

FUSIÓN DE SENSORES PARA LOCALIZACIÓN ROBUSTA DE VEHÍCULOS AUTÓNOMOS EN ÁREAS URBANAS

Kerman Viana, Mikel Diez, Asier Zubizarreta
Escuela de Ingeniería de Bilbao, Universidad del País Vasco (UPV/EHU)
kviana001@ehu.eus

Resumen

El posicionamiento por GPS en zonas urbanas densamente pobladas puede ser un reto, principalmente debido al bloqueo de señales por edificios o túneles. Es por ello que los vehículos autónomos necesitan implementar alternativas para estas situaciones mediante una estructura de localización tolerante a fallos. Este es un área de gran interés en la que predominan el uso de técnicas de duplicación-comparación en combinación con las belief function, además de técnicas de localización alternativas. Este trabajo propone una estructura de localización para zonas urbanas densamente pobladas que incluye tanto un algoritmo robusto de detección de errores, capaz de evaluar el rango de confianza de cada estimación, como una precisa técnica de localización alternativa basada en un algoritmo de map matching de bajo coste computacional. La validación en un entorno simulado ha verificado la funcionalidad de la propuesta.

Palabras clave: Conducción autónoma, fusión de sensores, detección de errores, localización.

1. INTRODUCCIÓN

Una localización precisa es fundamental para obtener un alto nivel de automatización en vehículos. En años recientes la tarea de localización ha sido habitualmente asignada a Sistemas de Posicionamiento Global o GPS, combinados con Sistemas de Navegación Inercial o INS [18]. Aun así, el posicionamiento de vehículos autónomos mediante el uso de estas técnicas requiere de una buena calidad de señal GPS. Esto no siempre es posible en áreas urbanas densamente pobladas, ya que la señal puede verse interrumpida o bloqueada en presencia de edificios o túneles; haciendo que el sistema de localización se vuelva poco fiable.

Por tanto, para poder obtener una localización precisa en áreas urbanas densamente pobladas, es necesario desarrollar técnicas de localización alternativas hasta que la señal GPS es recuperada. Las técnicas más comunes pueden agruparse en dos grandes grupos: aproximación basada en lo-

calización local, las cuales usan la última posición conocida para estimar una localización relativa; y las técnicas basadas en mapas digitales, que ofrecen una localización global mediante el uso de referencias globales a lo largo de la trayectoria del vehículo.

En la literatura se han propuesto diferentes aproximaciones basadas en técnicas de localización local en los últimos años. Una de las técnicas más tradicionales y todavía habituales es el uso de la odometría [10] y los sensores inerciales [2]. El uso de sensores visuales como las cámaras [12] y el LiDAR [20] también están aglutinando cada vez más interés, principalmente debido a la mejora en la precisión de sus mediciones. A pesar de ello, todas estas aproximaciones se basan en técnicas de integración a partir de una última posición conocida. En consecuencia, acumulan un error cada vez mayor con el tiempo y crean una deriva en la localización del vehículo respecto a su posición real.

Por otro lado, las técnicas basadas en mapas digitales consisten en obtener la localización del vehículo mediante el emparejamiento de su posición estimada con una red de carreteras y cruces globalmente referenciadas. Si los mapas usados son precisos, esta técnica puede aportar una localización global, convirtiéndola en un complemento interesante para el posicionamiento mediante GPS [14]. Es más, las técnicas previamente analizadas de localización local pueden ser combinadas con mapas digitales para crear algoritmos que permitan la corrección de la deriva o el error acumulado [19][8].

Como se puede observar, cada sensor o estrategia de localización tiene sus puntos fuertes y débiles. Es aquí donde la combinación de cada fuente de información se puede utilizar para lograr datos más precisos, lo cual es un campo ampliamente estudiado en la literatura [1]. En el caso del posicionamiento para vehículos, las estrategias de fusión suelen agruparse en dos grupos: técnicas de optimización y uso de filtros. En el caso del primero, técnicas como el *Bundle Adjustment* [21] son usadas habitualmente en la literatura, debido a su consistencia y precisión en una gran variedad de

escenarios. Por otro lado, el uso de filtros, principalmente basados en los filtros de Kalman o sus variantes extendidas (EKF), son también habituales debido a su fácil convergencia y consistencia. Además, éstos últimos también dan excelentes resultados a la hora de combinar sensores de naturaleza muy diferente [6]. Por último, los filtros de Kalman también ofrecen una gran versatilidad a la hora de estimar los diferentes estados de un mismo sistema [18], convirtiendo este grupo en la principal elección a la hora de fusionar información dentro del contexto de la localización para vehículos autónomos.

Por tanto, una correcta fusión de técnicas basadas en mapas digitales, posicionamiento local y sensores GPS/INS puede servir para obtener una correcta localización de vehículos en áreas urbanas densamente pobladas. Aun así, para poder sacar el máximo partido de todos los módulos de localización en la fusión, es necesario crear una estructura tolerante a fallos; es decir, tiene que ser capaz de detectar errores en el sistema y proponer estrategias adecuadas para cada situación. Además de lo anterior, es necesario implementar un sistema de toma de decisiones capaz de detectar y discriminar un módulo de localización erróneo, para evitar así que la deriva en la localización se acumule debido a un error.

La mayoría de las estructuras tolerantes a errores en los vehículos autónomos se basan en técnicas de duplicación-comparación y el uso de *belief function*, tal y como son definidas en el *Transferable Belief Model* (TBM) propuesto por Smets [16]. Estos modelos consisten en diseñar una estructura con módulos de localización redundantes que comparan entre sí sus estimaciones para desenmascarar posibles errores, y poder así discriminarlos para evitar una propagación del error en el posicionamiento. [7] y [9] ofrecen ejemplos de aplicación para estos modelos, donde se usa información redundante para detectar fuentes erróneas antes de fusionarlas en un filtro de Kalman. Aun así, estos enfoques suelen centrarse solamente en la detección de errores, y en contadas ocasiones abordan la problemática de desarrollar alternativas de posicionamiento realmente precisas. Además, la evaluación de fuentes de información suele realizarse mediante el análisis de señales de control, las cuales también pueden fallar en escenarios reales y llevar a conclusiones equivocadas.

Este trabajo propone una estructura de localización tolerante a errores para vehículos autónomos en áreas densamente pobladas. La estructura propuesta expone 2 principales aportaciones sobre los trabajos existentes en este área: 1) un sistema de localización alternativo y de bajo coste computacional basado en una técnica de *map matching*

[14], que también incluye un algoritmo de detección de cruces para limitar la deriva y mejorar su rendimiento; y 2) un sistema de detección de errores basado en la evaluación del conflicto entre las diferentes estimaciones de posicionamiento, el cual calcula una fiabilidad dinámica para cada fuente que, posteriormente, se implementa en un filtro de Kalman extendido o EKF a través de su matriz de covarianzas. Este filtro de Kalman trabaja tanto como método de fusión de información, como módulo de toma de decisiones. Esta estructura ha sido validada en un entorno de simulación donde un vehículo que transita a través de un área densamente poblada sufrirá una variedad de errores y tendrá que hacer frente a ellos.

El presente trabajo se organiza de la siguiente manera. En la Sección II, se explica la estructura de localización propuesta. En la Sección III se ofrece una detallada descripción del escenario de validación y un análisis de los resultados obtenidos. Finalmente, en la sección IV se resumen las ideas más importantes de este trabajo.

2. ESTRUCTURA DEL SISTEMA DE LOCALIZACIÓN

La estructura de localización propuesta se expone en la Figura 1. Se compone de 3 diferentes fuentes de información de entrada: GPS/INS, LiDAR y la odometría, la cual consiste en la velocidad de las ruedas y el ángulo de giro del volante. Además de esto, se hace uso de un detallado mapa digital del entorno de conducción. Al mismo tiempo, se implementa un bloque de detección de errores con el objetivo de desenmascarar módulos erróneos y transmitir la información al EKF trabajando como módulo de toma de decisiones.

Esta propuesta considera la estimación del módulo GPS/INS como la más fiable en circunstancias normales. Aun así, en paralelo, se implementan dos estrategias de localización: un algoritmo de *map matching* en conjunto con el LiDAR y la odometría; y una técnica de *dead reckoning* [13] basada en las lecturas de la odometría. Las tres estimaciones son comparadas por un módulo de detección de errores, el cual permite usar la estimación más fiable en cada caso en base a la detección de señales erróneas.

En consecuencia, la estrategia de localización puede dividirse en 4 etapas, tal y como se describen a continuación (Figura 1).

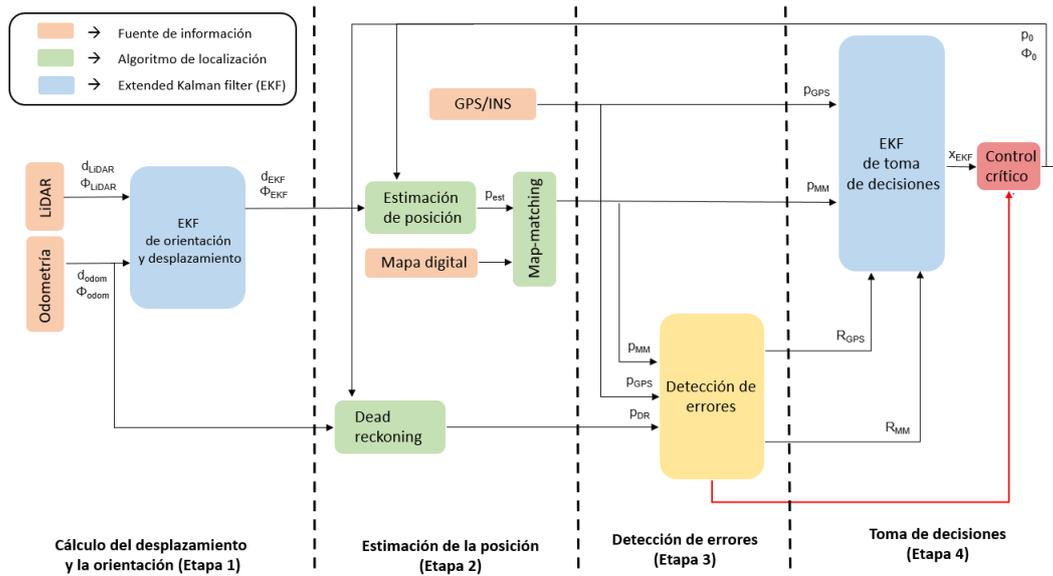


Figura 1: Estructura de localización

2.1. Cálculo del desplazamiento y la orientación

El desplazamiento y la orientación se calculan en base a las mediciones del LiDAR y la odometría. Para el primero, se aplica un algoritmo de *point-cloud matching*, donde el algoritmo de registro se basa en el *Normal-Distributions Transform* (NDT) [11]. De esa forma, se calcula el ángulo *Yaw*, ϕ_{LiDAR} , y el desplazamiento, d_{LiDAR} . De forma parecida, para la odometría, se calculan los respectivos valores del ángulo *Yaw*, ϕ_{odom} , y la odometría, d_{odom} , en base a la integración de las mediciones de la velocidad y el ángulo de giro del volante, asumiendo un modelo de bicicleta para el vehículo según se describe en [13]. Ambos valores de desplazamiento y ángulo *Yaw* son introducidos entonces en un EKF, el cual proporciona una estimación más precisa de ambas variables, ϕ_{EKF} and d_{EKF} .

2.2. Estimación de la posición

Tal y como se expuso en la sección anterior, en la estructura propuesta se implementan 3 estrategias de localización distintas.

La primera estrategia se basa en el uso de un módulo GPS/INS, el cual combina las estimaciones del GPS con las mediciones del INS en un filtro de Kalman integrado [18]. Se asume que esta señal no necesita más computación que la expuesta hasta ahora.

La segunda estrategia parte de la salida en forma de desplazamiento y ángulo *Yaw* del EKF, ϕ_{EKF} y d_{EKF} . Usando estas mediciones, se calcula una

estimación global del vehículo, p_{est} , mediante las siguientes ecuaciones:

$$\begin{bmatrix} x \\ y \end{bmatrix} = d \cdot \begin{bmatrix} \cos(\phi + \phi_0) \\ \sin(\phi + \phi_0) \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \quad (1)$$

donde se usan las estimaciones globales que proporciona la estructura de localización en los instantes anteriores (x_0 , y_0 y el ángulo *Yaw* ϕ_0) como posiciones iniciales conocidas para el cálculo relativo.

Este valor calculado se combina en un algoritmo *point-to-point* geométrico de *map matching* [14], mediante el cual se puede estimar la localización global del vehículo. Esta aproximación busca el punto más cercano a la red de carreteras del mapa digital, p_{MM} , respecto a la estimación correspondiente a la posición de la relativa, p_{est} . De esa forma, usando la distancia Euclídea como criterio, el algoritmo logra mejorar la estimación de p_{est} , posicionando el vehículo de forma global en el mapa con la estimación p_{MM} .

Con el objetivo de reducir la deriva inherente a las estimaciones, se propone una sencilla corrección basada en la identificación de elementos del entorno de la carretera o *feature navigation*; en particular, mediante la detección de cruces de carreteras. El algoritmo de corrección detecta la presencia de un cruce siempre que la variación, o valor derivado, del ángulo *Yaw* supera un umbral. Entonces, con la estimación del *map matching* como punto de partida, p_0 , se busca una nueva posición en la carretera, p' , con un ángulo *Yaw* más cercano al calculado y alrededor de una distancia de

control, d ; tal y como se explica en detalle en el Algoritmo 1. La distancia de control d es seleccionada buscando un compromiso entre dos aspectos: la cantidad mínima de puntos necesarios para una correcta identificación de la posición real; y el exceso de puntos a analizar, lo cuál puede aumentar el coste computacional hasta el punto de ralentizar la ejecución el algoritmo. De esta forma se obtiene una solución para 2 fuentes de errores de todos los algoritmos de *map matching*: errores en la elección de carreteras cuando muchas carreteras se encuentran; y la deriva longitudinal que se acumula con el tiempo, la cual es compensada gracias al algoritmo descrito en presencia de cruces. Estas suposiciones son ciertas siempre y cuando la carretera se encuentre libres de obstáculos, lo cuál entra dentro del presente marco.

Algoritmo 1 Corrección de cruces

```

Require:  $\Delta_\phi \geq threshold$ 
 $p_0, \phi_0 \leftarrow$  Estimación map-matching previa
 $d \leftarrow$  Distancia de control desde estimación  $p_0$ 
for all puntos del mapa  $x'$  a la distancia  $d$  desde  $p_0$  do
     $\phi' \leftarrow$  Orientación de los nuevos puntos  $p'$ 
    if  $|\phi' - \phi_0| < \Delta_\phi$  then
         $p \leftarrow p'$ 
    else
         $p \leftarrow p_0$ 
return  $x$  nueva estimación
    
```

Por último, el tercer algoritmo de localización se basa en una técnica de *dead reckoning* similar a la descrita en [10], y basada en los valores de desplazamiento y ángulo *Yaw* previos al EKF, ϕ_{odom} and d_{odom} . De esta forma se obtiene la estimación de la odometría, p_{DR} . Es importante reseñar que esta estimación tan solo será utilizada por el bloque de detección de errores para desenmascarar fallos en la señal GPS; pero nunca como estimación de la estructura de localización.

2.3. Detección de errores

El algoritmo de detección de errores, descrito en la Figura 2, se basa en una técnica de duplicación-comparación que calcula la precisión dinámica de cada fuente [4]. De esa forma, el algoritmo puede dividirse en 3 etapas que contienen las siguientes modificaciones respecto al trabajo referenciado:

- *Cálculo de conflicto*: el conflicto se define como la diferencia relativa entre dos estimaciones de distintas estrategias de localización. En este trabajo, se utilizan los valores absolutos y derivados en el tiempo para calcular dicho conflicto. Estos últimos valores son especialmente interesantes para una rápida de-

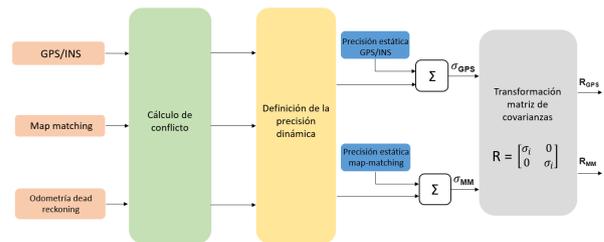


Figura 2: Algoritmo de detección de errores

tección de errores, ya que los valores derivados tienden a crecer a mayor velocidad en presencia de conflicto entre fuentes.

- *Definición de la precisión dinámica*: la precisión dinámica es un valor escalar dependiente del tiempo que pretende medir la fiabilidad de una fuente de información de forma instantánea. Para ello, se basa en el conflicto que existe entre dicha fuente y el resto; penalizando su medida si el conflicto con todas las demás fuentes crece. De esa forma, el conflicto se utiliza para evaluar si ha ocurrido algún tipo de error. Esta estrategia ofrece una doble comprobación en situaciones como giros bruscos en curvas o cruces, donde puede haber grandes variaciones en las derivas mientras los valores absolutos se mantienen constantes.
- *Precisión global (σ_i)*: cada fuente tiene su propia precisión estática, basada en la calidad y precisión esperada del sensor. De esa forma, tanto la precisión dinámica como la estática se combinan para obtener la precisión global en un instante en particular. Esta precisión global se representa mediante un valor escalar y posteriormente se transforma a la matriz de covarianza correspondiente al ruido de las mediciones para cada fuente, R , y se combina en un EKF. Esta transformación, tal y como se describe en la Figura 2, permite el uso del EKF como un módulo de toma de decisiones, lo cual es una contribución importante de este trabajo.

El algoritmo de error propuesto necesita un mínimo de 3 fuentes diferentes de información para detectar un error en el sistema [15]. En este contexto, el caso de estudio consiste en detectar errores en el GPS/INS; para lo cual, se utilizará el conflicto entre las siguientes 3 señales: el GPS/INS, el *map-matching* y la odometría en combinación con el *dead reckoning*. Aun así, esta propuesta no se restringe solamente a esta estructura, ya que puede ser ampliada a diferentes cantidades y combinaciones de fuentes de información y errores.

Escenario	Longitud (m)	Errores detectados / reales / falsos positivos	$error_{lat}$ (m) max.	$error_{long}$ (m) max.	$error_{lat}$ (m) medio	$error_{long}$ (m) medio
1	878.78	15 / 15 / 0	0.5985	3.5402	0.2683	0.6284
2	878.78	6 / 6 / 0	0.7855	5.2408	0.3031	1.4186
3*	511.79	6+1 / 6+1 / 0	0.545	5.153	0.3394	1.3543

Cuadro 1: Métricas de error para los escenarios (el Escenario 3 incluye un error crítico).

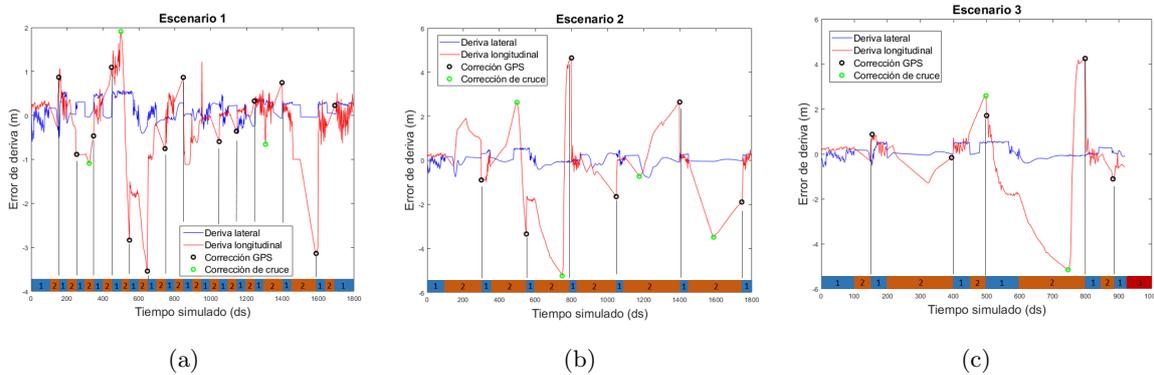


Figura 3: Deriva lateral y longitudinal para los 3 escenarios.

2.4. Toma de decisiones

En la última etapa de la estructura de localización, se introduce un EKF que combinará las tareas de fusión de información y toma de decisiones. Este módulo combina las estimaciones de la posición del *map matching* y del GPS/INS. Además de eso, es capaz de elegir como salida la estimación de la localización más precisa en todo momento. Para ello, se basa en los valores de la precisión global calculados en el bloque de detección de errores, que posteriormente son introducidos en forma de matriz de covarianza, *R*.

Además del módulo GPS/INS, en las simulaciones se han contemplado posibles errores o desviaciones en el resto de sistemas de localización. Por tanto, se ha implementando una señal crítica para cuando el conflicto entre todas las señales crece por encima del límite y el sistema de localización no puede funcionar correctamente con el objetivo de ordenar la detención del vehículo.

3. VALIDACIÓN

Con el objetivo de validar la estructura propuesta se hace uso del entorno de simulación Carla [5]. Se ha partido de un modelo Tesla 3 como caso de estudio, equipado con GPS/INS, LiDAR y sensores de odometría tanto para la velocidad como el giro del volante. La información obtenida en Carla es transferida mediante la Python API a Matlab/Simulink, donde la estrategia de localización será implementada y validada.

Se han propuesto 3 escenarios en este entorno, todos ellos incluyendo el mismo recorrido tal y como se describe en la Figura 4. El primer escenario considera fallos intermitentes de corta duración (10 s) en el módulo GPS/INS, con el objetivo de probar la funcionalidad del bloque de detección de errores. En el segundo escenario, la duración de los errores se extiende hasta los 30s, para validar la precisión del algoritmo *map matching*. Por último, en el tercer escenario se implementan errores en el módulo del GPS/INS y en el de la odometría, llevando el vehículo a estado crítico.

Los resultados para los 3 escenarios se resumen en la Tabla 1. De esta tabla se pueden obtener 2 conclusiones principales: 1) la estructura de localización es consistente en la tarea de detectar

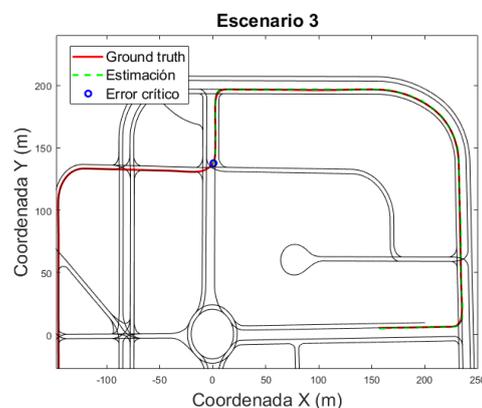


Figura 4: Escenario y trayectoria (correspondiente al Escenario 3)

toda la variedad de posibles errores en el sistema; y 2) la deriva del error de posicionamiento siempre permanece acotada, tanto en valores medios a lo largo de todo el trayecto como en picos de error. En la Figura 3 se muestra en detalle la evolución de los errores longitudinales (en el mismo eje del movimiento) y laterales (en el eje perpendicular al movimiento) en las tres simulaciones. En estas figuras, los diferentes modos de funcionamiento a lo largo de las simulaciones se caracterizan con una barra de colores horizontal que determina el error actual en el sistema y el modo de funcionamiento así como su extensión en el tiempo. El *Modo 1*, en azul, corresponde a un funcionamiento normal del GPS, el *Modo 2*, en naranja, a la estrategia de localización alternativa con el algoritmo *map matching*, y el *Modo 3*, en rojo, a un fallo crítico y la detención del vehículo.

Para el Escenario 1, donde se han simulado la mayor cantidad de errores, los resultados de la Tabla 1 prueban que el bloque de detección de errores trabaja de forma consistente. En cuanto a la deriva del error, tal y como se ve en la Figura 3a, los valores tanto longitudinales como laterales permanecen acotados, principalmente por la frecuente reconexión del GPS/INS y su corrección correspondiente; siempre que el Modo de funcionamiento cambia al Modo 1, reflejado en azul en las Figuras.

En el escenario 2, los errores en el GPS se extienden hasta los 30s. Durante ese periodo, la estructura de localización usa como salida el algoritmo de *map matching*. Tal y como se puede ver en la Tabla 1, los errores durante estos periodos permanecen acotados. Es más, tal y como se explicó en la sección anterior, el error longitudinal es corregido en la fase de *map matching* mediante el algoritmo de detección de cruces; tal y como se comprueba en los instantes $t = 500ms$, $t = 1200ms$ o $t = 1600ms$ de la Figura 3b. Esta corrección siempre se evalúa en el Modo 2, de color naranja en las Figuras, ya que requiere de que el mapa digital esté disponible y una variación del ángulo *Yaw* sea detectada por la presencia de un cruce. Por otro lado, en el instante $t = 750ms$, el algoritmo de detección de cruces se activa, corrige la deriva longitudinal, pero inmediatamente después de la recuperación, el algoritmo de *map matching* empieza a acumular error de nuevo debido a errores en el cálculo del desplazamiento y el ángulo *Yaw* durante ese periodo. Con el tiempo, el GPS se reconecta y permite una corrección de dicha deriva. Por tanto, el algoritmo de detección de cruces ha demostrado que mejora la precisión de la estimación y estabiliza la estructura de localización eliminando la excesiva desviación siempre que el GPS no está disponible.

Por último, en el Escenario 3, se simulan los errores de los dos escenarios previos junto con un fallo crítico. Tal y como se describe en la Figura 3c, los errores longitudinales y laterales permanecen acotados a lo largo de la trayectoria del vehículo; con un comportamiento similar al descrito para el Escenario 2 en los instantes $t = 500ms$ y $t = 750ms$.

Además de lo anterior, respecto al modulo de detección de errores, en la Figura 5 se pueden ver las 6 señales de conflicto para el Escenario 3; con 4 señales, 2 absolutas y 2 derivadas en el tiempo, para cada estimación. Tal y como se expuso en secciones anteriores, para desenmascarar una fuente errónea se aplica un protocolo de doble verificación; es decir, tanto las 2 señales absolutas como las 2 señales derivadas en el tiempo deben superar sus correspondiente límites al mismo tiempo. Siempre que esto ocurre, la Figura 5 lo señala con una línea roja. Los resultados prueban que el fallo de la señal GPS es detectado siempre que se implementa un error en el sistema, ya que tanto las señales de control absolutas como derivadas en el tiempo superan sus límites al mismo tiempo; mientras que el resto de señales correspondientes al *map matching* o a la odometría no se solapan. Por otro lado, tal y como se describe en los últimos instantes ($t > 900ms$) de la Figura 5, en presencia de un error crítico, donde coexiste más de un sistema en fallo, todas las señales de control de todas las fuentes superan el límite y, por tanto, se envía una señal de detección de vehículo.

Los resultados descritos hasta ahora prueban la validez de la propuesta. Es más, al contrario de los sistemas tolerantes a errores que se encuentran en la literatura, esta propuesta no solo se centra en detectar correctamente los errores a lo largo de la trayectoria, sino que además acota el efecto de dichos errores mejorando la precisión del sistema. Las métricas de errores obtenidas en las simulaciones se han comparado con trabajos de localización parecidos basados en algoritmos de *map matching* en escenarios equivalente [17][3], y los resultados demuestran que la precisión obtenida es correspondiente a otras propuestas de la literatura (5-7 metros para el error longitudinal máximo, y 1-2 para los valores medios del error).

4. CONCLUSIONES

Este trabajo propone una estructura de localización tolerante a errores para vehículos autónomos en áreas densamente pobladas, donde la fiabilidad de los sensores GPS/INS suele estar comprometida. Se ha diseñado un algoritmo de *map matching* como técnica alternativa de localización durante bloqueos de la señal GPS. Además, se ha implementado un algoritmo de cruces para abordar las

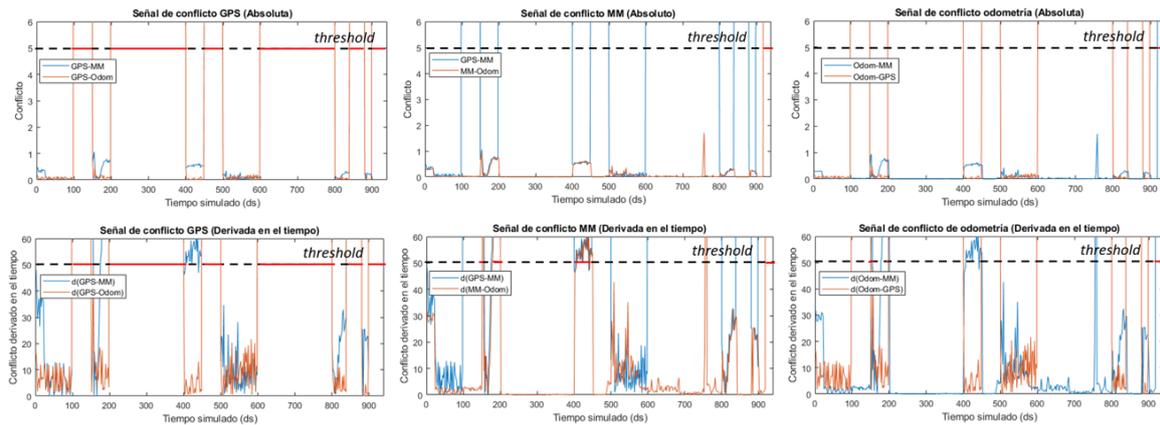


Figura 5: Señales de conflicto, absolutas y derivadas para todas las fuentes a lo largo del Escenario 3.

limitaciones más comunes de las técnicas basadas en mapas y relacionadas con la deriva longitudinal. La validación de la estructura, en base a un entorno de simulación, ha demostrado excelentes resultados en cuanto a la detección de errores, validando el uso del EKF como módulo de toma de decisiones; el cual está basado en una estimación de la fiabilidad dinámica de las fuentes y su posterior implementación a la matriz de covarianzas del ruido en las mediciones. Como trabajo futuro, se estudiará la implementación de mapas digitales más detallados en combinación con la detección y segmentación de objetos mediante sensores visuales. Esto permitirá incluir más elementos del entorno de conducción para una mejor corrección de la deriva longitudinal.

Agradecimientos

Este trabajo es subvencionado por la Universidad del País Vasco, UPV/EHU (proyecto GIU19/045 y beca PIF19/181), y por el Gobierno Vasco mediante la subvención IT914-16 y el proyecto AUTOEVAL KK-2021/00123.

English summary

SENSOR FUSION FOR ROBUST LOCALIZATION OF AUTONOMOUS VEHICLES IN URBAN AREAS

Abstract

Localization in dense urban areas may be difficult, as GPS signal usually gets blocked by buildings or tunnels. Therefore, autonomous vehicles need to implement alternatives for those circumstances with a fault-tolerant localization structure. This

is an area of great interest that most works address by combining duplication-comparison techniques and the use of belief functions and alternative localization algorithms such as map-matching or visual odometry. This paper proposes a localization framework for dense urban areas that includes both a robust fault-tolerant algorithm capable of determining the reliability of every position estimation, and a computationally low demanding map matching algorithm as an accurate alternative localization strategy. Validation on simulation environment has proved that the proposed approach is functional.

Keywords: Autonomous driving, sensor fusion, error detection, localization.

Referencias

- [1] M. B. Alataise and G. P. Hancke. A review on challenges of autonomous mobile robot and sensor fusion methods. *IEEE Access*, 8:39830–39846, 2020.
- [2] M. Brossard, A. Barrau, and S. Bonnabel. AI-IMU Dead-Reckoning. *IEEE Transactions on Intelligent Vehicles*, 5(4):585–595, 2020.
- [3] Lucas de Paula Veronese, Claudine Badue, Fernando Auat Cheein, Jose Guivant, and Alberto Ferreira De Souza. A single sensor system for mapping in GNSS-denied environments. *Cognitive Systems Research*, 56:246–261, 2019.

- [4] F. Delmotte and G. Gacquer. Detection of defective sources with belief functions. In *2th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pages 337–344, Malaga, Spain, 2008.
- [5] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017.
- [6] P. J. Escamilla-Ambrosio and N. Mort. A hybrid Kalman filter-fuzzy logic architecture for multisensor data fusion. In *IEEE International Symposium on Intelligent Control*, pages 364–369, Mexico City, Mexico, 2001.
- [7] S. Grubmüller, G. Stettinger, M. A. Sotelo, and D. Watzenig. Fault-tolerant environmental perception architecture for robust automated driving. In *IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, pages 1–6, Graz, Austria, 2019.
- [8] J. Guivant, M. Whitty, and A. Robledo. *Robust Global Urban Localization Based on Road Maps*, chapter 14, pages 267–284. IntechOpen, 2010.
- [9] B. Kaci, L. Benjamin, and S. Walter. A fault tolerant architecture for data fusion targeting hardware and software faults. In *IEEE 20th Pacific Rim International Symposium on Dependable Computing*, pages 1–10, Singapore, 2014.
- [10] M. Kochem, N. Wagner, C. Hamanna, and R. Isermann. Data fusion for precise Dead-Reckoning of passenger cars. In *IFAC*, volume 35, pages 397–402, 2002.
- [11] M. Magnusson. *The Three-Dimensional Normal-Distributions Transform — an Efficient Representation for Registration, Surface Analysis, and Loop Detection*. PhD thesis, Örebro University, Örebro, Sweden, 2013.
- [12] M. Magnusson, H. Andreasson, A. Nuchter, and A. J. Lilienthal. Appearance-based loop detection from 3D laser data using the Normal Distributions Transform. In *IEEE International Conference on Robotics and Automation*, pages 23–28, Kobe, 2009.
- [13] Jose A. Matute, Mauricio Marcano, Sergio Diaz, and Joshue Perez. Experimental validation of a kinematic bicycle model predictive control with lateral acceleration consideration. *IFAC-PapersOnLine*, 52(8):289–294, 2019.
- [14] M. Quddus. *High Integrity Map Matching Algorithms for Advanced Transport Telematics Applications*. PhD thesis, Imperial College London, 2006.
- [15] V. Ricquebourg, M. Delafosse, L. Delahoche, B. Marhic, A.M. Jolly-Desodt, and D. Menga. Fault detection by combining redundant sensors: a conflict approach within the TBM framework. In *COGNITIVE systems with Interactive Sensors (COGIS)*, Standford, 2007.
- [16] P. Smets. Data fusion in the transferable Belief Model. In *3rd International Conference on Information Fusion*, volume 1, pages 21–33, 2000.
- [17] J. K. Suhr, J. Jang, D. Min, and H. G. Jung. Sensor fusion-based low-cost vehicle localization system for complex urban environments. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1078–1086, May 2017.
- [18] T. P. Van, T. N. Van, D. A. Nguyen, T. C. Duc, and D.-T. Tran. 15-state extended Kalman filter design for INS/GPS navigation system. *Journal of Automation and Control Engineering*, 3(2):109–114, 2015.
- [19] J. Wahlström, I. Skog, J. G. P. Rodrigues, P. Händel, and A. Aguiar. Map-aided Dead-Reckoning using only measurements of speed. *IEEE Transactions on Intelligent Vehicles*, 1(3):244–253, 2016.
- [20] M. Yan, J. Wang, J. Li, and C. Zhang. Loose coupling visual-lidar odometry by combining VISO2 and LOAM. In *36th Chinese Control Conference (CCC)*, pages 6841–6846, Dalian, 2017.
- [21] Z. Zhang and Y. Shan. Incremental motion estimation through modified Bundle Adjustment. In *International Conference on Image Processing*, pages II–343, Barcelona, Spain, 2003.



© 2021 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution CC BY-NC-SA 4.0 license (<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).