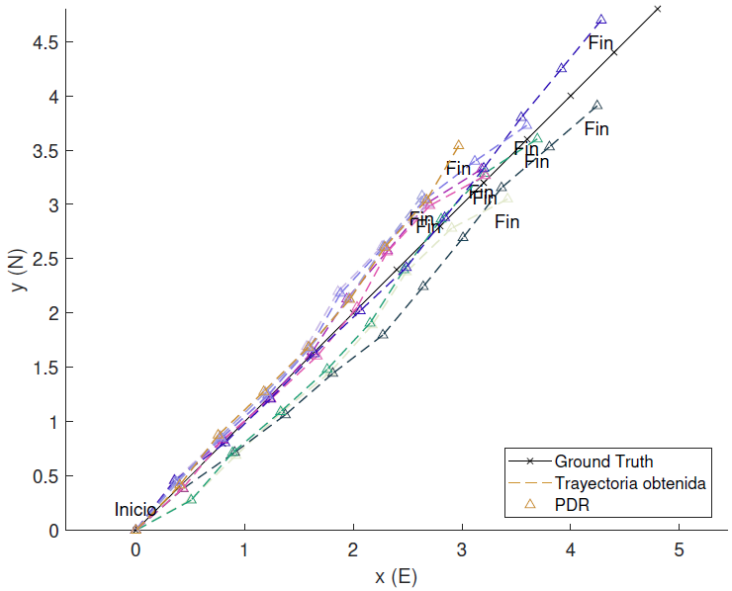


Algoritmo de estimación basado en PDR

Medidas sobre una recta

Resultados

Trayecto	max(E_d) (m)	$\bar{E}_d$ (m)	$\sigma^2(E_d)$ (m)	$\sigma(E_d)$ (m)
1	0,1990	0,1287	0,0026	0,0507
2	0,2439	0,1115	0,0053	0,0726
3	0,1068	0,0784	0,0006	0,0238
4	0,1058	0,0745	0,0010	0,0313
5	0,1261	0,0828	0,0010	0,0321
6	0,1212	0,0847	0,0010	0,0311
7	0,1026	0,0708	0,0005	0,0222
8	0,1179	0,0893	0,0007	0,0266
9	0,0839	0,0568	0,0007	0,0258
10	0,1015	0,0815	0,0005	0,0222



Algoritmo de estimación basado en PDR

Medidas sobre un paralelogramo

Configuración

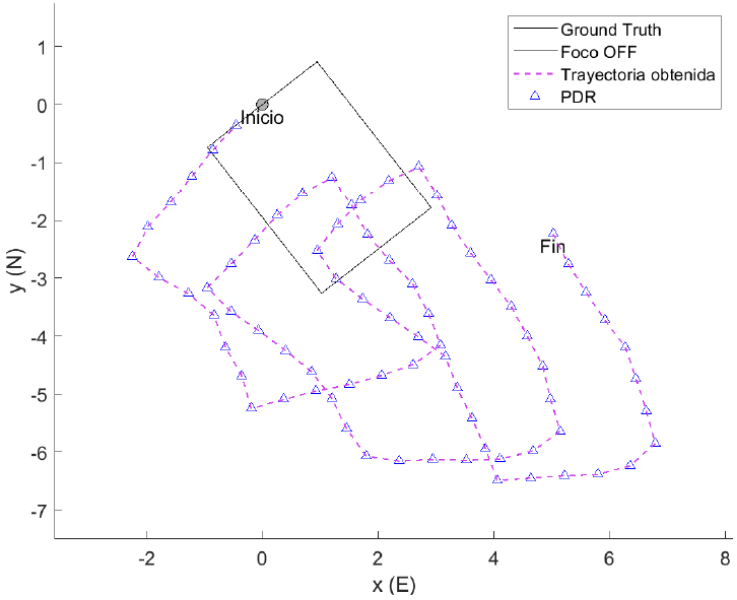
Se establece un paralelogramo como trayectoria a seguir

Se marca en el suelo cada una de las esquinas para obtener el error

Se dan 3 vueltas

Resultados

Vuelta	$\max(E_d)$ (m)	$\bar{E}_d$ (m)	$\sigma^2(E_d)$ (m)	$\sigma(E_d)$ (m)
1	2,3780	2,3478	0,014	0,0369
2	4,4528	3,0812	0,8801	0,9381
3	5,6130	4,4227	1,7074	1,3067



Fusión de los algoritmos

Resultados de la fusión sobre un paralelogramo

Configuración

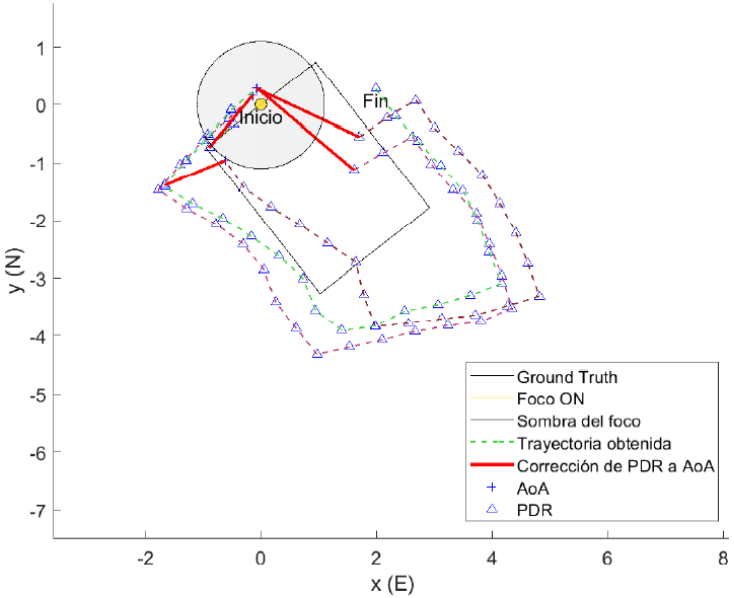
Se establece un paralelogramo como trayectoria a seguir

Se marca en el suelo cada una de las esquinas para obtener el error

Se dan 3 vueltas con **el foco encendido**

Resultados

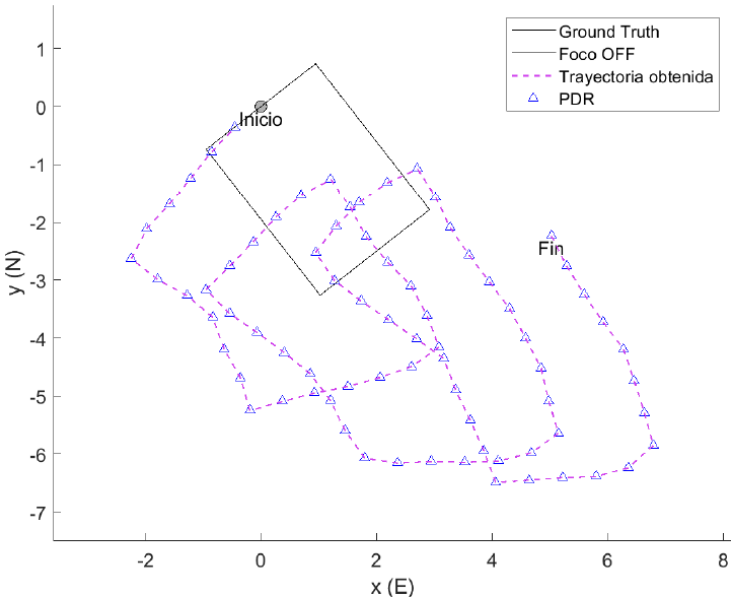
Vuelta	$\max(\mathbf{E_d})$ (m)	$\bar{\mathbf{E}}_\mathbf{d}$ (m)	$\sigma^2(\mathbf{E_d})$ (m)	$\sigma(\mathbf{E_d})$ (m)
1	2,2401	1,6321	0,3978	0,6308
2	2,4330	1,4490	0,7819	0,8843
3	1,8047	1,1685	0,2055	0,4533



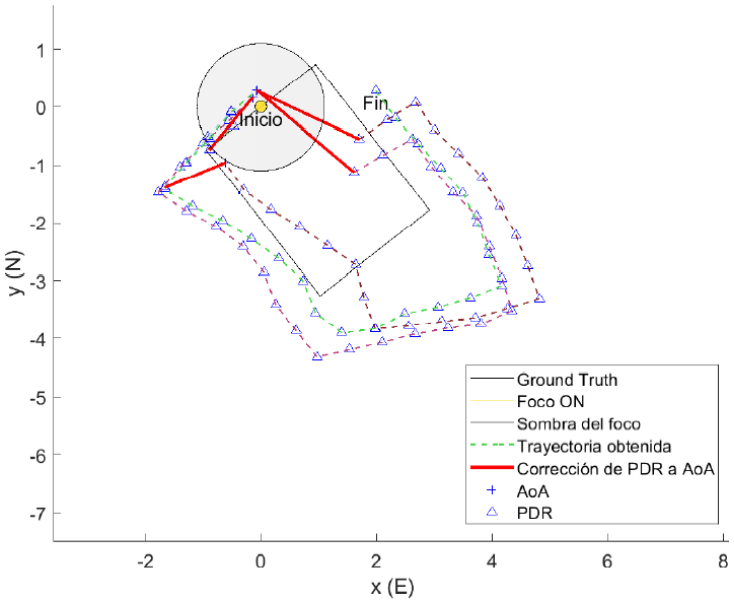
Fusión de los algoritmos

Comparación de resultados sobre un paralelogramo

Sin fusión de algoritmos



Con fusión de algoritmos



Conclusiones

Conclusiones

- Se **cumplen** los dos objetivos definidos en el TFM
- La fusión de los dos algoritmos mejora la **precisión** y la **calidad** de los resultados generales
- Los errores obtenidos son **válidos** en el contexto del proyecto GUIA
- La contribución realizada es **aplicable** a aplicaciones similares

Líneas de futuro

- Mejoras en cada uno de los algoritmos por separado para incrementar la calidad de los resultados
- Análisis de la colocación de las luminarias en el entorno
- Nuevos algoritmos de posicionamiento como el filtrado de Kalman o las señales GNSS
- Nuevas líneas de investigación para otras aplicaciones VLP
- Adaptar la aplicación para otros sistemas operativos (iOS)

Universitat Oberta
de Catalunya

Gracias por su atención

aitoralcazar@uoc.edu

UOC.universitat
@UOCuniversidad
UOCuniversitat