UNIVERSIDAD DE SEVILLA

DEPARTAMENTO DE LENGUAJES Y SISTEMAS INFORMÁTICOS

Técnicas de predicción de destinos geográficos futuros en desplazamientos de personas

Memoria de Tesis Doctoral presentada por
D. Juan Antonio Álvarez García
para la obtencion del grado de Doctor en Informática,
dirigida por los doctores:
D. Juan Antonio Ortega Ramirez
y **D. Luis González Abril**.

Sevilla, diciembre de 2009.

A mi mujer que ha sabido animarme en cada momento
y a mi hija que ha traído más alegría aún a nuestro hogar.

Agradecimientos

Quisiera expresar mi más sincero agradecimiento a mis directores de Tesis, Dr. D. Juan Antonio Ortega Ramírez y Dr. D. Luis González Abril, por todo el esfuerzo y tiempo dedicado a la elaboración de este trabajo. Del mismo modo, a Francisco Velasco y Francisco Javier Cuberos por su charla instructiva y amena.

Destacar también el apoyo de mis compañeros y amigos del Departamento de Lenguajes y Sistemas Informáticos, especialmente de Fernando Enríquez, Toñi Reina, María Teresa Gómez, Alejandro Fernández, José Antonio Troyano, Miguel Ángel Álvarez y Luis Miguel Soria.

Asimismo, mi agradecimiento a los compañeros de la UPC Dr. Cecilio Angulo, Dr. Andreu Català, Dr. Diego Esteban Pardo y a Carlos Pérez que me acogieron como a un amigo en mis fructíferas estancias en Vilanova.

Finalmente, un agradecimiento muy especial a mis padres y a mi hermano por su cariño, ánimos y ayuda.

A todos, muchas gracias.

ÍNDICE GENERAL

Índice

1

Introducción

2

Metodología off-line

3
Modelo de Markov Oculto sobre "Mapa Soporte"

4
Similitud de recorridos

5
Resultados

6

Predicción on-line

A

Sistema GPS

B

Curriculum

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

CAPÍTULO 1

INTRODUCCIÓN

Entre 1990 y 1991, el ejército de los Estados Unidos utilizó el Sistema de Posicionamiento Global (GPS) [Get93] en la Guerra del Golfo. Su desarrollo, inspirado en la localización del satélite Sputnik en el espacio mediante el efecto Doppler, comenzó formalmente en 1973 y no se completa hasta 1995.

Este sistema abrió las puertas al desarrollo de aplicaciones de localización en exteriores. Su uso inicial por parte del Departamento de Defensa estadounidense y el error aleatorio que introducían para el uso civil (técnica llamada "Disponibilidad Selectiva" y presente desde 1990), limitó su utilidad en investigación no militar.

La eliminación de la "Disponibilidad selectiva" el 2 de mayo del año 2000 (la precisión del sistema pasó de 100 a 20 metros), la bajada de precios en los receptores GPS, la miniaturización de éstos y la orden de la Comisión de Comunicaciones Federal estadounidense E911[(1)], relanzaron el interés por las aplicaciones basadas en la localización ("*Location Based Applications*") en exteriores.

[(1)] Orden aplicada desde 2001 que obliga a los proveedores inalámbricos a localizar dentro de decenas de metros a usuarios que realizan llamadas de emergencia. Más tarde Europa propuso la recomendación europea E112 (aplicada desde 2003) con el mismo propósito.

Durante el siglo XXI, la progresión del mercado de los receptores GPS ha sido tremenda. Según recientes estudios[2] la población mundial de suscriptores de servicios de localización crecerá desde los 12 millones de 2006 a los 315 millones en 2011.

Los factores para que el mercado de chips GPS se haya disparado se fundamenta en gran medida en los sistemas de guiado o navegación para vehículos[3] que ha provocado una estrecha alianza entre productores de navegadores GPS y empresas de cartografía, tanto es así que muchas de las segundas han sido adquiridas por las primeras[4].

Uno de los sistemas que más acogida ha tenido relacionado con los sistemas de localización ha sido el de gestión de flotas de empresas de transporte. Explotando la combinación de receptores GPS y antenas GSM es posible conocer en todo momento los movimientos de toda una flota de cualquier tipo de vehículo. Esta idea se ha extendido al seguimiento de niños[5], de ancianos[6], presos en régimen abierto como maltratadores[7] o incluso de coches robados mediante proyectiles[8] GPS que evitan peligrosas persecuciones.

Sin embargo la integración de los chips GPS en los terminales móviles (9 de cada 10 móviles contendrán chips GPS en el 2014 comparado con los 1 de cada 3 del 2008[9]) ha hecho que las aplicaciones con conocimiento de la localización se multipliquen. De ese modo encontramos aplicaciones de publicidad por localización[10] o localización de lugares de interés mediante realidad aumentada[11].

[2] http://www.abiresearch.com/abiprdisplay.jsp?pressid=766.
[3] http://reviews.cnet.com/best-gps/
[4] http://online.wsj.com/article/SB119677803171513059.html?mod=googlenews_wsj
[5] http://www.laipac.com
[6] http://www.keruve.com
[7] http://www.elmundo.es/elmundo/2009/07/08/espana/1247048060.html
[8] http://www.starchase.org/
[9] http://cp.gpsworld.com/gpscp/Latest+News/ABI-GPS-Enabled-Handsets-Will-Byass-Economic-Downt/ArticleStandard/Article/detail/580207?contextCategoryId=1385
[10] http://www.gpsworld.com/gps/the-business-location-driven-coupons-iphone-8657
[11] http://www.youtube.com/watch?v=b64_16K2e08

Por último, los proyectos de Rusia (Glonass), Europa (Galileo), China (Beidou), India (IRNSS) o Japón (QZSS) que desarrollan sus propias redes de satélites así como los sistemas de mejora de la señal basados en satélites como EGNOS, WAAS para Estados Unidos, MSAS para Japón y GAGAN para la India hacen prever un uso mucho más intensivo de la localización en exteriores en los próximos años. De hecho ya se ofertan productos con precisión de centímetros[(12)] para guiado automático de vehículos agrícolas.

1.1. Propuesta

A pesar de todos estos avances en la industria e investigación, estamos convencidos, y en este sentido va dirigido este trabajo, que se pueden crear **aplicaciones con conocimiento de la localización futura**, mucho más ricas en información para el usuario si conseguimos predecir sus rutas y destinos durante sus desplazamientos en exteriores de manera rápida y precisa. Para ello nuestra investigación se ha basado en el conjunto de los recorridos realizados por cada usuario en el pasado.

Dado que el comportamiento de la mayoría de las personas en sus desplazamientos es repetitivo y regular, es decir, nos dirigimos normalmente a los mismos destinos utilizando las mismas rutas, las necesidades para realizar predicciones en cuanto a mapas y cartografía no son las mismas que en un sistema de navegación y guiado de vehículos. Por ello en este trabajo se proponen **modelos predictivos** utilizando técnicas de aprendizaje supervisado basadas en **mapas personales** generados por los propios usuarios. Como comprobaremos, estos mapas personales permiten una drástica reducción en el coste de almacenamiento con respecto a los Sistemas de Información Geográfica (GIS) y, lo más importante, su especialización.

Para validar los modelos predictivos desarrollados se ha planteado una **metodología *off-line*** que permite verificar los resultados de cada uno de ellos. Una vez verificados los modelos se describe una posible implementación en teléfonos móviles

[(12)] http://www.outbackguidance.com/Default.aspx?tabid=431

con capacidades GPS que permite la **predicción *on-line***, es decir, obtener la predicción mientras el usuario se desplaza. Además se consigue la recuperación de los recorridos para que el modelo se retro-alimente con cada nuevo desplazamiento.

Por último, dado el elevado consumo de baterías de los chips GPS, se propondrán **técnicas para incrementar la vida útil del dispositivo móvil** entre cada recarga.

1.2. Escenario motivador

La predicción de rutas y destinos futuros supone una interesante fuente de aplicaciones que permiten adelantar conocimiento, tanto personal como social, de manera que podamos optimizar nuestras tareas diarias, evitando interrupciones y situaciones imprevistas. Veamos como ejemplo ilustrativo el siguiente escenario:

Julia sale de su despacho algo más tarde de lo habitual, a las 19:50 y se dirige hacia su coche con la intención de ir a casa. En cuanto sale al exterior, su dispositivo comprueba la localización de su familia (accediendo a cada uno de sus móviles) al tiempo que se actualiza su localización automáticamente con la etiqueta "saliendo del despacho". El dispositivo le indica por pantalla que su marido se encuentra en el trabajo (esa indicación es suficiente para ella por lo que no necesita un mapa que lo localice).

La noche anterior, su vivienda inteligente generó una lista de la compra y la transmitió a su terminal móvil. Como el terminal es capaz de intuir hacia donde se dirige Julia en cuanto lleva 2 minutos conduciendo, éste consulta de manera autónoma la información del tráfico de las rutas típicas que suele seguir. El tráfico es denso en todas ellas aunque la previsión consultada indica que media hora después la circulación será mucho más fluida. Así que le propone 2 opciones: utilizar una ruta alternativa (que no realiza con asiduidad) o detenerse en el supermercado al que suele ir en el que realizar sus compras y que se encuentra aproximadamente a 5

minutos del punto actual y a 20 minutos de casa. Como le gusta realizar la compra personalmente y el tráfico será previsiblemente más fluido un poco más tarde, elige la segunda opción. Cambia entonces su ruta y aparca en el parking del supermercado.

A las 20:10 Julia sigue en el supermercado y como acostumbra a llegar a casa sobre las 20:30, el dispositivo le avisa de que tal vez fuese interesante enviar un SMS a su marido que acaba de salir de una reunión. El aviso en este caso es escrito (la información es privada y no le gustaría que la gente le viese hablando con su móvil). El SMS ha sido ya redactado "Estoy en el 'supermercado de la avenida' comprando, tardaré unos 20 minutos en llegar a casa". Ella revisa el mensaje, cambia 20 por 30 (aún no ha terminado de comprar y hay cola en la caja), lo envía y poco después recibe confirmación de que ha sido leído.

Jesús, su marido, tras recibir el aviso decide ir a correr un poco antes de ir a casa, se cambia en el gimnasio de la empresa y sale a la calle por una nueva ruta que no ha hecho antes. Su terminal inteligente comprueba que se desplaza rápidamente pero no lo suficiente como para ir en bici o coche, entiende que está corriendo y le solicita su atención para elegir entre dos opciones: "guiado hacia un determinado lugar" o "programa de entrenamiento". Jesús pulsa sobre su pantalla el segundo y el dispositivo le informa de la velocidad, metros recorridos y le muestra una gráfica comparativa con sus últimas 2 carreras. Además al detectar que ha conectado los auriculares, escoge la carpeta de mp3's etiquetada como "deporte" para que incremente el ritmo.

Jesús termina la carrera pronto (40 minutos) se ducha y regresa a la vivienda, antes de comenzar a conducir consulta en su móvil la posición de su esposa, ve que su estado es "dirigiéndose a casa, tiempo estimado de llegada 10 minutos" así que llegará a tiempo para ayudarle a descargar la compra.

Como hemos comprobado, el único momento en el que en el escenario se ha podido necesitar un GIS ha sido cuando se le ofertó a uno de los usuarios un sistema de guiado, pero en esos casos, lo que haríamos sería iniciar una aplicación de navegación específica. Las predicciones, el etiquetado de los lugares y la comunicación sin información privada (se ha evitado utilizar coordenadas para la comunicación), se

pueden conseguir con una correcta identificación de destinos y rutas basados en una base de datos personal alojada en cada uno de los dispositivos. Por otra parte, la consecución de un sistema integrable en un móvil, que economice baterías, pudiendo hacer consultas cortas a Internet y a otros terminales móviles resulta un reto muy interesante. Para llegar a implementarlo, es necesario definir una metodología que combine algoritmos de extracción de información, clasificación y predicción. Una vez evaluadas las técnicas debemos conseguir que el terminal móvil tenga consciencia de la localización en todo momento tanto presente como futura.

Nuestra aproximación al problema es un modelo de predicción global, con supervisión de destinos con el objetivo de predecir rutas además de destinos. Aunque parte del mismo prueba la eficiencia de los algoritmos en diferido, también nos centramos en los aspectos prácticos de la predicción en tiempo real.

1.3. Contexto

La predicción de destinos de personas ha sido estudiada en los últimos años siguiendo diferentes líneas de investigación como transporte inteligente, robótica, telecomunicaciones o sistemas de información geográfica entre otros. En transporte inteligente, se desarrollan sistemas de ayuda a la conducción eficientes [KB03, TMK+06, TKTN09]; en robótica, se aplican algoritmos cuyo objetivo es la predicción de destinos en interiores para el seguimiento de personas por robots que les ayudan en su vida diaria [BBT02, VF04, VFAL05, OMM02]; en telecomunicaciones, ha tenido un papel importante para la gestión de la calidad de servicio y del ahorro de energía en los handoff o handover[13] de los usuarios con dispositivos móviles [LBC98, SKJH06, MW06]; en sistemas de información geográfica, el objetivo ha sido el de predecir situaciones de colapso del tráfico a partir del rastreo de las posiciones de vehículos almacenadas en bases de datos de objetos en

[13] Sistema utilizado en comunicaciones móviles celulares con el objetivo de transferir el servicio de una estación base a otra cuando la calidad del enlace es insuficiente. Este mecanismo garantiza la realización del servicio cuando un móvil se traslada a lo largo de su zona de cobertura.

movimiento [BPT04, BJ07, CJP05, FM09]. Incluso en biología, la predicción de destinos aparece en proyectos de seguimiento de la vida animal a través de su hábitat para poder estudiar hacia donde migran, realizar observaciones más precisas y considerar el impacto que tienen en su entorno en las diferentes épocas del año [oML02, fCB02, SLBB04, ZSML04].

Sin embargo, nuestro enfoque tiene como protagonista al usuario y el objeto de nuestro sistema es que a éste le sea útil en su vida cotidiana. Por ello, la mayoría de las referencias provienen del campo de la computación ubicua, nacida a partir de las revolucionarias ideas de Mark Weiser [Wei91] y su equipo de Xerox Parc [SAG+93, WSA+95]. Dentro de este campo encontramos los trabajos más relacionados con nuestra tesis y para poder encuadrar nuestras aportaciones de una manera más específica y clara, pasamos a clasificar estos trabajos según diferentes criterios.

Ámbito geográfico. Diremos que el sistema es local si sólo podemos realizar predicciones sobre los destinos que se encuentran en una zona determinada y limitada inicialmente. En caso de que no existan límites, es decir que podamos predecir cualquier destino en cualquier lugar del globo terráqueo, diremos que estamos ante un sistema global. Ejemplos de sistemas locales son aquellos que permiten predicciones en una única ciudad o en zonas concretas como un Campus Universitario. La localidad suele producirse al depender de un GIS que incluye datos de calles, carreteras y edificios de la zona limitada. Dado que la construcción y mantenimiento de un GIS que incluya toda esa información del globo terráqueo es muy costosa y poco manejable, las propuestas locales difícilmente pueden ampliarse a globales.

Supervisión de la predicción de destinos. Para realizar una predicción los sistemas clasifican de manera temprana los nuevos recorridos. Esta clasificación puede ser supervisada o no supervisada. En el primer caso el sistema tiene una base de conocimiento de recorridos pasados y sus rutas representantes etiquetadas por el destino que alcanzan. De ese modo, no podrán predecirse destinos que no están en la base de conocimiento. En el segundo caso, no se

tiene información de recorridos pasados, sino de datos que hacen más probable una zona como destino que otra. Normalmente los sistemas no supervisados son locales y además suministran una predicción con un rango de error importante, sin embargo, no es necesario almacenar recorridos previos durante un determinado periodo de tiempo.

Objetivo final. En este caso destacamos los sistemas a corto y a largo plazo. Los primeros tienen como objetivo descubrir cuál será la próxima calle a la que se dirigirá el usuario, suelen aplicarse a sistemas de transporte inteligente como la conducción semi-automática (intermitentes accionados autónomamente, frenado en curvas, etc.). En los de largo plazo pueden distinguirse los que buscan la predicción del destino final sin importar la ruta que se seguirá o los que además de detectar el lugar al que se dirige el usuario, predicen la ruta que se seguirá.

Instante de la predicción. En muchos casos, los estudios se realizan a-posteriori, para comprobar la eficiencia de los algoritmos y modelos propuestos, sin embargo existen algunos proyectos que son llevados a la práctica y realizan la predicción en tiempo real.

A continuación, describimos brevemente las aportaciones de algunos de los principales trabajos relacionados con la presente memoria, los cuales se han categorizado según los diferentes criterios en la Tabla 1.1.

- Daniel Ashbrook y Thad Starner. Georgia Institute of Technology.

  Estos investigadores describen, entre 2002 y 2003, las experiencias obtenidas de sus trabajos de campo [AS02, AS03]. Aunque gran parte de su trabajo se centra en la detección de lugares frecuentes, también proponen un modelo predictivo entrenado para encontrar el próximo destino más probable basándose en los últimos lugares visitados.

- Natalia Marmasse y Christopher Schmandt. Instituto de Tecnología de Massachusetts (MIT).

Tabla 1.1: Clasificación de diferentes sistemas de predicción.

Año-Autor-Trabajo	Ámbito	Supervisión	Objetivo	Instante
2002 Ashbrook [AS02]	Global	Supervisado	Destinos	Diferido
2002 Marmasse [MS02, Mar04]	Global	Supervisado	Rutas	T. Real
2004 Liao [LFK04, PLG$^+$04]	Local	Supervisado	Rutas	T. Real
2005 Saaman [SK05]	Local	No supervisado	Rutas	Diferido
2006 Simmons [SBZS06]	Local	Supervisado	Próxima calle	Diferido
2006 Krumm [KH06]	Local	No supervisado	Destinos	Diferido
2008 Krumm [Kru08]	Local	Supervisado	Próxima calle	Diferido
2008 Froehlich [FK08]	Global	Supervisado	Rutas	Diferido

Entre los años 2000 y 2004, desarrollan y perfeccionan comMotion [MS00, MS02, Mar04], un entorno de computación para el conocimiento de la localización que enlaza información personal con los lugares más frecuentes de la vida de los usuarios. No es hasta 2002 cuando introduce la idea de predecir rutas y destinos y en la tesis de Natalia en 2004 comenta las técnicas usadas. La aportación más importante son el dispositivo hardware diseñado, integrado en un reloj y la aplicación que permite recordar tareas relacionadas con los lugares donde deben producirse.

- Donald J. Patterson y Lin Liao. Universidad de Washington.

  Patterson implementa un sistema de ayuda a personas con problemas mentales en "Opportunity Knocks" [PLG$^+$04] utilizando la idea de desvíos de rutas frecuentes para orientar a los usuarios. En ese mismo año presentan un nuevo modelo probabilístico [LFK04] que mejora los resultados del anterior.

- Nancy Samaan y Ahmed Karmouch. Universidad de Ottawa.

  Samaan propone un modelo teórico [SK05] que combina información de mapas, perfiles de usuario y preferencias del mismo. Utiliza el razonamiento mediante evidencias utilizando la teoría de Dempster-Shafer [Dem68, Sen02]. Al contrario que en los anteriores trabajos, donde el ámbito de las predicciones eran

ciudades, centra sus estudios en un campus universitario.

- Reid Simmons y otros autores. Universidad Carnegie Mellon.

  En el año 2006 lleva a cabo una aproximación interesante en su trabajo [SBZS06], donde utiliza una porción de un mapa electrónico de las carreteras de la ciudad de Detroit. En él predice la ruta deseada por el conductor y el destino del usuario usando un modelo probabilístico aprendido de la observación de sus hábitos de conducción.

- John Krumm y Eric Horvitz. Microsoft Research.

  En 2006 [Kru06, KH06] introduce predicciones sobre lugares que no han sido visitados anteriormente basándose en el tipo de suelo de la ciudad de Seattle y en la idea de que los conductores intentan siempre utilizar rutas eficientes hacia los destinos a los que se dirigen. En 2008 [Kru08] desarrolla un modelo basado en el mapa de la ciudad de Seattle para predecir la próxima calle que tomará un determinado conductor.

- Jon Froehlich. Universidad de Washington.

  En 2008 [FK08] realiza un estudio de cómo predecir las rutas de los conductores basándose en similitudes.

1.4. Justificación

Es claro que la información que nos brindan hoy en día los sistemas de posicionamiento, tanto en exteriores como en interiores, es muy valiosa, sin embargo tras realizar un estudio en profundidad de las técnicas y métodos actuales utilizados en los trabajos relacionados con el que nosotros proponemos, encontramos una serie de carencias o debilidades que pasamos a describir y que pretendemos aliviar o subsanar:

Tratamiento de la información *off-line*. En la mayoría de trabajos, se realizan

estudios teóricos a partir de información de trazas GPS almacenadas por diferentes usuarios a lo largo del tiempo. Esto hace que sea complicado evaluar su utilidad en dispositivos que procesen los datos en tiempo real y que permitan una computación ubicua. En algunos casos el post-procesamiento de la señal y el modelo resultan demasiado complejos para implementarlos en dispositivos de capacidades de cálculo reducidas.

Información sobre recorridos realizados en automóviles. Los proyectos de predicción de destinos suelen considerar únicamente desplazamientos en vehículo ya que los dispositivos de localización y de análisis de la información se conectan a la batería del mismo. Sin embargo, si se pretende realizar un sistema disponible de manera continua, es necesario monitorizar los movimientos de las personas y no de sus vehículos, considerando aspectos como la invasión de su privacidad y la gestión de las baterías de los dispositivos utilizados.

Dependencia de la interacción del usuario. En el proceso de recogida de información espacial, la activación y desactivación del sistema y/o de los dispositivos de posicionamiento deben realizarse por el usuario. Este aspecto hace normalmente más engorroso, menos usable y más propenso a perder datos por no activar o desactivar interactivamente el sistema.

Falta de privacidad. En sistemas que se utilizan bases de datos externas o sistemas de información espacial, en definitiva cuando se utiliza un esquema cliente-servidor, se envía frecuentemente datos de localización personales por un medio poco seguro. Esto lleva asociado problemas de privacidad.

Dependencia de bases de datos externas. Como hemos comentado, la dependencia de GIS tiene dos consecuencias directas. La primera de ellas es que exista un envío de información a través de medios inalámbricos poco seguros (en caso de que el GIS no esté empotrado en el dispositivo móvil). La segunda posible consecuencia es que el dispositivo integre un GIS con lo que el coste en espacio y computación se ve aumentado en varios órdenes de magnitud.

Modelos cerrados. Entendemos por modelo cerrado aquél que no predice lugares

no visitados anteriormente. La gran mayoría de estudios se basa en bases de datos iniciales con muchos datos. En el Capítulo 7 propondremos una interesante herramienta que nos posibilitará ir aumentando los destinos y rutas sin haberlos realizado previamente.

Funcionalidad muy orientada. La mayoría de las predicciones tienen como objeto la ayuda a conductores de vehículos en su conducción pero las posibilidades son mayores.

1.5. Objetivos

Una vez justificada la temática de la tesis, explicaremos los objetivos fundamentales de la misma.

- Diseñar un modelo que permita obtener un mapa personal asociado a cada usuario que sustituya al GIS y que no dependa del lugar del mundo en el que se encuentre.
- Definir una metodología que nos permita recuperar datos geo-posicionados para hacer un tratamiento de los mismos y elegir las mejores técnicas y algoritmos para realizar una detección eficiente de destinos y rutas.
- Definir varios modelos que nos permitan realizar la predicción de destinos on-line.
- Definir similitudes entre recorridos finalizados para poder generar un cluster de estos y detectar rutas comunes.
- Aplicar la metodología y los resultados obtenidos en la fase *off-line* a un entorno *on-line*.
- Definir nuevas técnicas que permitan aumentar el tiempo de vida de las baterías del dispositivo pudiendo éste ejecutar de manera continua un servicio de predicción de destinos.

- Proponer nuevas aplicaciones que permitan demostrar la utilidad de las predicciones en tiempo real.

1.6. Hipótesis de partida

El desarrollo de este trabajo de investigación se orientará hacia un sector cada vez más amplio de usuarios que dispongan y utilicen frecuentemente dispositivos móviles con capacidades de gestión de base de datos y localización por GPS. Cuanta más capacidad tenga el dispositivo más posibles aplicaciones con conocimiento del contexto futuro podrán implantarse.

Aunque las propuestas de esta tesis se pueden aplicar a sistemas de localización diferentes, en la implementación nos centramos en la predicción de recorridos urbanos e inter-urbanos rastreados por receptores GPS. Éstos no permiten el seguimiento en espacios sin visión directa sobre los satélites, por lo que quedan fuera de este trabajo la predicción de destinos en interiores.

Nuestra propuesta, al usar como soporte mapas personales creados por el propio usuario al desplazarse diariamente, no necesitará de ningún GIS por lo que será utilizable en lugares donde no exista cartografía detallada, como por ejemplo en mares y océanos.

El público objetivo de las predicciones de destino serán personas con todo tipo de hábitos de desplazamientos y que utilicen cualquier medio de transporte en el que puedan utilizarse dispositivos electrónicos (quedan por tanto fuera de nuestra hipótesis los aviones).

1.7. Estructura del documento

El resto de este trabajo está estructurado en los siguientes capítulos:

Capítulo 2: Metodología off-line. Se propondrá una metodología para la recuperación de la información, su preprocesado y los algoritmos de extracción de conocimiento a partir de trazas de movimiento aportadas por diferentes voluntarios. Los objetivos de este capítulo son por una parte estudiar los métodos existentes para extraer los destinos frecuentes y las rutas seguidas, aportar nuevos algoritmos que mejoren los resultados, proponer una metodología con fases bien diferenciadas y estudiar los aspectos prácticos para poder desarrollar una metodología on-line realista.

Capítulo 3: Modelo de Markov Oculto sobre "Mapa Soporte". Basándonos en las rutas que genera un usuario, creamos un mapa personal que nos permite utilizarlo aplicando un Modelo Oculto de Markov (HMM). El mapa personal disminuye en gran medida la cantidad de información utilizada por lo que lo estudiamos para aplicarlo en dispositivos móviles.

Capítulo 4: Similitud de recorridos. Las medidas de similitud existentes entre recorridos no se adaptan a nuestras necesidades por lo que propondremos nuevas métricas que permiten comparar tanto recorridos finalizados entre sí como recorridos no finalizados con aquellos que sí lo han hecho, lo que nos permite predecir rutas y destinos.

Capítulo 5: Resultados. Revisaremos los resultados obtenidos aplicando la metodología off-line y evaluaremos los modelos de predicción utilizando rutas mediante similitudes y "Mapas soporte" a través de HMM, comparando las dos alternativas.

Capítulo 6: Predicción on-line. Una vez comprobados los resultados de los diferentes modelos de predicción, trataremos su implementación en un dispositivo móvil para permitir la predicción en tiempo real. Además se propondrán técnicas para reducir el consumo innecesario de baterías, mejorando la disponibilidad de los servicios de predicción.

Capítulo 7: Aplicaciones. Mostraremos las aplicaciones definidas para nuestro sistema de predicción on-line, de modo que se destaque la utilidad del sistema de predicción on-line propuesto.

Capítulo 8: Conclusiones y trabajo futuro. En este capítulo se presentan las conclusiones de esta tesis doctoral y las líneas de investigación abiertas sobre las que se trabajaremos en los próximos años.

Apéndice A: Sistema GPS Describiremos de forma breve en este apéndice el funcionamiento del *Global Positioning System* y la estructura de las sentencias NMEA.

Apéndice B: Currículum. En este apéndice incluimos las publicaciones y los proyectos en los que hemos participado durante la elaboración de este trabajo.

Bibliografía. Tras los dos apéndices, incluimos la bibliografía utilizada en la elaboración del documento.

CAPÍTULO 2

METODOLOGÍA OFF-LINE

En este capítulo se describen los procesos necesarios para transformar los datos obtenidos mediante receptores GPS en recorridos y rutas. De ese modo es posible preparar una predicción de destinos basada en recorridos pasados.

Para ello explicaremos en primer lugar el estado del arte para a continuación detallar nuestra metodología y todas sus fases.

2.1. Estado del arte

Para obtener predicciones de destinos basadas en el historial de desplazamientos de un usuario es necesario en primer lugar, tratar los datos mediante una serie de procesos que permita generar un conjunto de destinos frecuentes, un conjunto de recorridos y si se quiere realizar una predicción de rutas, un conjunto de rutas.

La terminología utilizada por los diferentes autores asume las siguientes definiciones:

Destino frecuente. Lugar que puede o no ser techado donde un usuario realiza una

estancia durante un determinado tiempo que le resulta significativa. Además las visitas a ese lugar las hace con cierta asiduidad.

Recorrido. Secuencia de puntos temporalmente ordenados, generados durante el desplazamiento de un usuario. Éste comienza con la salida del individuo de un destino frecuente y finaliza con la llegada a otro. Los recorridos podrán realizarse a pie o en cualquier medio de transporte, excepto en avión por no poder utilizar dispositivos electrónicos. Los puntos de los que se compone el recorrido tienen información diversa como la longitud, latitud, altura, velocidad, orientación e instante temporal.

Ruta. Secuencia de puntos temporalmente ordenados, que representa a uno o más recorridos. Los puntos de los que se compone la ruta tienen información de longitud, latitud, altura, velocidad y orientación. Aunque se podría almacenar información sobre el día de la semana o el periodo del día en el que se realizan esos recorridos, normalmente se obvia la información temporal.

Para conseguir obtener estos conjuntos, la metodología seguida por la mayoría de los autores se puede describir como la cadena secuencial de los siguientes procesos:

1. **Recuperación de la información de localización.** El objetivo de este proceso es el de obtener un conjunto de puntos válidos de localización con una marca temporal durante un periodo de tiempo suficiente como para poder realizar predicciones (en torno a un mes). Destaca de esta fase el protocolo que debe seguir el usuario que interviene en la recuperación para gestionar los dispositivos que debe llevar en todo momento y el filtrado de outliers.

2. **Segmentado en recorridos.** Una vez recuperados todos los puntos geolocalizados y marcados temporalmente es necesario realizar un segmentado que indique cuando un punto pertenece a un recorrido o al siguiente.

3. **Filtrado de recorridos.** Del conjunto obtenido, se deben eliminar aquellos recorridos espurios. Éstos pueden venir del comportamiento incorrecto del

usuario al seguir el protocolo de gestión del dispositivo de seguimiento, de errores en el sistema de posicionamiento o de problemas del dispositivo.

4. **Extracción de destinos.** Para recuperar los destinos frecuentes, se analiza la información de los desplazamientos recuperando los lugares visitados asiduamente.

5. **Generación de rutas.** En los sistemas de predicción que utilizan información sobre los caminos seguidos, se agrupan los recorridos en las rutas que los representan.

6. **Predicción.** El proceso de predicción permite detectar el destino final al que un usuario se dirige. Este proceso se surte del los conjuntos obtenidos (destinos frecuentes, recorridos y rutas).

A continuación mostraremos como se tratan la secuencia de procesos en otros estudios.

2.1.1. Recuperación de la información de localización

En todos los trabajos en los que se propone un sistema de predicción basado en recorridos pasados es necesario aplicar técnicas para recuperar los datos provenientes del sistema de localización. Aunque estos métodos dependen del sistema utilizado (GPS, Puntos de acceso WiFi geo-localizados, antenas GSM, etc.) siempre es necesario que los usuarios estudiados porten un dispositivo capaz de almacenar con una determinada frecuencia los valores de posicionamiento. Lo deseable sería que esto fuera menos invasivo, por ejemplo una aplicación que se ejecute de manera transparente en el teléfono móvil, sin embargo hasta estos últimos años, los dispositivos que integran un sistema de localización global no han aparecido en el mercado con precios razonables. Para conseguir una información útil, es necesario definir un protocolo de uso sencillo del dispositivo y motivar a los usuarios para que lo sigan de manera continuada durante todo el periodo de recuperación de datos.

En muchos casos la motivación era la investigación dado que eran los propios autores quienes portaban los receptores [AS03, Mar04, SBZS06] por lo que un protocolo de uso sencillo no era una prioridad. Sin embargo en otros casos [LFK04, PLG+04, KH05, RMI07] los usuarios eran voluntarios que no tenían relación con los autores por lo que sí que era una prioridad. Cabe destacar la experiencia de Microsoft [KH05] donde fomentaron el uso de receptores GPS con el sorteo de un reproductor de audio y vídeo, de ese modo consiguieron una alta participación de empleados de la compañía. Además conectaron la alimentación del receptor GPS a las baterías de los vehículos de los voluntarios de modo que con el arranque y paro de los vehículos éstos se activaban o paraban (aunque obviamente sólo conseguían desplazamientos en un único medio de transporte).

Una vez definido el protocolo y motivados los usuarios, es vital filtrar aquellos valores de localización espurios o redundantes. En el caso concreto del sistema GPS, al existir un gran número de receptores en el mercado, éstos implementan diferentes políticas de almacenamiento de los puntos, por lo que no es necesario realizar el filtrado. Los filtrados más comunes son los siguientes:

Por intervalo de tiempo. La captura de puntos se hace con una frecuencia temporal determinada, evitando así redundancia por puntos excesivamente cercanos en tiempo.

Por intervalo de espacio. Se fija la frecuencia espacial, de modo que hasta que el usuario no recorra un determinado espacio no se almacenará el siguiente punto.

Por cambio de dirección. No hay intervalos temporales ni espaciales, sólo se almacenan aquellos puntos que cambien la dirección en un número de grados con respecto a la original. Es muy útil para evitar almacenar muchos puntos en trayectos con largas rectas.

Por cambio de velocidad. Tampoco hay intervalos temporales o espaciales, sólo se guardan los valores en los que varíe en varias unidades la velocidad del recep-

tor. Se considera interesante cuando queremos determinar cuando se comienza un desplazamiento y cuando se detiene.

2.1.2. Segmentado en recorridos

Como los puntos GPS no ofrecen una indicación explícita de cuando comienza o finaliza un recorrido, es necesario segmentar la información para crearlos. Marmasse y Schmandt [MS00] usaron la pérdida de la señal GPS para indicar que el usuario había entrado en un edificio y por tanto se podía considerar un lugar de visita. Ashbrook y Starner [AS03] ampliaron esa técnica marcando como puntos de segmentado aquellos en los que un usuario permanecía más de 10 minutos, es decir, el punto de la secuencia que distaba temporalmente más de 10 minutos con el siguiente. Llamaremos a esta técnica a partir de ahora segmentado por salto temporal y podemos ver un ejemplo de ésta en la Tabla 2.1.

Tabla 2.1: Segmentado por salto temporal entre los puntos 659 y 660.

Id_Punto	Tiempo	Latitud	Longitud	Recorrido
658	13/12/2007 20:47:41	37.38184	-5.96976	1
659	13/12/2007 20:47:49	37.38176	-5.96977	1
660	13/12/2007 22:13:52	37.36250	-5.98221	2
661	13/12/2007 22:13:58	37.36250	-5.98220	2

El segmentado por salto temporal es la política seguida por diferentes autores [LFK04, HT04, KH06, Kru08, SBZS06, PLFK03] variando el tiempo de estancia entre tres y diez minutos. La elección de este valor de tiempo es una tarea compleja dado que es difícil diferenciar entre un semáforo que retenga el vehículo tres minutos y una parada del mismo tiempo para recoger a alguien en un lugar significativo. En los trabajos de Liao [LFK05, LFK07] se estudian los errores de esta técnica y se detecta que con valores entre cinco y diez minutos se obtienen el menor número de falsas detecciones. Concretamente el número de falsos positivos es inversamente

proporcional al incremento de tiempo y el número de falsos negativos lo es directamente. A continuación describimos el porqué de las apariciones de estos errores en el segmentado:

Falsos positivos. Se trata de puntos considerados erróneamente como fin de un recorrido. Pueden darse con semáforos en rojo de muy larga duración o atascos.

Falsos negativos. Son puntos no considerados como fin de recorridos pero que realmente sí lo son. Se suelen producir cuando conducimos un coche y llevamos a otra persona hasta un determinado lugar para continuar hasta nuestro destino. Lo mismo ocurre si en vez de llevar a alguien lo recogemos.

2.1.3. Filtrado de recorridos

Aunque no se trata en profundidad en ningún trabajo, el proceso de eliminación de recorridos espurios es fundamental para no incorporar desplazamientos erróneos en el sistema de predicción. Froehlich [FK08] propone una serie de algoritmos de filtrado para su conjunto de recorridos. Destacan los filtrados por tiempo mínimo (30 segundos), por número de puntos mínimo (10 puntos), por distancia mínima (160 metros), por cambio de dirección frecuente y por zona (Washington). Estos filtrados le permitieron reducir su conjunto de recorridos en un 47.2 %.

2.1.4. Extracción de destinos

Una vez obtenidos los recorridos es necesario recuperar los destinos de cada usuario. Para ello existen dos aproximaciones:

Basada en trazas. Utiliza toda la secuencia de puntos obtenidos por el sistema de localización para detectarlos. Destacan las propuestas de Kang [KWSB04, KWSB05] y la de Zhou [ZFL+04]. La primera, utiliza el sistema PlaceLab [LCC+05] y plantea un cluster de agregación con umbrales de tiempo y dis-

tancia. En la segunda se trabaja con un algoritmo de clustering basado en densidad de puntos.

Basada en puntos finales. En vez de utilizar todos los puntos, se trabaja directamente sobre el conjunto de puntos finales de recorridos mediante segmentado temporal. En esta aproximación encontramos los trabajos de Ashbrook [AS03] que utiliza múltiples ejecuciones del algoritmo K-Means para detectar el número de clusters necesarios, de Liao [LFK04, LFK05] que extrae destinos agrupando lugares cercanos localizados en un grafo generado previamente y la propuesta de Hariharan y Toyama [HT04] en la que utilizan, al igual que Kang, umbrales para el tamaño del destino y la duración de la estancia.

Un aspecto fundamental en ambas aproximaciones es el radio de los destinos considerados. Aunque las formas de los destinos pueden variar, se asume por la mayoría de los autores que siempre pueden representarse como círculos. La elección del tamaño del radio suele obtenerse del objetivo de los trabajos y de la experiencia por lo que encontramos propuestas muy diversas: Reddy [RSB+09] utiliza 250 metros, Ashbrook [AS03] 320 metros y Hariharan [HT04] 25 kilómetros, esta última para estudios de predicciones inter-urbanas.

Otro aspecto importante es la eliminación de destinos poco frecuentes. En el trabajo de Marmasse [MS00] que sigue la aproximación por puntos finales se considera un destino frecuente como aquel que se visita tres o más veces. Sin embargo otros autores [ZFL+04, AS03] no filtran los destinos poco frecuentes.

2.1.5. Generación de rutas

En los estudios basados en rutas es primordial agrupar los recorridos similares en una única ruta, para conseguir que el sistema sea escalable. Para conseguirlo, es necesario completar los siguientes procesos:

Definir una medida de similitud. Permite comparar diferentes recorridos entre sí.

Aplicar un algoritmo de cluster. De ese modo es posible agrupar diferentes recorridos.

Determinar la ruta representante. Para tener una única representación de todos los recorridos es necesario definir una ruta ya sea ficticia o por elección de un recorrido existente.

En la Tesis de Natalia Marmasse [Mar04] se propone una medida de similitud basada en el resumen de cada recorrido. Este resumen se obtiene calculando la latitud y longitud medias de cada tramo de 100 metros. Aunque no se especifica, se entiende que la similitud depende la distancia punto a punto de cada resumen. El cluster permite agrupar todos aquellos recorridos cuyos resúmenes sean suficientemente parecidos (difieren en menos de la desviación típica). Por último la ruta representante viene dada por la combinación de los puntos representantes de cada recorrido.

Esta idea también es explotada por Froehlich [FK08] donde propone una medida de similitud que estudiaremos en la Sección 4.2 junto con un cluster jerárquico y una política de elección de ruta canónica por combinación de recorridos tal y como hace Marmasse.

2.2. Nuestra propuesta

Basándonos en los trabajos del estado del arte, definimos una metodología que consigue mejorar la calidad de los datos generando estructuras útiles para realizar una clasificación supervisada temprana y que se consigue mediante las técnicas que veremos en los siguientes capítulos.

2.2.1. Esquema

Los procesos que integran esta metodología puede verse en la Figura 2.1. Aunque aplicamos la mayoría de los procesos comentados previamente, los modificamos e in-

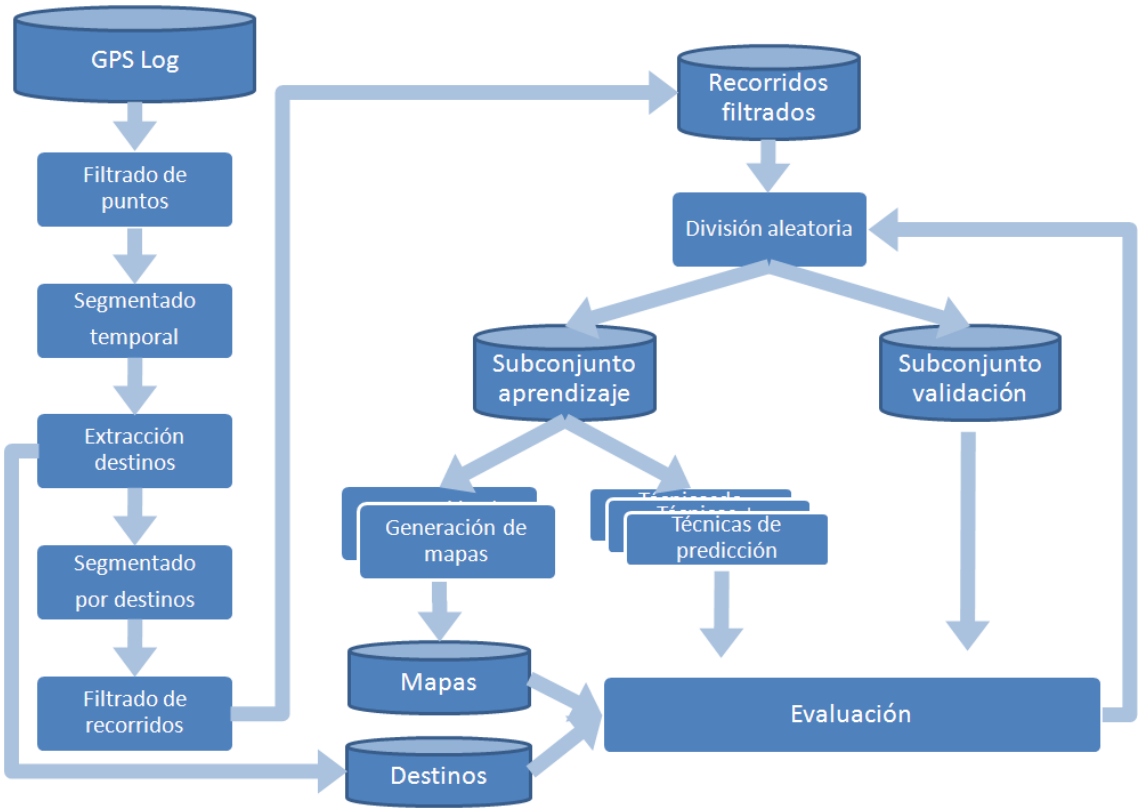

Figura 2.1: Metodología propuesta

cluimos nuevas fases. La secuencia que seguimos en esta metodología es la siguiente:

Tras recuperar los datos geo-posicionados, realizamos un filtrado de los puntos GPS y efectuamos el segmentado por salto temporal explicado en la sección anterior. Posteriormente, realizamos la extracción de destinos aplicando un cluster jerárquico con unas características particulares. Una vez obtenido el conjunto de destinos, los utilizamos para volver a segmentar los recorridos mediante una técnica simple pero novedosa. A continuación aplicamos un conjunto de filtros para eliminar recorridos inválidos o inútiles para nuestra predicción. Gracias a ello recuperamos el segundo conjunto fundamental para las predicciones: el conjunto de recorridos filtrados. A partir de entonces, aplicamos una validación cruzada por capas que nos permite evaluar diferentes sistemas de predicción. Para ello, dividimos en dos el conjunto de recorridos filtrados, uno para aprendizaje y otro de test o validación. Con el

de aprendizaje, generamos el conjunto de rutas que los representan y aplicamos la técnica de predicción seleccionada. Gracias a este esquema es posible evaluar la precisión de la predicción utilizada. Además al desacoplar los procesos, es posible modificarlos sin afectar al conjunto de la metodología.

A continuación, explicaremos detalladamente las fases de esta metodología.

2.2.2. Recuperación de puntos GPS

El objetivo de este proceso es el de obtener un conjunto de puntos geo-localizados y marcados temporalmente para cada usuario. Para ello fue necesario almacenar los datos recuperados tras interpretar las sentencias NMEA de los receptores GPS.

Definición 1 (*LogGPS*) *Para cada usuario, definimos el conjunto de todos los puntos obtenidos durante el periodo de experimentación por su receptor GPS, que notaremos como LogGPS, como:*

$LogGPS = \{a_1, \ldots, a_n\}$, *siendo*

$a_i =< lat_i, lon_i, alt_i, vel_i, rumbo_i, fecha_i, hora_i, HDOP_i, VDOP_i, PDOP_i >$, *siendo* $HDOP_i, VDOP_i, PDOP_i$ *medidas de precisión del dato que se explican en el Apéndice A.*

Para conseguirlo, desde finales de 2005 hasta 2008, se estuvieron utilizando diferentes receptores GPS para recopilar información geoespacial. Este proceso se hizo en dos fases, la primera desde finales de 2005 hasta 2006 y la segunda desde 2007 hasta 2008.

En la primera, para conseguir obtener el conjunto de itinerarios, utilizamos un receptor GPS Royaltek Bluetooth. La imposibilidad de almacenar ficheros de traza en dicho dispositivo, nos llevó a utilizar una PDA Dell modelo Axim X30. La información que recibía el receptor, se enviaba a través de una conexión Bluetooth a la PDA donde se almacenaba en ficheros de texto conteniendo los datos en formato

NMEA. Para registrar los itinerarios, el usuario encendía ambos aparatos (el receptor GPS y la PDA) y los configuraba antes de salir a realizar un desplazamiento. Una vez alcanzaba el destino, guardaba la traza en un fichero en formato NMEA y volvía a apagarlos. Esta primera recogida de datos la realizó un único sujeto dadas las necesidades de hardware y la complejidad y molestias ocasionadas en el protocolo de obtención de los datos ya que todo era manual.

En la segunda fase, para aumentar el número de usuarios que recogían datos y simplificar el proceso, se utilizaron cuatro receptores Wintec-BT100[(1)], dispositivos de reducidas dimensiones (6cm x 3.8cm x 1.6cm), que permitían un mayor tiempo de recepción de datos geoespaciales antes de tener que recargar sus baterías (9 horas) y que además funcionaba como "logger" guardando toda la secuencia de puntos que se obtenían (hasta 12.680 puntos), en el propio dispositivo. En la Figura 2.2 se pueden ver los dispositivos utilizados en ambas fases, la PDA conectada al receptor RoyalTek y el receptor Wintec-BT100.

En este caso el formato de almacenamiento era propio de Wintec, pero podía transformarse a NMEA utilizando un software también propietario. Los nuevos dispositivos facilitaban la labor de la recogida de puntos geo-posicionados, sin embargo, la posibilidad de llevarlo durante más tiempo sin necesidad de apagarlo y encenderlo cada vez que se estaba en exteriores, hacía que varios recorridos fueran recogidos en un mismo archivo, provocando a su vez la necesidad de introducir un proceso de segmentado para poder recoger los recorridos individuales.

Aunque el protocolo se simplificaba, (sólo había que estar pendiente de que un sólo dispositivo estuviese cargado y no dos como antes, además de no tener que establecer una conexión bluetooth) el problema fundamental era la motivación de los usuarios a seguirlo durante todos sus desplazamientos en exteriores y descargar los datos a un ordenador. Para ello se indicó el software incorporado para el receptor que permitía mostrar visualmente los datos en diferentes formatos como Google Earth[(2)]

[(1)] http://www.wintec.tw
[(2)] http://earth.google.com

Figura 2.2: PDA y receptores GPS utilizados en la fase de recuperación de recorridos.

o Visual Earth[3]. Esto permitió que los voluntarios recuperasen y analizasen sus desplazamientos pasados, midiendo distancias y tiempos, lo que alentó su uso incluso en periodos vacacionales.

[3]http://maps.live.com

2.2.3. Filtrado y normalizado de puntos

Una vez obtenidos los puntos GPS, es necesario realizar un tratamiento que permita eliminar los outliers y normalizar la secuencia de puntos. Aunque comprobamos que otros autores se apoyaban en las políticas de almacenamiento de los receptores GPS para evitar filtrar la información, nosotros al utilizar varios receptores decidimos independizarnos de esas implementaciones propietarias, capturar la información sin restricciones por parte del receptor y posteriormente crear un conjunto de filtros para generar el conjunto de puntos filtrados.

Para comprender los filtros propuestos, comentamos a continuación los problemas derivados del sistema GPS, de los dispositivos receptores utilizados y del comportamiento de los usuarios.

2.2.3.1. Problemas conocidos en la captura de datos

Tiempos de arranque del receptor. El tiempo que tarda un receptor GPS en obtener una primera posición válida varía según diferentes situaciones. Cuando el dispositivo se utiliza por primera vez o lleva mucho tiempo sin utilizarse, se produce el "arranque en frío": la información que dispone sobre la visibilidad de los satélites es muy imprecisa por lo que tiene que buscarlos, el tiempo que puede durar ese proceso puede ser del orden de varios minutos. En nuestro caso los receptores se utilizaron frecuentemente por lo que se daba el "arranque normal" que en nuestros receptores era del orden de 40 segundos en condiciones ideales. Sin embargo, si la visibilidad sobre el cielo no era suficiente o el receptor estaba en movimiento, el tiempo de primer fijado de señal válida aumentaba enormemente, por lo que nos encontramos con situaciones en las que el primer punto del recorrido grabado estaba muy lejano del primero del recorrido real.

Cañón urbano. Cuando estamos situados en calles circundadas por edificios altos se produce el llamado efecto "cañón urbano" que consiste en el "rebote" de las señales provenientes de los satélites en los edificios. Esto causa una reducción

importante de la precisión e incluso una interrupciones de posiciones válidas de duración variable. En las ciudades en las que hemos recogido itinerarios, este efecto no suele darse por rascacielos pero sí que se da en calles estrechas típicas de Andalucía.

Puntos obtenidos en interiores. Aunque hemos comentado que el receptor GPS debe estar en espacios abiertos con visión directa sobre el cielo para obtener señales de cuatro satélites, a veces también es posible obtener datos válidos en espacios techados si el receptor GPS está cerca de una ventana, provocando capturas de puntos validados por el sistema pero aislados temporalmente.

Errores de dispositivo. Por algún tipo de error del dispositivo, determinados puntos, normalmente el primero de un nuevo recorrido distaba cientos de kilómetros con los siguientes. También aparecían puntos aislados en los polos, que debían filtrarse.

Conducción. Además de los problemas derivados de las limitaciones de la tecnología GPS, también aparecían otros referentes a la experiencia real. Al proceder a analizar los datos, encontramos situaciones que no aportaban información relevante con respecto a la forma de los recorridos. Entre ellos los atascos en la que existían numerosos puntos muy cercanos unos de otros y la búsqueda de aparcamiento que además de la normal "desesperación" del conductor, incrementaba el número de puntos en los alrededores del lugar de destino, desvirtuando la forma de la secuencia de puntos.

Medio de transporte y velocidad del trayecto. Si dos personas realizan un recorrido por la misma ruta, con sus receptores obteniendo puntos a idéntica frecuencia pero el primero lo hace a una velocidad media mucho mayor que la del segundo, al comparar las trazas que generan ambos, el resultado podría asemejarse al observado en la Figura 2.3.

Todas estas situaciones, hacían que fuese necesario realizar en primer lugar un filtrado de puntos erróneos y de los derivados de situaciones anómalas y en segundo un

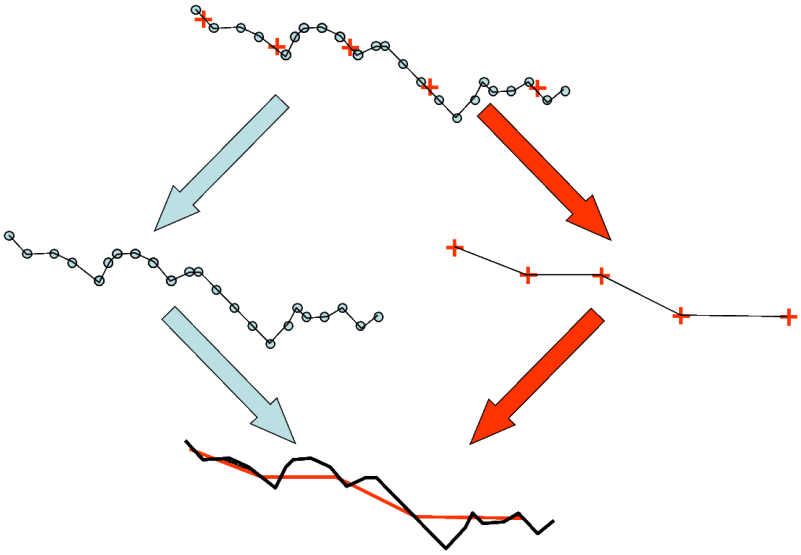

Figura 2.3: Ejemplo de recorridos idénticos con trazas diferentes.

normalizado para evitar que recorridos idénticos con velocidades diferentes tuviesen trazas GPS distintas. De ese modo tendremos puntos validados cuasi-equidistantes de manera que se pudiese comparar la componente espacial de manera independiente a la de velocidad, frecuencia de muestreo o medio de transporte utilizado.

2.2.3.2. Filtrado de puntos GPS

Definición 2 (*Filtrados*) *Definiremos el conjunto de puntos filtrados como el generado a partir de aplicar una serie de filtros al conjunto LogGPS (Definición 1) y lo notaremos como* $Filtrados = \{b_1, b_2, \ldots, b_m\}, (m \leq n)$ *siendo* $b_i = a_k$ *tal que*

$$\begin{cases} HDOP_k \leq HDOP_{max} & (Precisión) \\ vel_{min} \leq vel_k \leq vel_{max} & (Velocidad) \\ dist(a_{k-1}, a_k) \geq \delta \ o \ dist(a_{k-1}, a_{k+1}) > \delta & (Distancia) \end{cases},$$

con $HDOP_{max}, vel_{min}, vel_{max}$ *y* δ *parámetros que explicamos a continuación.*

Este conjunto, contiene una única secuencia de puntos GPS que indica todos los desplazamientos de un usuario sin indicar las paradas realizadas. Describimos a

continuación los algoritmos utilizados.

Filtrado por análisis de precisión. Como las componentes fundamentales en los recorridos son la latitud y la longitud, eliminamos aquellos puntos cuya HDOP sea muy elevada. Para calcular el valor de HDOP adecuado es necesario saber la precisión de los dispositivos receptores usados. Nuestros dispositivos poseían una precisión de 3 metros (CEP[4]) y 6 metros (2D RMS[5]). Es esta última medida la que utilizaremos. Para saber los metros de error máximo es necesario multiplicar la precisión 2D RMS del dispositivo por el valor de HDOP. Nosotros elegimos un valor de $HDOP_{max} = 6$ con objeto de tener valores dentro de los 36 metros de precisión (la mayoría de los puntos tienen una HDOP menor pero los peores casos los consideramos dentro de esa distancia). Es decir, si $HDOP_i > HDOP_{max} \Rightarrow a_k \notin Filtrados$.

Con este filtrado se elimina la información imprecisa especificada por el propio sistema GPS.

Filtrado por velocidad. Los puntos que indicaban una velocidad menor o igual al parámetro vel_{min} fueron eliminados debido a la ausencia de desplazamiento significativo. Así mismo los que indicaban una velocidad por encima de un umbral vel_max también se eliminaron. En nuestro caso se fijó $vel_min = 0$ (ausencia total de movimiento) y $vel_max = 300$ debido a que por encima de esa velocidad con vehículos terrestres es probable el error del dato. De ese modo, si $vel_k \leq vel_{min}$ o $vel_k \geq vel_{max} \Rightarrow a_k \notin Filtrados$.

Filtrado por distancia. Para eliminar puntos redundantes que provenían de situaciones en las que el usuario se desplazaba lentamente, por ejemplo, cuando un conductor está inmerso en un atasco, se consideró una distancia umbral que indicaría cuando eliminar puntos muy cercanos entre sí. Es decir, si $distancia(a_{k-1}, a_k) < \delta$ y $distancia(a_{k-1}, a_{k+1}) < \delta \Rightarrow a_k \notin Filtrados$.

Con este algoritmo conseguimos eliminar aquellos puntos demasiado cercanos

[4] Radio del círculo que contiene el 50 % de las medidas GPS.
[5] Radio del círculo que contiene el 98 % de las medidas GPS.

evitando filtrar aquellos que hacen que se alejen excesivamente cada par de puntos. Para escoger la distancia umbral δ se estudió la frecuencia de muestreo máxima de los receptores GPS (1 Hertzio) y los medios de transporte en los que podrían utilizar los usuarios los receptores (coche, autobús, moto, bicicleta, barco o a pie), descartando el avión por la prohibición del uso de aparatos electrónicos. Considerando una velocidad máxima de 144 kilómetros por hora (24 por encima de lo permitido en autopistas), la distancia máxima recorrida en un segundo sería de 40 metros. Esta distancia es la recorrida en 3.6 segundos a una velocidad media de 40 km/h (velocidad adecuada conduciendo un vehículo en ciudad) o en 36 segundos a una velocidad de 4 km/h (a pie) lo que nos permitiría filtrar una gran cantidad de puntos cuando los usuarios se desplazasen lentamente. De ese modo $\delta = 40$.

Algoritmo de eliminación de bucles. Inicialmente se contempló la posibilidad de filtrar los bucles en los recorridos. Se daban en giros indirectos durante recorridos en coche y en búsquedas de aparcamiento, pero la imposibilidad de determinar la influencia del bucle en el recorrido, terminó por descartarlo.

2.2.4. Segmentado en recorridos por tiempo

Como hemos comentado en el estado del arte, es necesario tratar los datos geoposicionados como un conjunto de recorridos por lo que debemos segmentar el conjunto *Filtrados*.

En nuestra aproximación el proceso de segmentado lo realizamos en varias fases. En primer lugar utilizamos la técnica salto temporal explicada en la sección 2.2.3. Posteriormente extraeremos los destinos (ver Sección 2.2.5) y finalmente aplicaremos un nuevo segmentado que llamaremos segmentado por distancia a destinos (ver Sección 2.2.6).

En nuestro trabajo utilizamos como umbral para el salto temporal un valor de cinco minutos ($salto_{temporal} = 5$) por considerar un tiempo suficiente como para

evitar paradas por semáforos y atascos y a la vez poder detectar breves paradas de esa duración.

Definición 3 (*SegTiempo*) *Definiremos el conjunto de puntos que indican el fin de un recorrido aplicando el segmentado temporal indicado anteriormente y notaremos como* $SegTiempo = \{c_1, \ldots, c'_n\}$, *siendo* $c_i = b_k$ *tal que* $(tiempo_k - tiempo_{k+1}) > salto_{temporal}$.

De la anterior definición determinamos el conjunto de recorridos:

Definición 4 ($Recorridos^T$) *Definiremos el conjunto de recorridos segmentados temporalmente como la secuencia ordenada de todos los puntos filtrados que finalizan en uno de los puntos de SegTiempo y notaremos como* $Recorridos^T = \left\{R_1^T, \ldots, R_n^T\right\}$ *tal que* $R_i^T = \{b_j, \ldots, b_k\}$ *siendo* b_j *el punto origen y* $b_k = c_i$ *el punto destino.*

2.2.5. Extracción de destinos

En nuestro trabajo utilizamos la aproximación de puntos finales (ver Sección 2.1.4), ya que trabajar sobre el conjunto *SegTiempo* en vez del *Filtrados* reduce en gran medida el tiempo de ejecución del algoritmo de cluster, facilitando de esa forma su implementación en dispositivos móviles.

Para poder escoger una técnica determinada de clustering observamos las siguientes características de nuestro problema.

Tamaño de los lugares. Cada lugar que se visita tiene un tamaño determinado. No es lo mismo visitar un campus universitario que una pequeña tienda.

Distancia entre lugares. Para evitar superposiciones entre lugares cercanos, debemos tener en cuenta que no podemos agrupar en un mismo cluster puntos finales muy distantes.

Falsos lugares. Largas paradas debidas a atascos o esperas en lugares que no significan nada para el usuario podrían dar lugar a detección de "falsos lugares". Este aspecto es significativo porque algoritmos como K-means incluyen todos los puntos en el cluster final, haciendo los resultados bastante sensibles al ruido. En nuestra propuesta deberemos poder dejar al margen este tipo de fenómenos.

De lo anterior podemos deducir que tendremos que establecer un umbral de tamaño máximo que permita detectar lugares que ocupen grandes extensiones (en la práctica lo que tenemos son diferentes puntos de acceso lejanos a un mismo lugar) pero que no agrupe dos lugares cercanos como uno solo. Por otra parte también debemos configurar nuestro algoritmo de cluster de modo que no incluya puntos que supongan ruido.

2.2.5.1. Radio de destinos

Aunque ninguno de los autores propone justificadamente sus medidas de distancia, parece lógico que cuanto mayor sea ésta, menor será la granularidad, es decir no distinguiremos entre destinos cercanos entre sí. A su vez conseguiremos predicciones más precisas, debido a que se distinguen menos lugares y hay menos probabilidades de que la predicción sea errónea. Si considerásemos una distancia umbral de 25 kilómetros, podríamos realizar predicciones de destinos entre diferentes ciudades sin embargo no valdrían las predicciones urbanas, donde suelen realizarse la mayoría de los desplazamientos. Si por contra consideramos una distancia reducida, tendremos una alta granularidad, distinguiendo por ejemplo entre edificios cercanos pero reduciendo la precisión de la predicción. Por ejemplo, con una distancia umbral de 20 metros, podríamos distinguir entre el edificio donde trabajamos y el parking del mismo. Eso provocaría problemas a la hora de predecir el destino cuando nos dirigimos a trabajar ya que ambos lugares están muy relacionados y muchas veces se indicaría que vamos al parking en vez de al trabajo.

Al depender el radio de los destinos de la distancia mínima entre lugares que

visita cada usuario, decidimos basarnos en otro parámetro para establecer la granularidad del sistema: la utilidad de las predicciones. Las predicciones resultan más útiles cuánto más se adelanten al hecho que predicen. Consideramos útiles las predicciones que adelanten el destino al que se dirige un usuario en más de 3 minutos. Si tenemos en cuenta la menor velocidad de desplazamiento estudiada (a pie: unos 4 kilómetros por hora), la distancia que podremos recorrer en esos 3 minutos a esa velocidad es de unos 200 metros. Esa será el valor de nuestro parámetro.

2.2.5.2. Mínimo número de puntos por cluster

Como comentamos, se dan situaciones en las que se podrían detectar "falsos lugares". Un ejemplo que se repite frecuentemente son los atascos en carreteras de acceso a las grandes ciudades en horas punta. Si cada lugar en los que nos quedamos parados en esa carretera más de cinco minutos (tiempo utilizado para el segmentado temporal) lo consideramos como un lugar, obtendríamos múltiples destinos nada significativos. De ese modo, definimos un parámetro de mínimo número de puntos por cluster para que aquellos lugares que no fuesen visitados menos de un cierto número de veces no se considerasen destinos frecuentes.

El número de repeticiones se escogió variándolo en el rango desde una a cinco y comprobando el número de lugares detectados. Un número muy elevado de repeticiones provocaba una pérdida de información significativa sabiendo que se admitieron datos de usuarios que sólo recogieron dos semanas de recorridos. Por otra parte, un número reducido, detectaba lugares muy poco frecuentes por lo que se escogió un valor de tres repeticiones, coincidiendo con el utilizado en el trabajo de Marmasse [MS00].

2.2.5.3. Cluster jerárquico modificado

El cluster utilizado para agrupar los puntos pertenecientes a $SegTiempo$ en destinos es una modificación de un cluster jerárquico por distancia mínima. La modifi-

cación consistió en que, tras aplicar este cluster, eliminamos aquellas agrupaciones con menos de tres elementos.

Definición 5 (*Destinos*) *Definimos el conjunto de destinos como el obtenido tras aplicar un cluster jerárquico por distancia mínima utilizando los parámetros de radio del destino (radio) y mínimo de puntos (minPtos) al conjunto SegTiempo y lo notaremos como*

$$Destinos = CJerarquico(SegTiempo, radio) = \{d_1, d_2, \ldots, d_n\},$$

siendo $|d_i| \geq minPtos$.

Una vez obtenido *Destinos* para cada recorrido $R_i^T = \{b_i, \ldots, b_j\}$ etiquetamos su origen y destino con los destinos más cercanos a sus puntos extremos, es decir, R_i^T tendrá como origen d_k y destino d_l siendo:

$$\text{mín}(dist(b_i, Destinos)) = dist(b_i, d_k),$$
$$\text{mín}(dist(b_j, Destinos)) = dist(b_i, d_l).$$

En la Figura 2.4 podemos ver un extracto del mapa de destinos obtenidos para el usuario 1 estudiado en el Capítulo 5. En este mapa los puntos muy cercanos no pueden verse por lo que hemos agregado la lista de sus identificadores. Se puede observar cómo se han obtenido dos puntos finales (marcados con los números 7 y 26) que no han sido agrupados como ningún lugar al repetirse menos de tres veces.

Debemos remarcar que sólo los puntos finales y no los puntos orígenes del conjunto $Recorridos^T$ se usaron como candidatos para obtener los destinos frecuentemente visitados. Se hizo así para evitar el problema de tiempo de arranque del receptor explicado en la Sección 2.2.3.1. Recordamos que ese tiempo provocaba que un recorrido contuviese puntos no válidos en su inicio: $R_i^T = \{a_i, a_{i+1}, \ldots, a_j, b_k, b_{k+1}, \ldots, b_n\}$. De ese modo, al filtrar los primeros puntos, el primer punto válido b_k distaba mucho del origen real, sin embargo el último punto válido b_n sí era un candidato adecuado.

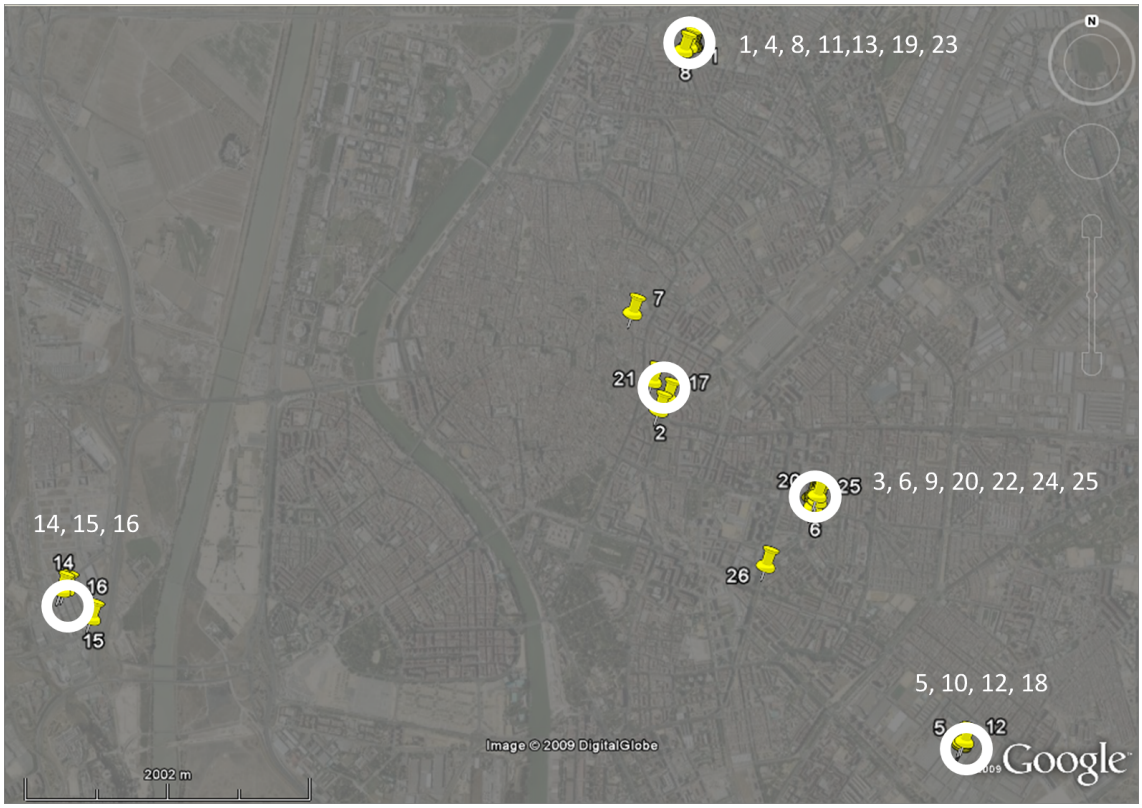

Figura 2.4: Extracto de los puntos finales de los recorridos realizados por el usuario 1.

2.2.6. Segmentado por cercanía a destinos

Tras obtener *Destinos* pasamos a definir el conjunto *Recorridos* que evita los falsos negativos existentes en $Recorridos^T$ (ver Sección 2.1.2).

Como lo que queremos detectar son portes rápidos (recoger o llevar a alguien o algo) a un destino conocido, realizamos un nuevo proceso de segmentado temporal sobre $Recorridos^T$ bajando el umbral de tiempo (el tiempo mínimo para parar un vehículo, esperar a que entre o salga alguien y se vuelva a poner en marcha) y considerando sólo los puntos que estén próximos a *Destinos*. Tras comprobar diferentes tiempos de parada y arranque, consideramos el nuevo umbral de tiempo de 40 segundos. Por otra parte, el umbral de cercanía a un destino se estableció en 90 metros debido a que no siempre es posible aparcar el vehículo exactamente en el

centro del destino.

De ese modo, se analizan de nuevo cada uno de los recorridos y si existen puntos cercanos a un lugar frecuente que no corresponde ni al origen ni al fin del recorrido y el tiempo de permanencia en esa zona es suficiente para considerarlo una parada, segmentaremos de nuevo el recorrido en ese punto.

Definición 6 (*SegDestinos*) *Dado el conjunto Recorridos*^{T}*, para cada* $R_i^T = \{b_i, b_{i+1}, \ldots, b_j, b_{j+1}, \ldots, b_k\}$ *siendo* b_i *el origen del recorrido y* b_k *el destino, definimos el punto de segmentado por cercanía a destinos* b_j *y notamos como* $SegDest_j$ *si* $distancia(b_j, Destinos') < Umbral$ *y* $(tiempo_{j+1} - tiempo_j) > 40seg$*, siendo* $Destinos' = Destinos - \{d_i, d_k\}$ *(el conjunto de todos los destinos exceptuando el destino origen y destino del recorrido i-ésimo). Definimos el conjunto* $SegDestinos = \{SegDest_1, \ldots, SegDest_n\}$ *como todos los puntos encontrados que cumplen la anterior condición.*

De ese modo se generan nuevos puntos de corte que complementan a los ya encontrados anteriormente. En la Figura 2.5 podemos ver cómo se vuelve a segmentar un recorrido tras haber realizado la detección de destinos. A la derecha se muestran los dos recorridos resultantes del recorrido de la izquierda tras aplicar el segmentado por proximidad a destinos habiendo detectado un salto temporal de 2 minutos en los alrededores del destino 4.

Mediante la unión de los conjuntos *SegTiempo* y *SegDestinos* obtendremos todos los puntos de segmentado que definen el conjunto *Recorridos*:

Definición 7 (*Recorridos*) *Sea* $SegTiempo \cup SegDestinos = \{e_1, \ldots, e_n\}$ *Definiremos el conjunto de recorridos como* $Recorridos = \{R_1, \ldots, R_n\}$ *tal que* $R_i = \{b_j, \ldots, b_k\}$ *y* $R_{i+1} = \{b_{k+1}, \ldots, b_l\}$*siendo* $b_k = e_i$ *y* $b_l = e_{i+1}$.

Al igual que hicimos con el conjunto $Recorridos^T$ etiquetamos los puntos orígenes y destinos con los destinos más cercanos.

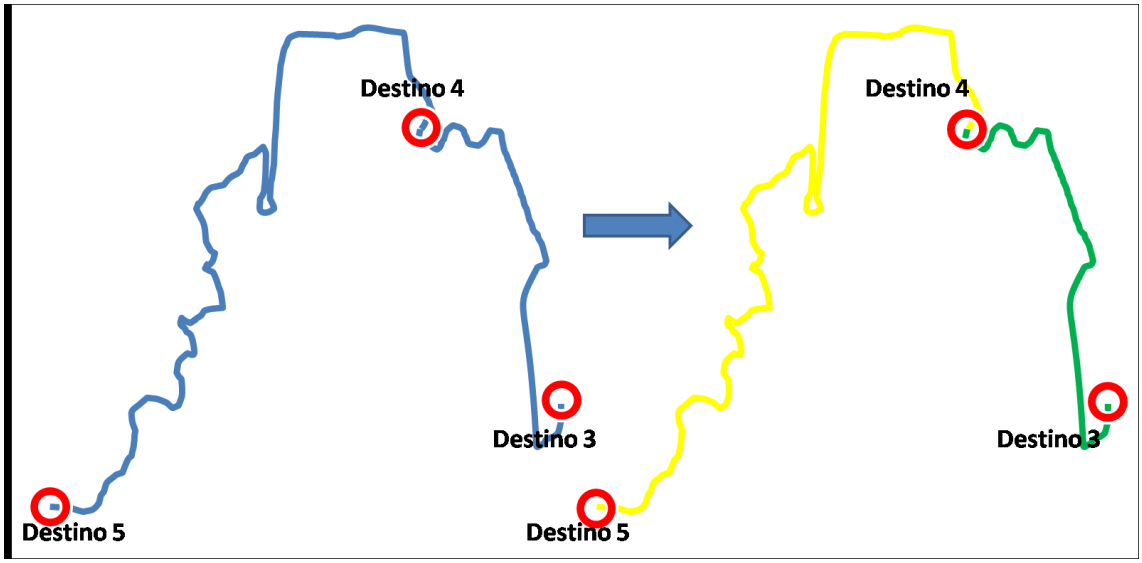

Figura 2.5: Ejemplo de segmentado por cercanía a destinos

2.2.7. Filtrado de recorridos

Una vez obtenidos el conjunto *Recorridos*, eliminamos aquellos elementos inútiles para nuestro objetivo de predicción de destinos o que incluso pueden ser ruido. Los primeros son los que finalizan muy lejos del conjunto *Destinos*. Un ejemplo de ello es aquel que finaliza en un punto final no considerado destino por repetirse menos de tres veces. Los segundos son recorridos que tienen el mismo origen y fin. En ambos casos procedemos a eliminar los recorridos de nuestro conjunto, generando uno nuevo de recorridos filtrados.

Definición 8 (*RecorridosFiltrados*) *Definimos el conjunto de recorridos filtrados como aquél cuyos recorridos cumplen las anteriores restricciones y notamos como* $RecorridosFiltrados = \{R_1, \ldots, R_n\}$ *tal que para cada recorrido* R_i, $origen_i \neq destino_i$ *y* $distancia(destino_i, Destinos) \leq radio$.

2.2.8. Validación

En nuestro análisis, el criterio usado para generar recorridos de test y de aprendizaje, fue una validación cruzada de M-capas, donde M se escogió de acuerdo al tamaño de conjunto de datos. El procedimiento se repitió 50 veces para poder asegurar un buen comportamiento estadístico y de esta forma evitar comportamientos extraños al considerar valores promedios. En este punto dividimos el conjunto $RecorridosFiltrados$ en $RecorridosAprendizaje$ y $RecorridosTest$. A partir de este momento, la obtención de mapas y las técnicas de predicción serán ejecutadas sobre $RecorridosAprendizaje$.

2.2.9. Obtención de mapas personales

Con el conjunto $Destinos$ y $RecorridosFiltrados$ podemos prever el destino que alcanzaremos cuando un usuario comienza un nuevo recorrido, pero las aplicaciones aumentan cuando somos capaces de predecir las zonas que atravesaremos para llegar al destino indicado como ayudas al conductor para activar intermitentes cuando se predigan giros en el recorrido [Kru08].

Proponemos dos técnicas para crear mapas personales, es decir, mapas generados por y para el propio usuario y por tanto muy útiles para el problema de la predicción de destinos:

Rutas Como hemos comentado, las rutas son representaciones atemporales de los recorridos. Si tenemos el conjunto de rutas que representa a cada uno de los recorridos, tendremos un mapa personal más ligero que el correspondiente a todos los recorridos y además almacenaremos toda la información geográfica de las zonas que recorre el usuario frecuentemente.

Puntos soporte Tras generar las rutas, consideramos que la importancia de éstas venía dado por los puntos posteriores a un cruce entre ellas, del mismo modo que en un cruce de calles nos interesa saber cuál es la calle que escoge el usua-

rio para poder predecir con mayor probabilidad el destino final. Este mapa corresponderá únicamente con puntos estratégicos extraídos de cruces y bifurcaciones de rutas, siendo más ligero que el anterior. Al estar tan relacionado con el modelo de predicción que lo usará (HMM), será explicado en la Sección 3.3.1.

2.2.10. Obtención de rutas

Como vimos en la sección 2.1.5, para poder realizar un agrupamiento de recorridos en rutas, lo primero que necesitamos es una medida de similitud entre éstos. Esta medida se estudiará en el Capítulo 4.

Una vez que podemos comparar recorridos, los agrupamos mediante un cluster jerárquico por distancia mínima y utilizamos un umbral de disimilitud para definir el corte de las agrupaciones. Esta técnica también es usada en el trabajo de Froehlich [FK08].

Tras obtener el conjunto de recorridos que forman un cluster es necesario conocer cómo será la ruta que lo represente. En caso de existir un único recorrido en un determinado cluster, la ruta es directa (simplemente eliminamos la componente temporal). Sin embargo cuando hay más de un recorrido, caben diversas posibilidades para que se mantenga la información fundamental. A continuación comentamos las diferentes opciones:

Fusión de recorridos. Consiste en la inclusión de todos los puntos de los recorridos que pertenecen a un cluster en la ruta. El problema es el aumento proporcional del número de puntos que contiene esa ruta al número de recorridos del cluster.

Fusión con remuestreo. La opción anterior resulta costosa en el número de puntos de la ruta, para evitarlo, se puede generar la ruta remuestreando cada recorrido del cluster a intervalos de distancia fija. De ese modo se calcula el punto medio de cada intervalo (latitud y longitud media).

Selección de canónico. Por último cabe plantearse la elección del recorrido más representativo del cluster y evitar fusiones y remuestreos. De ese modo nuestra ruta es el recorrido elegido sin su componente temporal.

En nuestro trabajo nos decidimos por la última opción. Para escoger el recorrido más representativo caben diversas posibilidades. Una de ellas es elegir al recorrido con menor distancia a todos los demás recorridos del cluster. Otra opción es elegirlo considerando la proximidad de sus extremos al origen y destino. Tras implementar las dos políticas, vimos que era más aconsejable la segunda opción, por encontrarse en algunas ocasiones recorridos en el mismo cluster con el problema del arranque del receptor.

Esto se ilustra en la Figura 2.6. En ella podemos observar tres recorridos pertenecientes a un mismo cluster cuyos extremos son los destinos 3 y 4. Aunque los recorridos se superponen, los hemos desplazado para permitir una correcta visualización. Utilizando la política elección de canónico por menor distancia a los demás, encontramos como representante el recorrido C. Sin embargo comprobamos que el que mejor representa la ruta es el B que está mucho más cerca del origen y el destino que los demás.

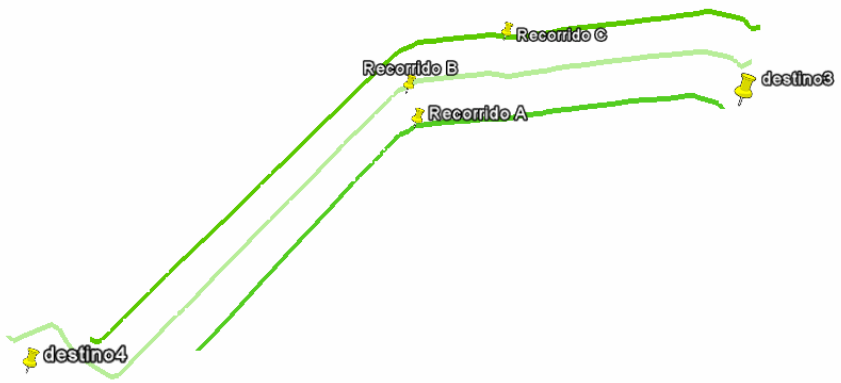

Figura 2.6: Superposición de tres recorridos desplazados manualmente contenidos en el mismo cluster.

2.2.11. Técnicas de predicción

Se evaluaron dos técnicas sobre el conjunto de datos obtenidos:

Modelos de Markov Ocultos Utilizando el mapa de puntos soporte obtenido en la fase de generación de mapas personales, se aplicó un HMM en el que los estados son los destinos y las observaciones los puntos soporte. Esta técnica se explicará en el Capítulo 3.

Similitudes dinámicas Utilizando el conjunto de rutas como mapa personal, se aplicó una medida de similitud dinámica que compara el recorrido de test con las rutas del mapa. Esta técnica se comentará en el Capítulo 4.

2.2.12. Evaluación

Para evaluar los resultados de la predicción de cada técnica, en cada recorrido de test, se obtuvieron las resultados para el 10, 25, 50, 75 y 90 por ciento del trayecto recorrido sobre el total. De ese modo pudimos evaluar los sistemas de predicción prácticamente al comienzo del recorrido y en pasos intermedios hasta casi la finalización del mismo. Cabe destacar que aún la predicción realizada en el tramo final (en el 90 % del recorrido) resulta útil en muchos casos donde la antelación de la predicción no es crítica. Esto lo veremos en las aplicaciones, concretamente en una herramienta útil para ciclistas.

Con objeto de cuantificar la bondad de la predicción para cada método hemos considerado el siguiente índice:

$$IP = 0{,}5 \sum_{i=0}^{n-1} (P_{i+1} - P_i)(Q_{i+1} + Q_i)$$

donde P_i representa el porcentaje de recorrido parcial atravesado (en tanto por uno) y Q_i el porcentaje de acierto en la predicción (en tantos por uno). Además consideramos que $P_0 = Q_0 = 0$ (sin desplazamiento el acierto es cero) y que $P_n = Q_n = 1$ (cuando estamos en el lugar final asumimos que se acierta el 100 % de las

veces) y n denota el número de predicciones realizadas (en nuestro experimento es 6 que se corresponde con el 10 %, 25 %, 50 %, 75 %, 90 % y 100 %). El índice considerado se basa en las mismas consideraciones que el índice de Gini, el cual es utilizado en la literatura económica para el estudio de desigualdades en el reparto de un volumen de unidades monetarias. Así se tiene que $0 < IP < 1$ y cuanto mayor sea IP mejor será el método de predicción ya que representa el área encerrada por el eje $\overrightarrow{OX}$, la recta $r : X = 1$; y la curva que describe el índice. En la figura 2.7 podemos observar la representación de este índice para un usuario ficticio.

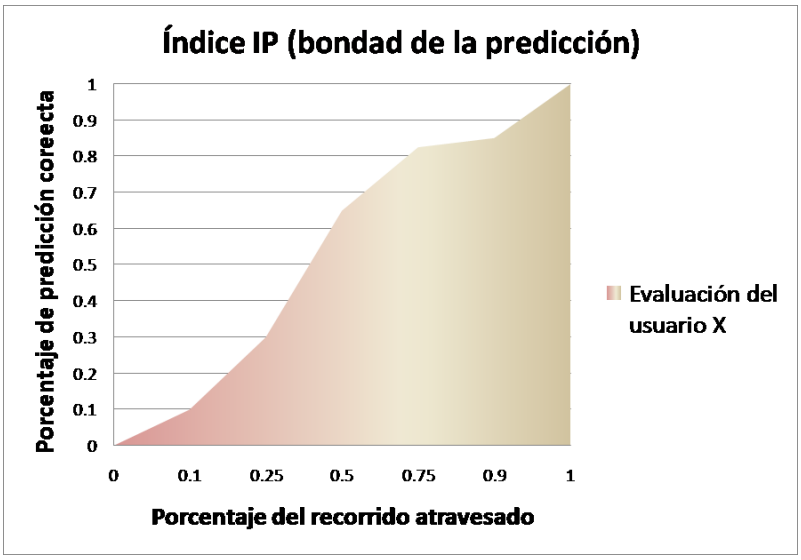

Figura 2.7: Representación del índice IP definido para cuantificar la bondad de nuestras predicciones.

2.3. Resumen

En este capítulo hemos estudiado en primer lugar las fases seguidas por otros autores para extraer información off-line a partir de una base de datos de puntos geo-posicionados obtenidos por personas durante sus desplazamientos habituales en exteriores. Tras ello hemos propuesto una metodología propia que permite recuperar información sobre destinos frecuentes, recorridos y mapas personales y a partir de

ella evaluar dos técnicas de predicción: HMM y similitudes dinámicas. Por último aportamos un índice que permite evaluar la bondad de los métodos de predicción basándose en su comportamiento durante la ejecución parcial del recorrido de test.

De este capítulo cabe destacar el enfoque novedoso de los mapas personales independientes del lugar donde se realicen las predicciones y por tanto de si existe cartografía de GIS. Así mismo, veremos como la fase de segmentado por cercanía a destinos mejora sustancialmente los resultados obtenidos con respecto a la técnica de segmentado temporal que es la adoptada por la mayoría de autores.

Una vez explicada la metodología off-line, pasamos a describir las técnicas utilizadas para la predicción. En el Capítulo 3 estudiaremos la utilización de los modelos de Markov así como la generación de mapas de puntos soporte y en el Capítulo 4 utilizaremos medidas de similitud tanto para predecir el destino final como para realizar el cluster de rutas (ver Sección 2.2.9). Finalmente en el Capítulo 5, veremos el comportamiento de ambas técnicas.

CAPÍTULO 3

MODELO DE MARKOV OCULTO SOBRE "MAPA SOPORTE"

Al no utilizar ningún GIS, propondremos un novedoso método para generar un mapa personal, que llamaremos "mapa soporte" y que permitirá aplicar modelos probabilísticos convencionales. En nuestro caso se seleccionó un modelo de Markov oculto.

En primer lugar, nos centramos en describir los trabajos existentes en la literatura relacionados con nuestra aproximación. Seguidamente haremos una breve descripción de los modelos de Markov y los modelos de Markov ocultos, que serán fundamentales para nuestro sistema de predicción, para a continuación detallar la generación de mapas soporte. Finalmente explicaremos la integración de estos mapas con el modelo probabilístico.

3.1. Estado del arte

En esta sección se muestran algunos trabajos que, al igual que en nuestra aproximación, utilizan modelos probabilísticos para obtener predicciones de rutas y des-

tinos de un usuario en un entorno geográfico.

Los primeros en utilizarlos en este campo fueron Ashbrook y Starner entre 2002 y 2003 [AS02, AS03]. Ellos propusieron modelos de Markov de segundo orden para realizar la predicción de destinos en la zona de Atlanta (Estados Unidos). Su objetivo era prever el siguiente destino más probable al que se dirigiría un usuario. Su propuesta utilizaba el historial de los lugares visitados en el pasado, de modo que conociendo el último lugar visitado, es decir el lugar en el que se encontraba la persona estudiada y el penúltimo lugar visitado, predecir el próximo desplazamiento. Su limitación fundamental es que además de no tener en cuenta las rutas que seguiría, no se consideraban ni el lugar actual del transeúnte o conductor ni si éste estaba en movimiento o no. Aún así, para el objetivo propuesto, la simplicidad del modelo usado les reportó buenos resultados. Un ejemplo del planteamiento de esta aproximación puede ser seguida de la Figura 3.1. En ella pueden verse las probabilidades de desplazamiento entre 3 lugares. Las flechas etiquetadas indican la fracción del número de desplazamientos que se hicieron desde el lugar origen hasta el lugar destino entre el número total de desplazamientos realizados desde ese lugar.

En la Tabla 3.1 podemos ver las probabilidades de transición entre dos (identificadores A y B) de los tres lugares de la figura. Los demás identificadores corresponden con otras localizaciones.

En 2003 desde la Universidad de Washington, Patterson propuso un modelo basado en una red Bayesiana dinámica [PLFK03], que representa un modelo de Markov de primer orden, en la que se infería el medio de transporte (coche, autobús o a pie) y el destino de un conductor de manera no supervisada. En la Figura 3.2 podemos ver su configuración y sus resultados. Al ser no supervisada, las predicciones no eran los lugares a los que se dirigía la persona, sino la distancia medida en número de manzanas hasta donde era posible conocer el destino. De ese modo se puede observar como en el 50 % de los casos era posible conocer la futura localización del usuario hasta en 17 manzanas si iba en autobús.

Un año más tarde, este grupo, mediante la iniciativa de Liao [LFK04] mejoró el

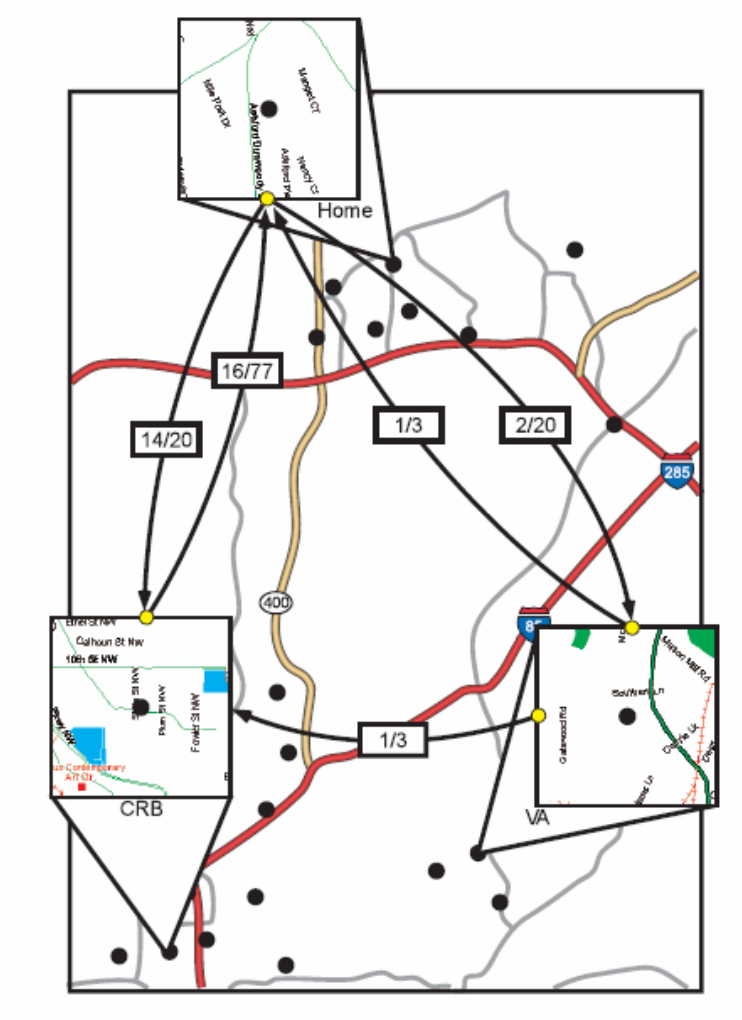

Figura 3.1: Ejemplo de modelo de Markov de segundo orden.

modelo anterior utilizando un modelo de Markov jerárquico que puede representarse de manera compacta por una red Bayesiana dinámica. Realizan la inferencia mediante un filtro de partículas "Rao-Blackwellizado" que combina el filtro de partículas y los filtros de Kalman. El modelado del mundo al igual que en su anterior trabajo es un grafo dirigido donde las aristas corresponden con las calles de la ciudad de Washington y los vértices con las intersecciones. Los resultados comparados con los de Ashbrook pueden verse en la Tabla 3.2.

En la Figura 3.3 podemos observar los instantes temporales k y $k-1$ del modelo

Tabla 3.1: Probabilidades de transición de los modelos de Markov de primer y segundo orden.

Transición	Frecuencia Relativa	Probabilidad
A→B	14/20	0.7
A→B→A	3/14	0.2142
A→B→C	2/14	0.1428
A→B→D	3/14	0.2142
A→B→E	3/14	0.2142
A→B→F	3/14	0.2142
B→A	16/77	0.2077
B→A→B	13/16	0.8125
B→A→J	3/16	0.1875

Tabla 3.2: Tabla de precisión de modelos de Markov de segundo orden (Ashbrook) y jerárquico (Liao).

Modelo	Precisión media en cada instante			
	Al inicio	25 %	50 %	75 %
Modelo de Markov 2º Orden	0.69	0.69	0.69	0.69
Modelo Jerárquico	0.66	0.75	0.82	0.98

que proponen. El modelo más completo y complejo, mejora en mucho al anterior de Patterson y al propuesto por Ashbrook. La capa superior permite la detección de novedades, la intermedia infiere destinos y segmentos de carretera. La capa inferior permite la detección del lugar, velocidad y medio de transporte usado. Es interesante observar que aunque el modelo jerárquico que proponen es mejor según avanza el recorrido, el modelo de Markov de segundo orden permite realizar una muy buena estimación de los destinos probables desde el inicio.

El mayor problema del modelo de Liao, es que al ser necesario la estructura de un mapa de calles de la ciudad, el modelo es local y almacena e intervienen muchísimos

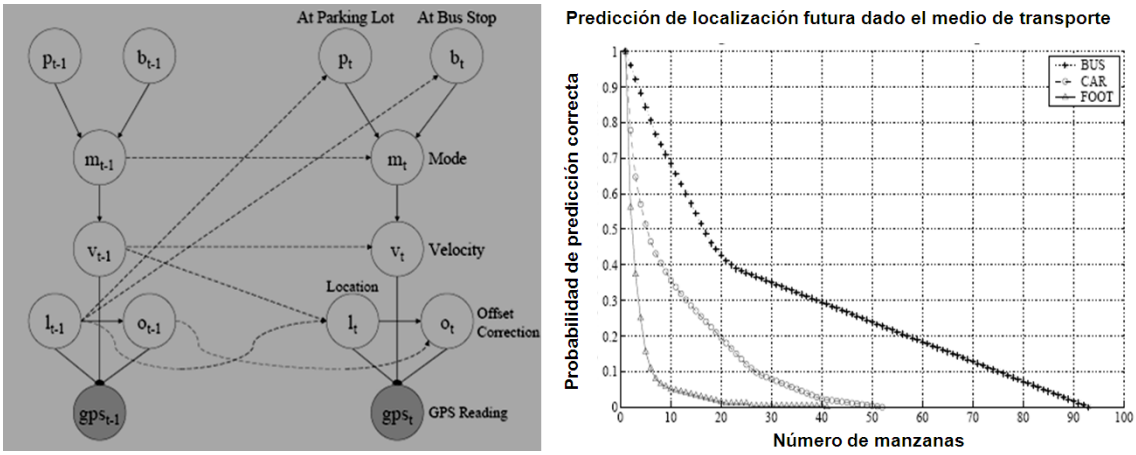

Figura 3.2: Red Bayesiana dinámica utilizada por Patterson (derecha) y gráfico de predicción de correcta localización (izquierda).

datos que difícilmente se usarán, dado que un usuario "tipo" recorre una mínima fracción de la red de calles de una ciudad.

En 2005, Gogate [GDB+05], incide en el uso de un modelo de red Bayesiano, concretamente en uno híbrido y dinámico para el problema de la predicción de rutas y destinos. Como ellos comentan tanto su modelo como varios más analizados no son más que representaciones de procesos de Markov que permiten tanto variables discretas como continuas, debido a que pretenden modelar aspectos de la vida real que contienen información probabilística y determinista.

En 2006 el grupo de Microsoft liderado por Krum [KH06, Kru06] propone un método de predicción original, llamado "Predestination". En él se analiza el área de Seattle generando una malla de 40 kilómetros cuadrados cuyas celdas miden 1 kilómetro cuadrado, siendo el número de celdas $N = 40 * 40 = 1600$.

Utilizan un conjunto de 7335 recorridos realizados por 169 conductores e infieren la distribución de probabilidad de que un usuario finalice un recorrido sobre cada una de las celdas. Aplicando el teorema de Bayes bajo distintas hipótesis "a-priori" obtienen dos modelos diferentes. El primero llamado "modelo cerrado", donde consideran que un usuario sólo visita una serie de lugares que ya ha visitado previamente y el segundo, "modelo abierto" en el que se supone que el usuario podrá despla-

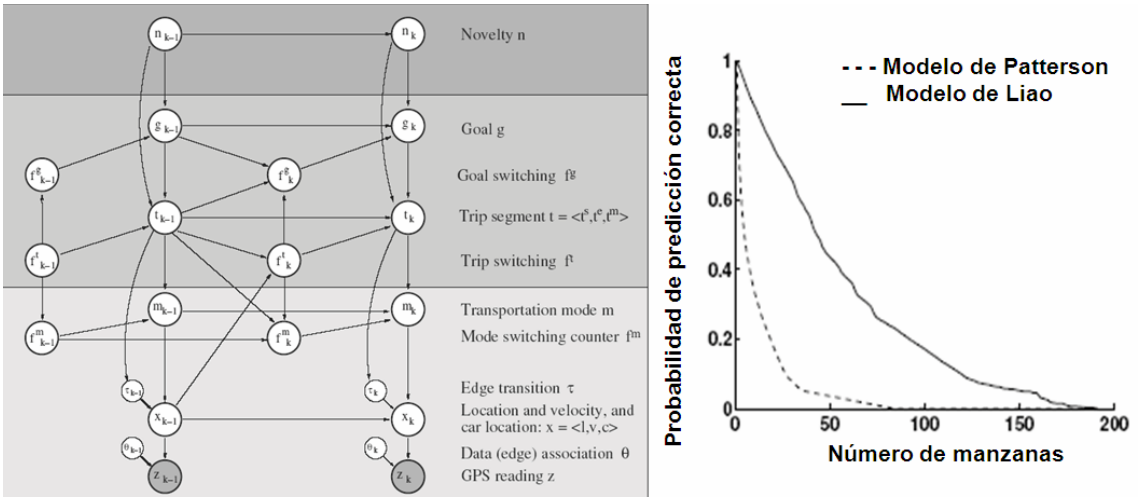

Figura 3.3: Modelo de Markov Jerárquico utilizado por Liao (derecha). Comparación del modelo propuesto por Patterson y el propuesto por Liao (izquierda).

zarse hacia lugares desconocidos (muy útil cuando se analiza por primera vez un determinado conductor).

En ambos modelos calculan la probabilidad de que cada celda sea el destino, es decir: $P(c_i|S)$ donde c_i representa la celda destino i-ésima de las N posibles y $S = \{s_1, s_2, ..., s_N\}$ es una variable aleatoria que representa el vector de características de las celdas atravesadas hasta el instante actual. Aplicando el Teorema de Bayes,

$$P(c_i|S) = \frac{P(S \bigcap c_i)}{P(S)} = \frac{P(S|c_i)P(c_i)}{\sum_{j=1}^{N} P(S|c_j)P(c_j)}, \tag{3.1}$$

donde $P(c_i)$ es la probabilidad "a priori" de que el destino sea la celda i y $P(S|c_i)$ es la probabilidad de que se de la secuencia S dado el destino c_i (se denominan "verosimilitudes").

En el modelo abierto, la probabilidad "a priori" considerada en cada celda es $P(c_i) = \frac{1}{N}$, dando a todas las celdas la misma probabilidad. En el modelo cerrado, se utiliza un histograma de celdas que el usuario utiliza como destinos, donde para calcular la probabilidad a priori, se basan en dos tipos de información:

Tipo de suelo e infraestructura existente. Utilizando un mapa de Estados Unidos con la cobertura del suelo categorizado en 21 tipos, calculan la probabilidad de que los destinos sean un tipo de suelo u otro. Contemplando únicamente la primera información, asignan más probabilidad a zonas comerciales y de industria que contienen infraestructura como carreteras y raíles de tren, y a zonas residenciales que a zonas como barrancos o lagos.

Lugares frecuentes. Basándose en los lugares en los que el usuario tiene como destino, dichos lugares poseen una probabilidad mayor de ser destino que otro en el que no haya estado aún.

Para calcular la probabilidad futura, se basan en otros dos tipos de información:

La probabilidad de una conducción eficiente. De manera intuitiva, si un conductor toma un camino muy ineficiente para llegar a un determinado lugar, seguramente es que no va a dicho lugar. Sabemos que cuando comenzamos un recorrido (estado s_0), el tiempo que resta para llegar al destino (t_{s_0}) será mayor que el tiempo restante cuando llegamos a él (estado s_N): $t_{s_N} = 0$. Si segmentamos el recorrido en trozos, basado en la división del mapa comentado, teóricamente se cumplirá que $t_{s_i} > t_{s_{i+1}}$. Tras realizar los cálculos con los datos de sus recorridos y utilizando un planificador de rutas obtuvieron que la probabilidad de que un conductor redujese el tiempo de llegada al avanzar de celda era $p = 0{,}625$. El planificador daba estimaciones de tiempos mejores que al utilizar la distancia euclídea con una velocidad constante aunque no era del todo preciso por no considerar instantes temporales en los que era más rápido utilizar un camino más largo (por ejemplo circunvalaciones) que otro más directo.

La distribución de la duración del viaje. Krumm y Horovitz también utilizan la duración del viaje para calcular la probabilidad de fin de trayecto. Se basaron en los datos de 66000 estadounidenses que mostraban que los viajes que más se repetían eran los relativamente cortos, de 5 a 19 minutos.

Utilizando el modelo abierto (aquel que no supone ningún lugar frecuentemente visitado) junto con el modelo cerrado (el que sí supone dichos lugares), la precisión llegaba a conseguir un error medio de 2 kilómetros del lugar destino habiendo recorrido el conductor la mitad del trayecto. Un ejemplo de este tipo de predicción puede verse en la Figura 3.4. En esta secuencia se observa cómo la probabilidad de cada celda de ser el destino varía según el recorrido avanza. En la imagen de la izquierda todas las celdas son equiprobables por comenzar el recorrido en el centro. En la imagen del centro, tras atravesar 4 celdas hacia el Sur, se reducen las probabilidades de las del norte, finalmente a la derecha vemos las celdas más probables son las del Sudoeste.

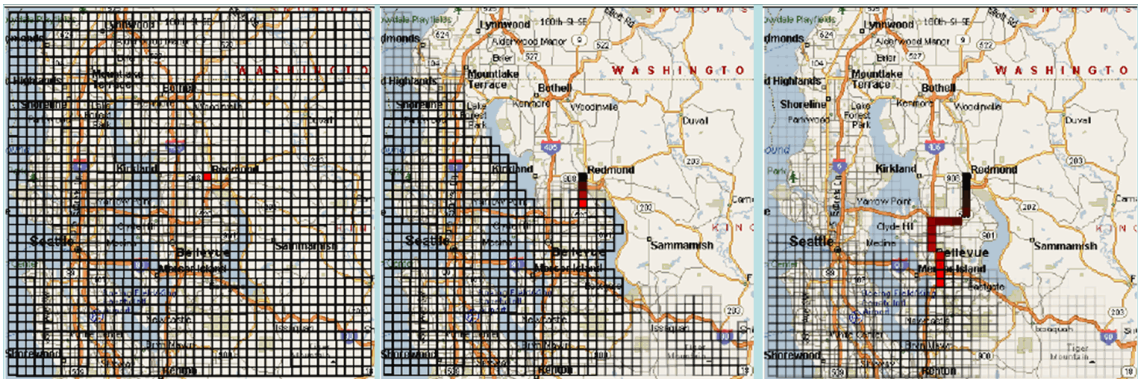

Figura 3.4: Secuencia de ejemplo del modelo "Predestination".

En 2006, Simmons [SBZS06] diseña un Modelo de Markov Oculto (HMM) peculiar para el objetivo de predicción basándose en un modelo de Markov de primer orden. Al igual que en las anteriores propuestas, utiliza una pequeña parte del mapa de la ciudad a la que va dirigido (en este caso Detroit). En el modelo de primer orden, considera como estados cada uno de los puntos intermedios de las calles (su longitud y latitud). Su estudio se centra en predecir la siguiente calle que tomará el conductor, conocida la calle que está atravesando. Los resultados son muy buenos pero están condicionados por la red de calles que estudia dado que el 95 % de ellas tiene una única calle como desembocadura (considera el sentido en la que se atraviesa). De ese modo el algoritmo garantiza la corrección de la siguiente calle en gran parte del test. Además, incluye la información del lugar destino hacia el que

se dirige (el objetivo que nosotros estudiamos), para a partir de ese modelo visible de primer orden, inferir la variable oculta que en este caso se trata del destino final. Sin embargo de esa predicción de destinos no aporta resultados.

En 2008, Krumm [Kru08] utiliza modelos visibles de Markov para conseguir determinar las calles que seguirá un determinado conductor. La idea es la misma que la de Simmons, sin embargo, en vez de usar un modelo de primer orden, prueba modelos de mayor orden (desde 2 hasta 10 órdenes). En la Figura 3.5 vemos un ejemplo en el que se consideran los últimos 4 segmentos de carretera para inferir el siguiente. En su trabajo se explica como la predicción mejora cuantos más segmentos de carretera se consideren, sin embargo para recorridos en los que se atraviesan pocos segmentos los modelos de mayor orden pueden no servir de mucho. Por otro lado en el trabajo de Simmons el porcentaje de calles con una única desembocadura era del 95 %, en este caso sólo es del 28 %, siendo la ciudad estudiada Seattle.

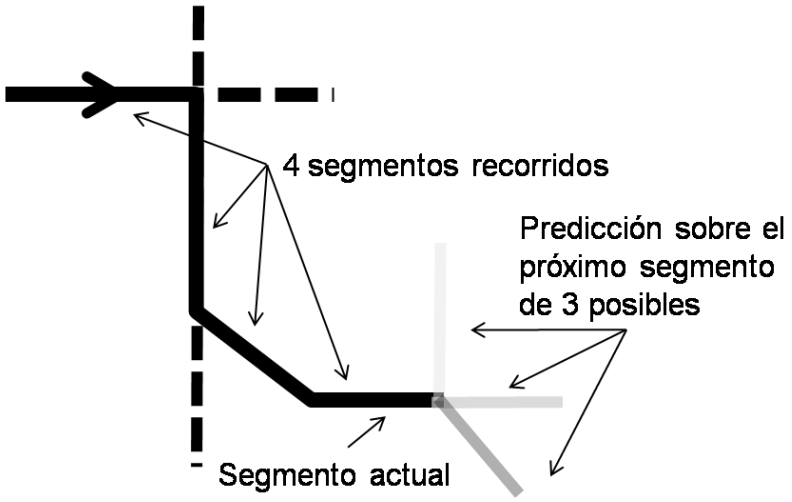

Figura 3.5: Modelo de Markov de 4º orden en el que los estados son los segmentos de carretera.

Teniendo en cuenta lo comentado anteriormente, hemos considerado como herramienta básica para llevar a cabo nuestro trabajo, utilizar como modelo probabilístico los modelos de Markov, que desarrollaremos en la siguiente sección.

3.2. Modelos de Markov

Los procesos de Markov fueron desarrollados inicialmente por Andrei A. Markov en 1907. En esta sección se estudian éstos así como su evolución hasta los modelos de Markov ocultos.

3.2.1. Modelos de Markov visibles

El objetivo inicial de Andrei A. Markov era modelar secuencias de valores de una variable aleatoria en las que su valor futuro depende del valor de la variable en el presente, independientemente de la historia de dicha variable (Modelos de Markov de primer orden). Sin embargo, esta aproximación es generalizable como veremos a continuación.

Consideremos un sistema que en cada instante de tiempo, se encuentre en un determinado estado, $S_1, S_2, \ldots, S_N$. Transcurrido un espacio de tiempo, y de forma regular, el sistema cambia de estado, pudiendo volver al mismo. Los instantes de tiempo se denotan como $t = 1, 2, \ldots, T$ y el estado actual como q_t. Una descripción probabilística completa del sistema requeriría la especificación del estado actual, así como de todos los estados precedentes. Sin embargo, las cadenas de Markov presentan dos características muy importantes:

- *Horizonte limitado.* Permite truncar la dependencia probabilística del estado actual y considerar, no todos los estados precedentes, sino únicamente un subconjunto finito de ellos. En general, una cadena de Markov de orden n es la que utiliza los n estados previos para predecir el siguiente. Por ejemplo, para las cadenas de Markov de tiempo discreto de primer orden, tenemos que:

$$P(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, \ldots) = P(q_t = S_j | q_{t-1} = S_i), \qquad (3.2)$$

  donde se cuantifica la probabilidad que el estado actual sea S_j, sabiendo que en el instante anterior fue S_i.

- *Tiempo invariante (estacionario).* Nos permite considerar sólo aquellos procesos en los cuales la parte derecha de la ecuación 3.2 es independiente del tiempo. Esto nos lleva a considerar una matriz $A = a_{ij}$ de probabilidades de transición entre estados de la forma:

$$a_{ij} = P(q_t = S_j | q_{t-1} = S_i) = P(j|i), \qquad 1 \leq i, j \leq N \tag{3.3}$$

independientes del tiempo pero con las restricciones estocásticas estándar:

$$a_{ij} \geqslant 0, \quad \forall i, j \qquad y \qquad \sum_{j=1}^{N} a_{ij} = 1, \quad \forall i. \tag{3.4}$$

Sin embargo es necesario especificar también el vector $\pi = \{\pi_i\}$, que almacena la probabilidad que tiene cada uno de los estados de ser el estado inicial:

$$\pi_i = P(q_1 = S_i), \quad \pi_i \geqslant 0, \quad 1 \leq i \leq N, \quad \sum_{i=1}^{N} \pi_i = 1. \tag{3.5}$$

A un proceso estocástico que satisface estas características se le denomina modelo de Markov observable o visible, porque su salida es el conjunto de estados por los que pasa en cada instante de tiempo, y cada estado se corresponde con un suceso observable.

3.2.2. Modelos de Markov ocultos

Los modelos visibles pueden resultar demasiado restrictivos a la hora de aplicarlos a problemas reales. Por esta razón se extiende el concepto de los modelos de Markov de forma que sea posible incluir aquellos casos en los cuales la observación es una función probabilística del estado. El modelo resultante es el denominado modelo de Markov oculto o HMM (Hidden Markov Model). En él coexisten dos procesos estocásticos, uno visible y otro que no se puede determinar directamente (invisible u oculto). El modelo permite detectar el estado del proceso invisible a través de una secuencia de observaciones del proceso visible.

La teoría de estos modelos se desarrolló a finales de los 60 y principio de los 70 por Baum y su equipo [Bau72, BE67, BP66]. Los HMM's ganaron popularidad gracias

a su exitosa aplicación en el reconocimiento del habla [Rab89], aunque también se han aplicado al reconocimento de escritura y de gestos así como en diferentes tareas de bioinformática, entre ellas la de la búsqueda de genes en las cadenas de ADN.

Definamos ahora formalmente cuales son los elementos que forman un modelo de Markov oculto, los cuales se resumen en la siguiente tupla (Q, V, π, A, B):

- Q: Es el conjunto de estados del modelo. Los estados se etiquetan como $1, 2, \ldots, N$, y el estado actual en el instante de tiempo t es q_t.
- V: Es el conjunto de sucesos que pueden observarse en los distintos estados del modelo. Cada uno de ellos se denotará como v_k con $k = 1, \ldots, M$.
- $\pi = \{\pi_i\}$ (Vector de estados iniciales): Contiene la probabilidad de cada estado de ser el estado inicial tal y como vimos en (3.5).
- $A = \{a_{ij}\}$ (Matriz de transición): Contiene las probabilidades de transición entre estados (ver ecuaciones (3.3) y (3.4)). De esta forma a_{ij} nos cuantifica la probabilidad de estar en el instante t en el estado S_i, cuando en el instante anterior estaba en el estado S_j, con $i, j = 1, \ldots, N$.
- $B = \{b_j(v_k)\}$ (Matriz de emisión): Contiene las probabilidades de aparición de los distintos sucesos para cada estado del modelo de Markov oculto. También se conoce como conjunto de probabilidades de emisión. Se cumple, con la restricción estocástica

$$b_j(v_k) = P(o_t = v_k | q_t = j) = P(v_k|j), \qquad b_j(v_k) \geqslant 0 \tag{3.6}$$

  donde o_t es la observación en el instante t, $1 \leq j \leq N$, $1 \leq k \leq M$ y $1 \leq t \leq T$ cumpliéndose la siguiente restricción,

$$\sum_{k=1}^{M} b_j(v_k) = 1, \quad \forall j = 1, 2, \ldots, N. \tag{3.7}$$

Señalar que $b_j(v_k)$ cuantifica la probabilidad de que ocurra el suceso v_k estando en el estado S_j.

Normalmente, al hacer referencia a un modelo de Markov oculto, solo es necesario especificar los tres últimos parámetros de la tupla que acabamos de describir, ya que el resto aparecen implícitos en estos tres. La notación compacta quedaría de la forma $\mu = (\pi, A, B)$.

3.2.3. Problemas a resolver usando HMM

Al utilizar un HMM en aplicaciones reales, hay que solventar las siguientes preguntas recogidas por Rabiner [Rab89]:

1. Dada una secuencia de observaciones $O = (o_1, o_2, \ldots, o_T)$ y dado un modelo $\mu = (\pi, A, B)$, ¿cómo calculamos de manera eficiente $P(O|\mu)$, es decir, la probabilidad de dicha secuencia dado el modelo? Para resolver este problema se utiliza el algoritmo de avance-retroceso [BE67].

2. Dada una secuencia de observaciones $O = (o_1, o_2, \ldots, o_T)$ y dado un modelo $\mu = (\pi, A, B)$, ¿cómo elegimos la secuencia de estados $S = (q_1, q_2, \ldots, q_T)$ óptima, es decir, la que mejor explica la secuencia de observaciones? Este problema se resuelve con el algoritmo de Viterbi [Vit67].

3. Dada una secuencia de observaciones $O = (o_1, o_2, \ldots, o_T)$, ¿cómo estimamos los parámetros del modelo $\mu = (\pi, A, B)$ para maximizar $P(O|\mu)$, es decir, ¿cómo podemos encontrar el modelo que mejor explica los datos observados? Este problema se resuelve con el algoritmo de Baum-Welch [Bau72].

3.3. Nuestra propuesta

Nuestro sistema, difiere de los estudiados en la Sección 3.1 en dos características fundamentales:

- Debe ser integrable en dispositivos móviles con características de almacenamiento y baterías reducidas; y

- Debe ser utilizable por cualquier usuario en cualquier parte del mundo.

Además como indicamos en la Sección 1.1 las necesidades para realizar predicciones en cuanto a mapas y cartografía no son las mismas que un sistema de navegación y guiado de vehículos por lo que podemos evitar conexiones a GIS costosas y almacenar cartografía de diferentes países sino sólo de nuestro mapa personal.

Vistos los diferentes trabajos de predicción de destinos basados en modelos probabilísticos y concretamente en los modelos de Markov, decidimos proponer un sistema basado en un HMM. En él consideramos los estados como los lugares hacia los que tenemos la intención de desplazarnos en un instante determinado. Esta es la parte oculta del modelo y lo que pretendemos inferir. Para ello, contamos con las observaciones visibles que nos proporciona el receptor GPS de donde nos encontramos en cada momento.

Dado que esas observaciones serían inabordables por la cantidad de puntos GPS diferentes que podemos obtener únicamente considerando la latitud y la longitud en cualquier punto del mundo, es necesario reducir éstas a los puntos que aportan información relevante para el usuario estudiado. Veremos cómo generar un mapa personal o mapa soporte a partir del historial de recorridos de cada usuario. Este mapa tendrá información significativa de todas las ciudades o incluso zonas sin cartografiar que frecuente el usuario. Consultando este mapa, nuestro sistema podrá inferir el destino más probable al que se dirige. A continuación pasamos a describir la generación del mapa soporte y la definición de los parámetros fundamentales de nuestro HMM.

3.3.1. Generación del mapa soporte

La generación de este mapa, debe contener información de los recorridos pasados del usuario. El problema es que cada recorrido tiene cientos de puntos y cada usuario puede tener una cantidad importante de recorridos. Si queremos incluir todos los puntos en un modelo estadístico, tendríamos un modelo tan grande como inútil. Para reducir el conjunto inicial de datos, podemos utilizar las rutas descritas en la Sección 2.2.9 en vez del conjunto de todos los recorridos. Nuestro objetivo es identificar cada ruta por un número reducido de puntos para incorporarlos a nuestro HMM. Los puntos que incluyamos deben ser significativos de modo que nos ayuden a predecir hacia donde se dirige el usuario simplemente sabiendo que éste está cerca del punto y su dirección.

En la Figura 3.6 podemos ver un ejemplo de qué tipo de mapa queremos crear. En él pueden observarse tres destinos que en nuestro modelo corresponden con estados no visibles. Los aros corresponden con los puntos significativos que pasamos a explicar a continuación.

3.3.1.1. Selección de puntos de cruce

Consideremos la situación mostrada en la Figura 3.7: un usuario avanza por un camino y aparece ante él un cruce en el que puede escoger 4 posibles desviaciones. Descifrar su destino en ese instante será mucho más complejo que si esperamos unos segundos a que escoja una de las bifurcaciones. En este último caso, los destinos que se alcanzan por los tres caminos que ha dejado atrás resultan mucho menos probables que el que se alcanza siguiendo la ruta elegida. Esta es la justificación del porqué decidimos seleccionar los puntos tras un cruce de recorridos como los puntos significativos.

Para evitar confusiones, a partir de ahora, el término “cruce” indicará un cruce real o un solapamiento de recorridos seguido de una separación de éstos como se puede ver en la Figura 3.8.

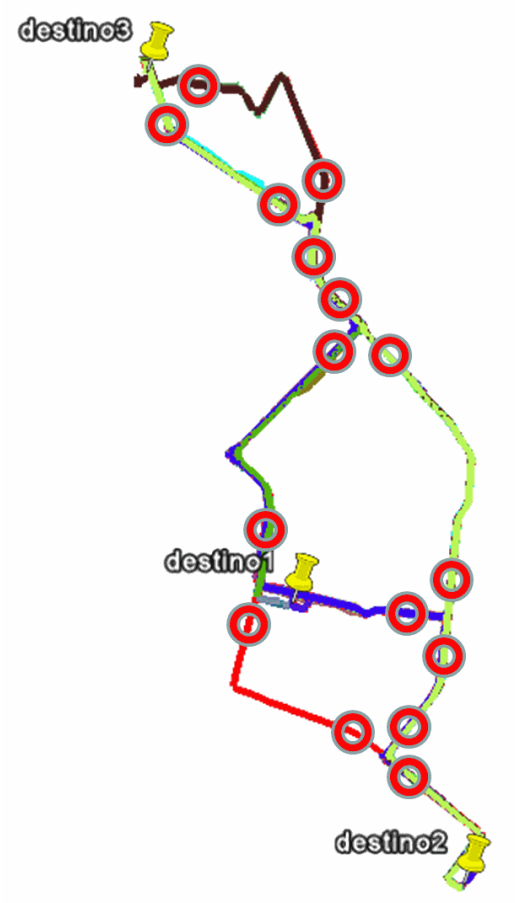

Figura 3.6: Ejemplo de mapa soporte objetivo.

Como comentamos previamente, nuestro sistema no posee un mapa de carreteras o calles dado que no tenemos la cartografía mundial, por lo que no podemos escoger los cruces entre calles sino que utilizaremos los cruces de rutas. Además, como las rutas no tienen asociado un sentido, los puntos significativos aparecen antes y después de cada cruce.

En el proceso de generación de puntos significativos, seleccionamos todos los puntos comunes que hay entre cada par de recorridos. Eso nos permite saber las zonas en las que se unen o bifurcan éstos.

Definición 9 (Punto de cruce) *Dado el conjunto Rutas* $= \{R_1, \ldots, R_n\}$*, para cada par de rutas* R_i, R_j *definiremos el conjunto de puntos de cruces que notaremos*

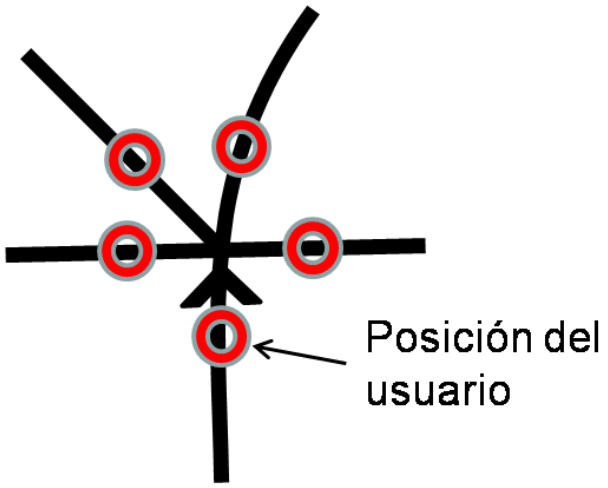

Figura 3.7: Camino con cuatro desviaciones.

como $Cruce^{ij} = \{cruce_1^{ij}, \ldots, cruce_m^{ij}\}$, *siendo* $cruce_k^{ij} = \{lat_k^{ij}, lon_k^{ij}\}$ *un punto de cruce cuyas componentes básicas son latitud y longitud. El conjunto de todos los puntos de cruce entre cada par de rutas lo notaremos como sigue:* $Cruces = \{cruce_1, \ldots, cruce_m\}$

A partir de los extremos que nos indican las bifurcaciones y uniones, analizamos el tipo de cruce, pudiendo ser extremo-extremo, extremo-punto medio o punto medio-punto medio, indicando los extremos los puntos iniciales o finales y el 'punto medio' una zona intermedia del recorrido. Dependiendo del tipo de cruce, se genera por cada uno, desde dos hasta cuatro puntos significativos como podemos ver en la Figura 3.8. Podemos observar como se trata de una matriz simétrica en la que sólo varían cuál es la ruta 1 y cuál la 2, por lo que sólo tenemos cuatro tipos de cruces. Una vez que finaliza este proceso, llamaremos a los puntos significativos seleccionados "candidatos".

3.3.1.2. Generación de puntos significativos

Definición 10 (Punto significativo) *Dado el conjunto Cruces, definiremos el conjunto de puntos significativos que notaremos como* $Significativos =$

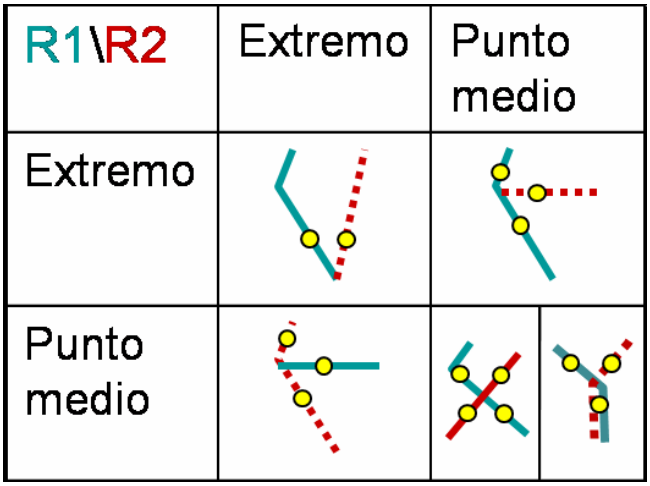

Figura 3.8: Tipos de cruces entre dos recorridos y sus puntos previos y posteriores.

$ClusterJerarquico(Cruces) = \{sign_1, \ldots, sign_m\}$, *siendo* $sign_i = \{lat_i, lon_i\}$ *el punto resultante de agrupar un conjunto de cruces mediante un cluster jerárquico por distancia umbral media.*

Dado que estamos analizando los cruces por cada par de recorridos, habrá muchos de ellos en los que el punto candidato se genere prácticamente en el mismo lugar que uno existente ya. Cuando eso ocurre, debemos agruparlos, realizando un cluster de puntos que elimine puntos redundantes y reduzca la complejidad del modelo. Este agrupamiento se representa en la Figura 3.9. Utilizamos para ello un cluster jerárquico con criterio de distancia umbral que indica cuánto de cerca deben estar dos candidatos para considerarse como uno solo.

3.3.1.3. Puntos soporte

Definición 11 (Punto soporte) *Dados los conjuntos* $Recorridos = \{Rec_1, \ldots, Rec_n\}$ *y* $Significativos = \{sign_1, \ldots, sign_m\}$, *para cada recorrido* Rec_i *se observarán los puntos significativos que atraviesa y en qué sentido lo hace (Norte, Sur, Este u Oeste). Dado un punto significativo* $sign_i = \{lat_i, lon_i\}$ *definiremos un punto soporte y lo notaremos como* $sop_i = \{lat_i, lon_i, ori_i\}$ *de modo que tenga las mismas latitudes*

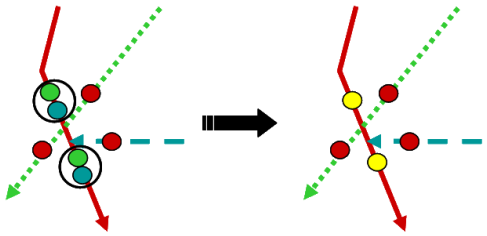

Figura 3.9: Operación de agrupamiento de puntos previos y posteriores de un cruce cuando están muy próximos.

y longitudes que el punto significativo y además incluya la componente del sentido en el que es atravesado (ori_i). Como cada punto significativo puede ser atravesado desde uno a cuatro sentidos, por cada punto significativo podrán existir de uno a cuatro puntos soporte.

Los puntos significativos, están identificados por una latitud y una longitud. Sin embargo que el receptor GPS indique exactamente esa latitud y longitud resulta complejo por los consabidos problemas de precisión. Debido a ello, se considerarán como círculos centrados en esa posición y con un determinado radio para permitir que un usuario atraviese ese círculo aunque no pase exactamente por su centro. Además, cada círculo significativo, podrá ser atravesado en diferentes sentidos de marcha, por ejemplo un punto puede ser atravesado en sentido norte si utilizamos el camino de ida o sur si utilizamos el de vuelta. Para que la información sea suficientemente significativa, asociamos una de las cuatro direcciones cardinales según sea el sentido de la marcha. Para asociar cada una de las 4 direcciones aplicamos una regla simple: Si la diferencia de latitudes entre los puntos anterior al significativo y éste es mayor que la diferencia de longitudes, escogemos la orientación Norte o Sur dependiendo del resultado de la resta (mayor o menor a cero), en caso contrario asignaremos las

direcciones Este u Oeste considerando el resultado de la resta de longitudes.

$$Orientacion = si\,(|\Delta lon| - |\Delta lat|) \begin{cases} > 0 \quad \rightarrow \begin{cases} \Delta lon > 0 \rightarrow \quad Este \\ \Delta lon \leq 0 \rightarrow \quad Oeste \end{cases} \\ \leq 0 \quad \rightarrow \begin{cases} \Delta lat > 0 \rightarrow \quad Norte \\ \Delta lat \leq 0 \rightarrow \quad Sur \end{cases} \end{cases}$$

En la Figura 3.10 se puede ver un ejemplo de una situación real de puntos soporte. En ella se pueden ver tres segmentos de rutas (una de ellas solapada con otra) que generan un cruce de tipo punto medio - punto medio y que producen ocho puntos soporte, dos por cada punto significativo debido a que existen recorridos que atraviesan en ambas direcciones esas calles. Cada punto está lo suficientemente lejos del cruce para evitar que la imprecisión del sistema GPS indique puntos erróneos. Como cada punto significativo puede ser atravesado en diferentes direcciones, añadimos una etiqueta indicando la dirección de éste y generando así cada uno de los puntos soporte.

Para resumir el proceso, ilustramos la generación de puntos soporte en la Figura 3.11. En ella se observan tres recorridos parcialmente solapados. En 1), se marcan como círculos negros las zonas comunes entre cada par de recorridos, en 2) tras analizar los tipos de cruces (todos punto medio - punto medio) se usan estrellas para destacar las zonas posteriores y anteriores a las zonas comunes para posteriormente en 3), realizar el cluster de candidatos cercanos. Finalmente en 4) se observa como para los tres puntos centrales, se generan seis puntos soporte añadiendo sus posibles direcciones. Se ha evitado mostrar otros seis puntos soporte (las 3 estrellas exteriores con dos direcciones cada una) para mejorar la visibilidad de la ilustración.

Mostramos también la aplicación del proceso de obtención de puntos soporte sobre un conjunto de estudio real en la Figura 3.12. En ésta puede verse la situación inicial de los puntos GPS que formaban los recorridos de un usuario y los puntos que finalmente generan el mapa soporte. En a) puede observarse el conjunto de todos los puntos GPS etiquetados con la fecha de obtención. En b) se muestran los puntos

Figura 3.10: Puntos soporte generados en una situación real

soporte obtenidos tras el proceso de generación del mapa. Al reducirse drásticamente el número de puntos, el modelo probabilístico que va a resultar es mucho más ligero.

Definición 12 (Mapa soporte) *Al conjunto de todos los puntos soporte lo llamaremos "mapa soporte" y lo notaremos como* $Soporte = \{sop_1, \ldots, sop_n\}$.

Las ventajas de este mapa soporte ligero son además de la independencia de cualquier GIS y la posibilidad de hacer predicciones en cualquier lugar que frecuente el usuario, es que dado que no se utilizan mapas cartografiados, se evita el problema de "map-matching", es decir, el proceso de discernir a qué calle corresponde un determinado punto GPS y que se aborda en los demás trabajos estudiados [SBZS06, Kru08] mediante diversos algoritmos.

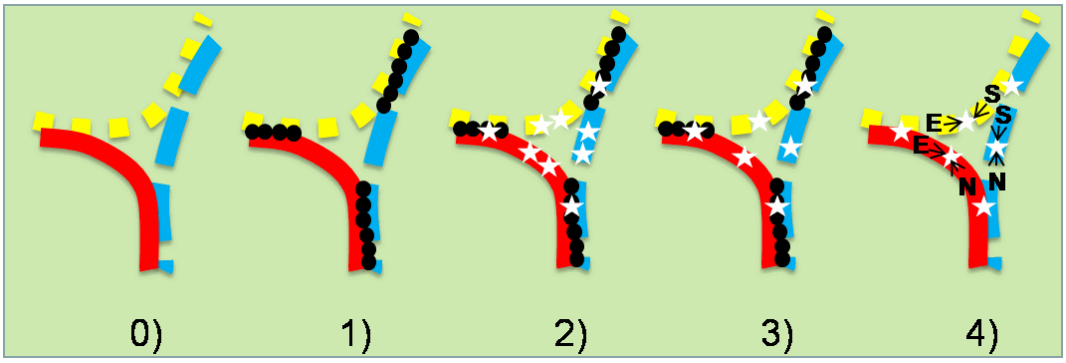

Figura 3.11: Secuencia de generación de puntos soporte.

3.3.2. Generación del modelo

Una vez creado el mapa soporte, debemos especificar el modelo de Markov oculto que usaremos. En primer lugar debemos decidir el diseño de la estructura, es decir, qué estados hay y cómo están conectados. En segundo lugar asignaremos las probabilidades de transición (A), emisión (B) y estado inicial (π).

Como comentamos anteriormente el proceso invisible que queremos obtener es el destino que se pretende alcanzar. Para ello utilizamos las observaciones del proceso visible, es decir, los puntos soporte que nos indicará nuestro receptor GPS cuando pasemos cerca de ellos.

A continuación, especificaremos los componentes del modelo de Markov oculto: (Q, V, π, A, B):

- Q: Nuestros estados corresponden con los lugares hacia los que nos dirigimos en un desplazamiento. Para tener una predicción basada en el histórico de desplazamientos, consideraremos sólo aquellos lugares que visitamos frecuentemente. N será el número de estados.
- V: Corresponderán con los puntos soporte obtenidos anteriormente compuestos por la localización (latitud, longitud) y la orientación (N, S, E, O). M será el número de observaciones.

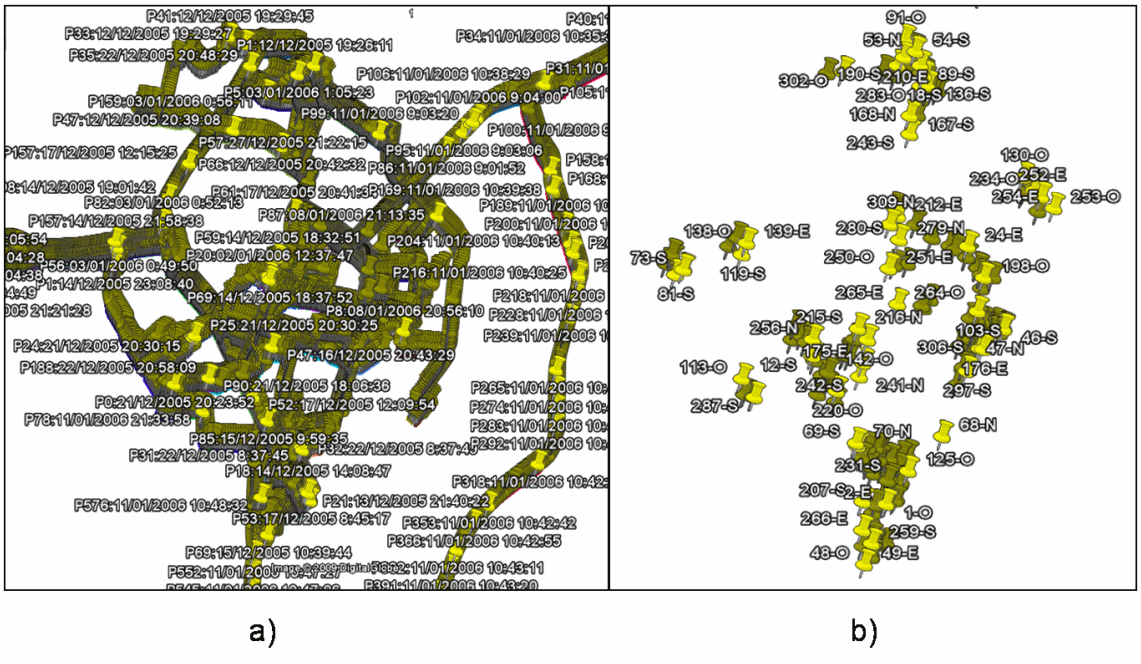

Figura 3.12: Ejemplo gráfico de la drástica reducción en el número de puntos.

- $\pi = \{\pi_i\}$: Contiene la probabilidad del lugar al que nos dirigiremos por primera vez. Normalmente el estado con mayor probabilidad de estado inicial es aquel al que suele dirigirse con más frecuencia, en muchos de los casos se trata del hogar de la persona estudiada.

- $A = \{a_{ij}\}$: Esta matriz, indicará la probabilidad de que estando en un estado, cambiemos a otro. Es decir, que dirigiéndonos a un destino, cambiemos nuestra ruta para ir hacia otro lugar destino.

- $B = \{b_j(v_k)\}$: Contiene las probabilidades de que dirigiéndonos a un lugar destino, pasemos por un punto de soporte.

Una vez explicados los componentes del modelo, es necesario explicar cómo se genera cada uno de ellos. Existen dos posibilidades para ello:

Sin datos previos. No disponer de secuencias pasadas de observaciones y los estados a los que corresponden. Sería necesario realizar una estimación inicial de

los valores de las matrices para posteriormente aplicar el algoritmo de Baum-Welch (ver Sección 3.2.3), que permite estimar los parámetros del modelo que hacen máxima la probabilidad de una secuencia de observaciones a partir de las matrices de transición y emisión estimadas.

Con datos previos. Disponer de secuencias pasadas de observaciones y los estados del proceso no visible que correspondían a cada una. En este caso es más sencillo obtener las matrices. Este caso es el más frecuente.

Nosotros estamos en el segundo caso ya que contamos con un conjunto de entrenamiento formado por una selección de los recorridos pasados (utilizaremos una validación cruzada de K capas) y sus estados asociados a cada una de las observaciones. Dado que son recorridos pasados, consultamos el destino final de cada uno de los recorridos y lo asociamos a cada una de las observaciones registradas (aprendizaje supervisado).

Veamos un ejemplo: Sea un conjunto de secuencias de entrenamiento $X = \{X^1, \ldots, X^n\}$. Cada secuencia representa el conjunto de puntos soporte obtenidos al atravesar un recorrido: $X^i = \left\{X^i_1, \ldots, X^i_{m_i}\right\}$, siendo X^i_j un punto soporte que tiene como atributos fundamentales su latitud, longitud y orientación. Además, cada recorrido X^i tiene un único destino D^i, por lo que podemos asociar la intención de llegar a ese destino a cada una de las observaciones del recorrido. De ese modo, la secuencia de observaciones será: $\left\{X^1_1, \ldots, X^1_{m_1}, X^2_1, \ldots, X^2_{m_2}, X^n_1, \ldots, X^n_{m_n}\right\}$ y la secuencia de estados asociada: $\{D^1, \ldots, D^1, D^2, \ldots, D^2, D^n, \ldots, D^n\}$.

Con esas secuencias de observaciones y estados, podemos obtener las matrices de transición y observación, así como la de estados iniciales. Para la matriz de transiciones contamos el número de veces que se produce una transición entre cada estado i y j, colocando el resultado en la posición aij, para luego normalizarla. Actuamos del mismo modo en la matriz de emisión, en la que se suman el número de apariciones de una determinada observación j estando en un determinado estado i para situar el resultado en bij y al igual que en la anterior, normalizar la matriz. Por último, el vector de estados iniciales puede completarse considerando cualquier

estado equiprobable o considerar la frecuencia de aparición de cada estado i para así indicar los valores a cada posición de π_i.

3.3.3. Predicción de destinos

Completo ya el modelo de Markov Oculto, pasamos a la predicción del lugar destino al que se dirige un usuario cuando comienza un recorrido nuevo. Para ello, vamos recogiendo aquellas localizaciones GPS derivadas de la ruta seguida por el mismo. Basándonos en el conjunto de los recorridos no usados para el entrenamiento, simulamos su ejecución tal y como lo haría un conductor o transeúnte, avanzando uno a uno por sus puntos GPS. Cuando se recoge un punto suficientemente cercano a una de las observaciones del modelo o puntos soporte, aplicamos el algoritmo de Viterbi [Vit67] para ver el destino más probable. Dado que la política de elección de observaciones ha hecho que éstas se reduzcan de manera significativa por cada uno de los recorridos, el algoritmo se aplicará un número reducido de veces, disminuyendo así el número de predicciones repetidas y mejorando el rendimiento.

Para hallar la secuencia de estados más probable dada una secuencia de observaciones (segundo problema de los tres más frecuentes en HMMs 3.2.3), el criterio más ampliamente utilizado consiste en encontrar la mejor secuencia de estados posible considerando globalmente todos los instantes de tiempo, es decir, la secuencia de estados $S = (q_1, q_2, \ldots, q_T)$ que maximiza $P(S|O, \mu)$, lo cual equivale a maximizar $P(S, O|\mu)$. Este es el procedimiento que sigue el algoritmo de Viterbi que se explica a continuación, basado en técnicas de programación dinámica.

Para encontrar la secuencia de estados más probable, $S = (q_1, q_2, \ldots, q_T)$, dada la observación $O = (o_1, o_2, \ldots, o_T)$, consideramos la variable $\delta_t(i)$ definida como

$$\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P(q_1, q_2, q_{t-1}, q_t = i, o_1, o_2, \ldots, o_t|\mu), \tag{3.8}$$

es decir, $\delta_t(i)$ almacena la probabilidad del mejor camino que termina en el estado i , teniendo en cuenta las t primeras observaciones. Se demuestra que:

$$\delta_{t+1}(j) = \left[\max_{1 \leq i \leq N} \delta_t(i) a_{ij}\right] b_j(o_{t+1}). \tag{3.9}$$

Una vez calculadas las $\delta_t(i)$ para todos los estados y para todos los instantes de tiempo, la secuencia de estados se construye hacia atrás, mediante una traza que recuerda el argumento que maximizó la ecuación (3.9) para cada instante t y para cada estado j. Esta traza se almacena en las correspondientes variables $\psi_t(j)$. La descripción completa del algoritmo es la siguiente:

1. Inicialización:
$$\delta_1(i) = \pi_i b_i(o_1),\ 1 \leq i \leq N \tag{3.10}$$
Se produce cuando nos encontramos el primer punto soporte y nos indica el o los caminos más probables (dado que puede haber ramificaciones, lo normal es que al encontrar este primer punto aparezcan diferentes estados equiprobables).

2. Recurrencia:
$$\delta_{t+1}(j) = \left[\max_{1\leq i\leq N} \delta_t(i)a_{ij}\right] b_j(o_{t+1}) \tag{3.11}$$
$$\psi_{t+1}(j) = \arg\max_{1\leq i\leq N} \delta_t(i)a_{ij} \tag{3.12}$$
donde $t = 1, 2, \ldots, T-1$ y $1 \leq j \leq N$. Vamos aplicando la técnica por cada punto, reduciendo los posibles destinos más probables.

3. Terminación:
$$q_t^* = \arg\max_{1\leq i\leq N} \delta_T(i) \tag{3.13}$$
Para nuestro algoritmo, por cada nuevo punto soporte, tendremos una predicción posible.

4. Reconstrucción de la secuencia hacia atrás:
$$q_t^* = \psi_{t+1}(q_{t+1}^*),\ t = T-1, T-2, \ldots, 1 \tag{3.14}$$

3.4. Resumen

La elección de los puntos anteriores y posteriores a los cruces como puntos soporte u observaciones proporciona una importante información sobre los recorridos

que se están realizando a la vez que permite una drástica reducción de puntos a incluir en el modelo probabilístico. Estas características junto a la independencia de cualquier Sistema de Información Geográfico y su posibilidad de realizar predicciones en cualquier zona que frecuente el usuario, hacen que este modelo probabilístico sea prometedor con respecto a los trabajos anteriores.

En el Capítulo 5, veremos los resultados obtenidos con este modelo sobre conjuntos de datos reales.

CAPÍTULO 4

SIMILITUD DE RECORRIDOS

En este capítulo realizaremos un estudio sobre diferentes medidas de similitud y distancias entre recorridos grabados mediante receptores GPS. Para ello revisaremos las aproximaciones proporcionadas por otros autores y a continuación propondremos una nueva alternativa.

Los objetivos perseguidos con este estudio son básicamente los dos siguientes:

- La clasificación automática de recorridos GPS finalizados, es decir, dada una base de datos de recorridos finalizados y un recorrido nuevo finalizado, encontrar el recorrido que más se parece al nuevo. Esta clasificación nos permitirá agrupar los recorridos según el camino que realicen, es decir, tendremos un cluster de recorridos finalizados.

- La clasificación automática de recorridos GPS no finalizados, es decir, dada una base de datos de recorridos finalizados y un recorrido parcial NO finalizado, encontrar el recorrido que más se parece al nuevo. Esta clasificación temprana nos permitirá seleccionar el destino con la mayor evidencia de ser el destino final.

Un ejemplo aclaratorio de estas cuestiones podemos verlo gráficamente representado en la Figura 4.1.

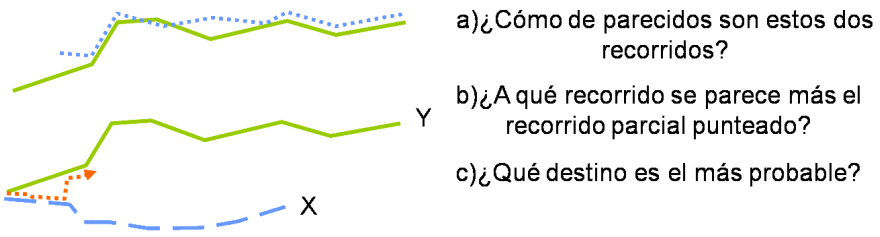

Figura 4.1: Utilidad de tener una similitud entre recorridos.

Para conseguir estos objetivos, comenzaremos analizando las variables de los recorridos que podemos usar, a continuación estudiaremos diferentes medidas de similitud propuestas por otros autores y finalmente propondremos nuevas medidas, las cuales son más apropiadas para nuestra aproximación.

4.1. Tipos de variables a comparar

Para trabajar con una medida de similitud hemos de tener en cuenta que ésta depende del tipo de representación de los objetos a comparar así como de las características de éstos. Por ello en primer lugar analizaremos las componentes de cada recorrido y después justificaremos las elegidas para nuestros objetivos.

Recordamos que cada recorrido del conjunto $RecorridosFiltrados$ se componía de puntos GPS filtrados y éstos a su vez de diferentes atributos:

$b_i =< lat_i, lon_i, alt_i, vel_i, rumbo_i, tiempo_i, HDOP_i, VDOP_i, PDOP_i >.$

Con objeto de disminuir el número de componentes del vector n-dimensional, y utilizar aquellas que nos sean útiles en los posteriores desarrollos es necesario hacer una selección de atributos que nos permita alcanzar nuestros objetivos de forma más eficiente. Para esta selección es primordial destacar que en primer lugar nos importa

el destino que vamos a alcanzar y en segundo el camino que seguiremos. Como podemos observar, realmente nos interesa la componente espacial del recorrido. Aún así cabría plantearse analizar la componente temporal, es decir el instante en que cada uno de los puntos ha sido obtenido. Sin embargo, esa componente temporal puede ser analizada como atributos del propio recorrido de manera resumida como tiempo de inicio o de fin, ahorrándonos mucha información a comparar.

Dado que nuestro interés se centra en el espacio y no en el tiempo, vamos a seleccionar únicamente las componentes de latitud y longitud (la altura la obviamos dado que sería difícil hacer un mismo recorrido a diferentes alturas aunque usásemos un transporte aéreo). Como cada punto b_i del recorrido ha sido obtenido en el instante $tiempo_i$ y la secuencia se ha obtenido en orden temporal, es decir, $tiempo_i < tiempo_{i+1}$, podemos considerar los recorridos como vectores ordenados en los que cada punto está indexado por su orden de obtención. De ese modo aunque en la metodología hemos definido los puntos de los recorridos con todas sus componentes, para obtener una medida de similitud entre ellos, trabajaremos sólo con las componentes de latitud y longitud. Para simplificar la notación utilizada en este capítulo, notaremos los recorridos como $R = \{R_1, \ldots, R_n\}$ de los que sólo consideramos sus componentes espaciales.

4.2. Estado del arte

Tal y como expone Lin [Lin98] para poder dar una definición formal al concepto intuitivo de la similitud, en primer lugar debemos describir nuestras intuiciones sobre ésta. Así:

1. La similitud entre dos objetos A y B se relaciona con sus elementos en común. Cuantos más componentes compartan A y B, más similares.

2. También se relaciona con sus diferencias. Cuanto más difieran A y B, menos similares.

3. Se obtendrá la máxima similitud cuando A y B sean idénticos.

La similitud es por tanto una cantidad que refleja la relación entre dos objetos, i.e. si λ denota el conjunto de objetos entonces una similitud S es una aplicación $S : \lambda \times \lambda \rightarrow \mathbb{R}$. A partir de estas intuiciones, es posible definir múltiples medidas de similitud dependiendo de la interpretación que se le dé a cada una de ellas. Así, por ejemplo en el trabajo de Cha [Cha07] podemos ver una clasificación de hasta 8 diferentes familias de distancias/similitud. Referente a nuestro trabajo, los objetos sobre los que se defina la similitud, deben ser recorridos.

Aún habiendo descartado en el apartado anterior la componente temporal, cabe destacar el trabajo de Michail Vlachos [VKG02] que plantea una interesante medida de similitud basada en el uso del modelo de la subsecuencia común más larga (LCSS) para series temporales de varias dimensiones espaciales. Además se compara con otras propuestas como métricas euclídeas o el Alineamiento Dinámico Temporal (DTW). La medida propuesta, define umbrales de tiempo y espacio que permiten adaptar mejor las series y además incluye funciones de traslaciones para buscar patrones de comportamiento paralelos en espacio y tiempo. La propuesta de Vlachos es genérica por lo que permite comparar series como los trazos generados al escribir una palabra o la grabación de vídeo de los gestos de las manos para expresarse en el lenguaje de los signos. Sin embargo, dado que nuestros datos sólo son de recorridos, definiremos similitudes y distancias adaptadas a nuestro ámbito, sin incluir la variable temporal.

Debemos tener en cuenta que para calcular las similitudes entre recorridos, es necesario medir distancias entre puntos de los mismos. Si consideramos los recorridos en un plano, la distancia entre dos puntos $x = (x_1, \ldots, x_n)$ e $y = (y_1, \ldots, y_n)$ puede ser cualquier variante de la distancia Minkowski o norma L_p:

$$d_{MK}^{P}(x, y) = \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p} \tag{4.1}$$

siendo n el número de dimensiones del punto (en nuestro caso $n = 2$ por tener únicamente las componentes de latitud y longitud) y p el valor de la norma. De

ese modo, $p = 1$ indica la distancia Manhattan, $p = 2$ la Euclídea, $p = +\infty$ la de Chebyshev (con $d = 2$ también llamada distancia del ajedrez), etc..

Sin embargo, en la práctica, tenemos que calcular distancias en una geometría esférica. Aunque existen diversas soluciones que calculan la distancia geodésica entre dos puntos de una elipsoide, la fórmula que normalmente se utiliza es la de Haverseno que permite conocer la distancia ortodrómica[(1)] a partir de las latitudes y longitudes de dos puntos $x = (x_{lat}, x_{lon})$ e $y = (y_{lat}, y_{lon})$:

$$d_{Hav}(x, y) = 2R \arcsin\left(\sqrt{\sin^2\left(\frac{y_{lat} - x_{lat}}{2}\right) + \cos\phi_i \cos\phi_f \sin^2\left(\frac{y_{lon} - x_{lon}}{2}\right)}\right), \tag{4.2}$$

siendo R el radio de curvatura de la tierra en la zona.

También es posible utilizar la fórmula de Vincenty [Vin75] que permite obtener una precisión mayor aunque al tratarse de un algoritmo iterativo, resulta más costoso que la de Haverseno. Sin embargo, una exactitud en el rango de centímetros no tiene sentido cuando nuestro sistema GPS produce errores del rango de 5-10 metros.

Pasamos a continuación a analizar diversas medidas de similitud propuestas en el estado del arte referidas a recorridos. Para ello las clasificaremos según la representación que se le dé a un punto de dicho recorrido.

4.2.1. Punto como símbolo de alfabeto finito.

En los trabajos que destacamos, un punto de un recorrido es representado por un símbolo que forma parte de un conjunto de símbolos finito. Esos símbolos pueden ser una letra o una secuencia de caracteres, pero lo fundamental es que dos puntos son iguales si los símbolos que los representan lo son también.

Laasonen [Laa05] realiza comparaciones de recorridos utilizando la red GSM. Los usuarios del estudio integraron en sus móviles un software que guardaba el identificador de la estación base, equivalente a un punto del recorrido, cuando realizaban

[(1)] La distancia ortodrómica es el arco del círculo máximo que une dos puntos.

una transición entre celdas GSM. De ese modo, un recorrido es una secuencia de identificadores de estaciones base. Para comparar recorridos, utilizan técnicas de de extracción textual, emparejando cadenas (los identificadores de las estaciones) y utilizando el algoritmo de la subsecuencia común más larga para definir la similitud:

$$Sim_{Laa}(R,S) = \frac{|LCSS(R,S)|}{|S|} \tag{4.3}$$

Así, como ejemplo, $Sim_{Laa}(\text{"}abcdef\text{"},\text{"}acbdg\text{"}) = \frac{3}{5}$.

Desde una línea de investigación totalmente diferente, Ningning Hu [HS06] desarrolla una medida de similitud de rutas en Internet definidas. Las rutas en Internet se definen como una secuencia de nodos unidos por unos enlaces de subida o de bajada por lo que su representación en forma de grafo es directa. Cada nodo o punto del recorrido, es representada por el nombre de la máquina. La medida de similitud que utilizan es el ratio entre el número total de enlaces compartidos y el número total de enlaces de las dos rutas. Extrayendo la idea fundamental para que nos sirva como similitud de referencia:

$$Sim_{Nin}(R,S) = \frac{2 * AristasComunes(R,S)}{TotalAristas(R,S)} \tag{4.4}$$

siendo $AristasComunes(R,S)$ el número de enlaces comunes que tienen las rutas y $TotalAristas(R,S)$ es el número de enlaces totales de ambas rutas, es decir el número de aristas comunes entre el total de aristas.

En la Figura 4.2 podemos ver un ejemplo para dos rutas utilizando la anterior aproximación. En este caso la $Sim_{Nin}(s_1,s_2) = \frac{2*4}{17} = \frac{8}{17}$.

Utilizando técnicas de extracción de información textual, Doherty [DGJS06], transforma los ficheros de coordenadas GPS en texto, consultando un servicio de 7 millones de entradas para asignar a cada par longitud-latitud el nombre de la localización más cercana (normalmente el nombre de una calle). Es decir, un recorrido está compuesto por puntos representados por los nombres de las calles más cercanas. Tras este proceso, aplica un modelo de extracción de texto (BM25). Aunque

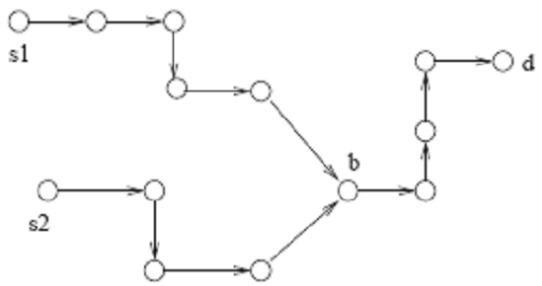

Figura 4.2: Similitud entre rutas de Internet propuesta por Hu.

es un enfoque bastante novedoso, los resultados no son aplicables en la mayoría de casos ya que encuentra segmentos de ruta comunes pero el resto es poco parecido en términos de distancias geográficas.

4.2.2. Punto como variable bidimensional (latitud y longitud).

Las similitudes que normalmente se utilizan entre recorridos representan los puntos de cada uno como una variable con dos componentes continuas, la latitud y la longitud, normalmente influenciados por la tecnología GPS que provee de esos datos de manera precisa. Esta representación nada tiene que ver con la anterior en la que trabajábamos con símbolos en un alfabeto finito, dado que dos puntos sólo serán iguales si sus atributos (latitud y longitud) coinciden. Teniendo en cuenta que estas variables tienen entre 5 y 10 decimales y la precisión del sistema GPS, la igualdad entre dos puntos no suele darse nunca y lo que es más importante, al tratarse de variables cuantitativas son más adecuadas para un tratamiento más ricos en términos de extracción de conocimiento.

Jan [JHP00] utiliza una medida de separación para aquellos recorridos que tienen los puntos iniciales y finales idénticos (considerando un área circular de coincidencia). Crean un índice de desviación de los caminos determinado por el área encerrado entre

los 2 recorridos dividido entre la distancia del camino más corto T (que no tiene porqué ser ninguno de los 2 que se están comparando) al cuadrado:

$$Sep_{Jan}(R, S, T) = \frac{d_{area}(R, S)}{longitud(T)^2} \tag{4.5}$$

El problema de esta medida es la necesidad de conocer el camino más corto T que en su caso era el de menor tiempo recorrido para hallar la desviación entre los recorridos que compartían los inicios y los finales (R y S). Además, para hallar el área encerrada entre ambos recorridos, utilizaban polígonos dibujados a mano donde los lados de los polígonos eran las trazas de los recorridos.

Froehlich y Krumm [FK08]computan la distancia mínima de cada uno de los puntos de un recorrido con los segmentos del otro. Eso les hace tener en cuenta las variaciones en la frecuencia de muestreo del GPS. Calculando esta similitud en ambos recorridos y haciendo la media obtienen la medida de distancia. La propuesta se ilustra en la Figura 4.3. Sean $R = \{R_1, \ldots, R_n\}$ y $S = \{S_1, \ldots, S_m\}$ dos recorridos

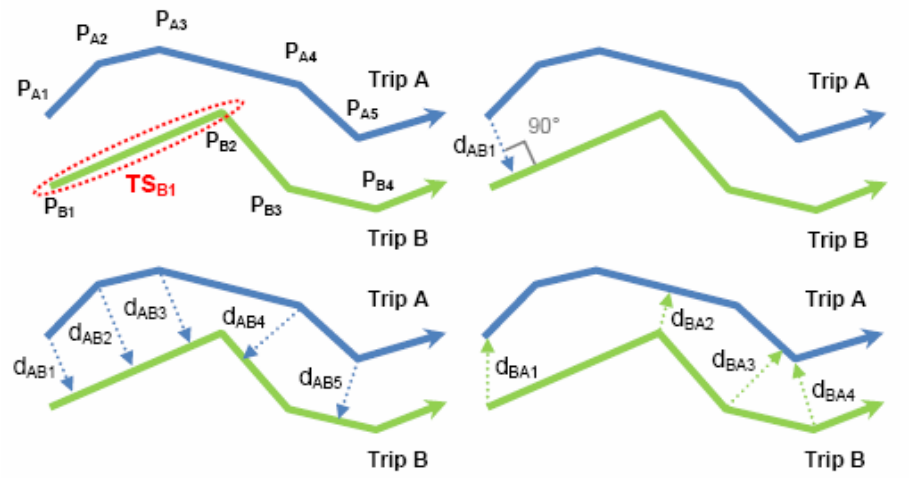

Figura 4.3: Ejemplo de la similitud punto-segmento.

formados por sus puntos GPS, se define la separación Froehlich entre ellos y se denota $Sep_{Fro}(R, S)$ como sigue:

$$Sep_{Fro}(R, S) = \sum_{i=1,\ldots,n} \min_{j=1,\ldots,m-1} d_{Pto-segmento}(R_i, S_{j,j+1}), \tag{4.6}$$

donde $S_{j,j+1}$ es el segmento determinado por los puntos S_j y S_{j+1}. Para calcular la distancia $d_{Pto-segmento}(R_i, S_{j,j+1})$, se halla la distancia Haverseno entre R_i y el punto resultante (S_k) de obtener una perpendicular a la recta generada por el segmento $S_{j,j+1}$ que pase por R_i. Si S_k no pertenece a $S_{j,j+1}$, entonces el resultado que se considera es mín$(d_{Hav}(R_i, S_j), d_{Hav}(R_i, S_{j+1}))$. Aunque la distancia es más precisa que cuando se utiliza la de punto a punto, la complejidad es mayor. Además en su estudio no se demuestra que ciertamente es una distancia ya que ciertamente no lo es, al no ser simétrica $Sep(R, S) \neq Sep(S, R)$.

A partir de la anterior separación se define la distancia Froehlich como sigue:

$$D_{Fro}(R, S) = \frac{Sep_{Fro}(R, S) + Sep_{Fro}(S, R)}{2}, \tag{4.7}$$

la cual ya si es simétrica, pero sigue sin demostrarse que es una distancia. Para decidir si dos recorridos son similares o no, Froehlich establece un umbral de unos 80 metros de diferencia, por encima de la cual, ambos recorridos no serán similares. Sin embargo, esta aproximación tiene una clara limitación ya que si consideramos dos recorridos totalmente paralelos entre sí manteniendo una distancia entre ellos de sólo un metro, la distancia de Froehlich será directamente proporcional al tamaño de los recorridos (concretamente a su número de puntos), por lo que si los recorridos tienen más de 80 puntos cada uno, se indicará que éstos no se parecen, mientras que si son más cortos indicará lo contrario.

4.3. Nuestra propuesta

En esta sección plantearemos diferentes medidas de similitud específicas para nuestro problema teniendo en cuenta los trabajos previos estudiados. Comenzaremos definiendo similitudes y separaciones para recorridos finalizados y finalmente propondremos medidas de similitud para comparar recorridos no finalizados con los ya completos.

4.3.1. Aspectos de las medidas de similitud deseadas

Las similitudes que propongamos deben considerar los siguientes aspectos de nuestros datos:

1. **Precisión GPS:** Aunque realizamos el filtrado de outliers en el preprocesado, nos encontraremos que una misma posición puede ser representada por múltiples puntos GPS diferentes dado que el dispositivo receptor tiene una precisión de varios metros.

2. **Independencia del tamaño:** Al comparar 2 recorridos su similitud será independiente del tamaño que tengan. Es decir, dos recorridos que se parezcan visualmente desde una perspectiva aérea deberán tener una similitud elevada tengan 100 metros de longitud o 100 kilómetros.

3. **Caminos de ida y vuelta:** En muchos casos, los recorridos de ida son prácticamente iguales a los recorridos de vuelta, cambiando el sentido de la marcha y existiendo un pequeño desplazamiento espacial dado que se realizan normalmente por diferentes carriles. Permitir una alta similitud entre caminos de ida y vuelta, permitirá reducir los datos almacenados.

4. **Diferentes velocidades:** Dos recorridos realizados por el mismo camino uno en bicicleta y otro en coche, deben tener una similitud elevada aunque sus velocidades sean completamente diferentes.

5. **Pequeñas variaciones de camino:** Dos recorridos deben parecerse si comparten gran parte del trayecto aunque durante una parte del mismo utilicen diferentes itinerarios.

4.3.2. Distancia Hausdorff-GPS

Desde el ámbito de visión por computador [DR93, Ata83, Sho89, Sho91], la distancia Hausdorff es la más utilizada para medir las diferencias entre dos polígonos.

Dado que los recorridos GPS pueden verse como líneas poligonales abiertas, se puede aplicar esta distancia. Un ejemplo de ello puede verse en el trabajo de Morris [MB08] en el que utiliza la distancia Hausdorff para determinar la similitud entre recorridos con inicio y fin definidos y ver si sus algoritmos de detección de caminos rurales a partir de imágenes aéreas y de satélite eran o no correctos.

Definición 13 *La separación de Hausdorff entre un conjunto de puntos* $R = \{R_1, \ldots, R_n\}$ *y* $S = \{S_1, \ldots, S_m\}$ *se define como la máxima de las distancias mínimas entre sus puntos y notamos como sigue:*

$$Sep_{Hff}(R, S) = \max_{i=1,\ldots,n} \min_{j=1,\ldots,m} d^P_{MK}(R_i, S_j). \tag{4.8}$$

Dado que no se trata de una función simétrica, suele darse una definición de distancia:

$$D_{Hff} = \max(Sep_{Hff}(R, S), Sep_{Hff}(S, R)). \tag{4.9}$$

Si los conjuntos en vez de puntos son líneas o polígonos, entonces d_{Hff} se aplica a todos los puntos de cada una de las líneas o polígonos, no solo los vértices, por lo que tenemos infinitos puntos en vez de un conjunto finito. Uno de los algoritmos que suele usarse para solventar ese problema se describe en el trabajo de Atallah [Ata83].

En nuestro trabajo [AGOV07] propusimos una variación de esta distancia:

Definición 14 *Dados dos recorridos* R *y* S*, se define la separación Hausdorff-GPS entre ellos y se denota por* $Sep_{H-GPS}(R, S)$*, como sigue:*

$$Sep_{H-GPS}(R, S) = \max_{i=1,\cdots,n} \min_{j=1,\cdots,m} d_{Hav}(R_i, S_j). \tag{4.10}$$

Lo que significa aplicar la medida de diferencia de Hausdorff entre conjuntos (de ese modo se evita trabajar con infinitos puntos) en geoides utilizando la distancia Haverseno en vez de la de Minkowski. La medida de similitud busca en el conjunto

de las mejores correspondencias entre los puntos de los dos recorridos, quedándose con el máximo de ellas.

Definición 15 *Dados dos caminos R y S, se define la distancia Hausdorff-GPS entre ellos, y se denota por $D_{H-GPS}(R,S)$, como sigue:*

$$D_{H-GPS}(R,S) = \frac{Sep_{H-GPS}(R,S) + Sep_{H-GPS}(S,R)}{2}. \tag{4.11}$$

Las propiedades fundamentales de esta separación son:

1. $D_{H-GPS}(R,S) \geq 0$
2. $D_{H-GPS}(R,S) = D_{H-GPS}(S,R)$
3. $D_{H-GPS}(R,S) = 0 \Leftrightarrow R = S$ dado que para cualquier R_i existe S_j tal que $R_i = S_j$ y para cualquier S_j existe R_i tal que $S_j = R_i$.
4. $D_{H-GPS}(R,S)$ verifica la desigualdad triangular.

De las cuatro propiedades se sigue que $D_{H-GPS}(R,S)$ es una medida de distancia. Este resultado es remarcable ya que las mayorías de las similitudes definidas en los anteriores aproximaciones no sustentas sus afirmaciones en una base matemática tan sólida como la nuestra.

Definición 16 *Dados dos caminos R y S, se define la similitud Hausdorff-GPS entre ellos, y se denota por $Sim_{H-GPS}(R,S)$, como sigue:*

$$Sim_{H-GPS}(R,S) = \frac{1}{1 + D_{H-GPS}(R,S)} \tag{4.12}$$

Debemos indicar que esta similitud está acotada entre 0 (mínima similitud) y 1 (máxima similitud).

Unos meses más tarde de nuestra propuesta, en septiembre de 2007 [TZL+07], el investigador Torkkola del Laboratorio de Sistemas Inteligentes de Motorola propuso una medida de separación no simétrica basada en la de Hausdorff muy parecida a

la nuestra: Sean $R = \{R_1, \ldots, R_n\}$ y $S = \{S_1, \ldots, S_m\}$ dos recorridos formados por sus puntos GPS,

$$Sep_{Tor}(R, S) = \max_{i=1,\ldots,n} \min_{j=1,\ldots,m} d(R_i, S_j). \tag{4.13}$$

Cabe destacar que no especifican el tipo de distancia utilizada entre puntos, aunque cabría suponer que es la Haverseno. Aún así, no justifican formalmente una distancia entre recorridos (no puede serlo al no cumplir la propiedad de simetría). Para decidir si ambos recorridos son similares o no, Torkkola considera que si $Sep_{Tor}(R, S) < \max(R_i, R_{i+1})$ y $Sep_{Tor}(S, R) < \max(S_i, R_{i+1})$ para todo i, entonces se trata de recorridos similares. En otro caso no lo serán.

4.3.3. Similitud Jaccard-GPS

En las similitudes que representan los puntos de un recorrido como símbolos de un alfabeto finito, hemos visto que podían aplicar operaciones como la subsecuencia común máxima, la intersección o la unión. Esto era posible debido a que los puntos podían ser idénticos si sus representaciones lo eran. Sin embargo, en los casos en los que un punto se representa por su latitud y longitud, dos puntos pueden no ser iguales incluso si el instante y el lugar en el que se obtiene la información GPS son idénticos, como ya comentamos anteriomente. Para evitar ese problema, en primer lugar, consideraremos los puntos GPS como regiones bola en vez de simples puntos GPS, en segundo lugar definiremos una equivalencia entre regiones bola y por último definiremos la operación de intersección entre recorridos.

4.3.3.1. Puntos como regiones bola

Sea x_i el punto GPS definido por su latitud y longitud. Consideraremos la región bola que lo representa como

$$B^{\delta}(x_i) = \left\{x \in \mathbb{R}^2 : d(x, x_i) < \delta\right\},$$

siendo $d(\cdot, \cdot)$ una distancia y $\delta > 0$.

A partir de ahora, podremos considerar los recorridos GPS como secuencias de regiones bola: Sea S^δ una secuencia de datos GPS:

$$S^\delta = \left\{B^\delta(x_1), B^\delta(x_2), \cdots, B^\delta(x_n)\right\}.$$

Usaremos a partir de ahora la notación simplificada:

$$S = \{S_1, S_2, \cdots, S_n\}.$$

Una vez hecha esta conversión, conseguimos solucionar el problema de la precisión del sistema GPS: nuestros datos ya no serán puntos sino regiones circulares centradas en ese punto concreto.

Ahora estamos en disposición de definir la operación de equivalencia entre regiones bola:

Definición 17 (α-equivalencia entre regiones bola) *Decimos que la región bola $B^\delta(x)$ es α-equivalente a la región bola $B^\delta(y)$ y notamos como $x \equiv^\alpha y$ si se satisface que $d_{Hav}(x, y) \leq \alpha$ donde $\alpha > 0$.*

En realidad no es propiamente una equivalencia ya que no verifica la transitividad ($x \equiv^\alpha$ e $y \equiv^\alpha z \nRightarrow x \equiv^\alpha z$). Sin embargo, si verifica una "cuasi"-transitividad ya que si $x \equiv^\alpha$ e $y \equiv^\alpha z \nRightarrow x \equiv^{2\alpha} z$

En la Figura 4.4 si escogemos $\alpha = 2\delta$ podemos considerar α-equivalentes todas los pares de regiones bola del recorrido A que cuyas áreas intersecten con las del recorrido B. De ese modo B_3 sería equivalente a A_2 y a A_3. La $\alpha-$intersección puede verse como aquellas regiones bola de un recorrido cuyas áreas intersectan con regiones bola del otro recorrido.

A partir de la anterior definición, definimos la operación de equivalencia entre recorridos:

Definición 18 (α-equivalencia entre recorridos) *Dos recorridos*

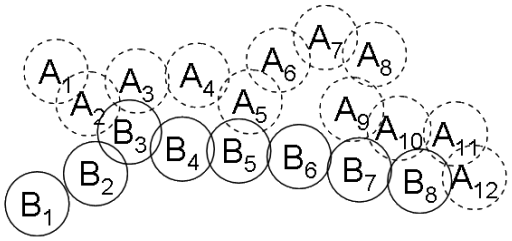

Figura 4.4: Ejemplo de dos recorridos GPS.

$R = \{ R_1, \ldots, R_n\}$ y $S = \{S_1, \ldots, S_n\}$ son α-equivalentes y notamos como $R \equiv_\alpha S$ si se satisface que $\forall R_i \, \exists S_j$ tal que $R_i \equiv^\alpha S_j$ y $\forall S_i \, \exists R_j$ tal que $S_i \equiv^\alpha R_j$.

Por último, definimos la operación intersección entre recorridos:

Definición 19 (α-intersección) *Sea $R = \{R_1, \ldots, R_n\}$ y $S = \{S_1, \ldots, S_m\}$ dos recorridos GPS. Definimos la α-intersección y notamos como $R \cap^\alpha S$ al conjunto de puntos que pertenecen a uno de los recorridos y son α-equivalentes a alguno del otro:*

$$\{R_i : \exists S_k, R_i \equiv^\alpha S_k\} \cup \{S_j : \exists R_k, S_j \equiv^\alpha R_k\}.$$

En la Figura 4.4 vemos como

$$A \cap^\alpha B = \{A_2, A_3, A_5, A_9, A_{10}, A_{11}, A_{12}, B_3, B_5, B_7, B_8\}.$$

4.3.3.2. Definición de similitud

A continuación generaremos una medida de similitud basada en los elementos comunes y diferentes tal y como hemos visto en las intuiciones de Lin. Tras diversas aproximaciones, decidimos utilizar el coeficiente de similitud de Jaccard o índice Jaccard para variables no binarias:

Definición 20 (α-similitud Jaccard) *Denominamos α-similitud Jaccard y notamos como $SimJ_\alpha(R, S)$ al conjunto número de regiones bola que intersectan entre R y S dividido entre el número total de regiones bola de ambos recorridos:*

$$SimJ_\alpha(R,S) = \frac{|R \cap^\alpha S|}{|R \cup S|}. \tag{4.14}$$

En la Figura 4.4 podemos observar cómo $SimJ_\alpha(A,B) = \frac{11}{20}$.

Las propiedades fundamentales de esta similitud son:

1. $0 \leq SimJ_\alpha(R,S) \leq 1$
2. $SimJ_\alpha(R,S) = SimJ_\alpha(S,R)$
3. $SimJ_\alpha(R,S) = 1 \Leftrightarrow R \equiv^\alpha S$.

Además, esta similitud permite comparar correctamente recorridos de diferentes tamaños, la similitud no depende de la orientación de los recorridos: un recorrido de ida y otro de vuelta tienen una similitud elevada y además recorridos realizados por el mismo camino a diferente velocidad también son parecidos, es decir, satisface los requerimientos que destacamos en la Sección 4.3.1.

A partir de la similitud anterior podemos definir separaciones entre recorridos:

Definición 21 (α-separación Jaccard) *Dados dos recorridos R y S, definimos la separación entre ellos como:*

$$SepJ_\alpha(R,S) = 1 - \frac{|R \cap^\alpha S|}{|R \cup S|}. \tag{4.15}$$

Hacemos notar que denominamos separación y no distancia ya que no tenemos probada la desigualdad triangular.

4.3.4. Similitud por áreas

En esta sección se considera una medida clásica de similitud entre trayectorias (área definida entre dos funciones). Así, se supone que un camino está definido por puntos en vez de bolas, i.e.

$$R_i^\delta = \{A = x_{i1}, x_{i2}, \cdots, x_{in_i} = B\}.$$

Sea $R(AB)$ el conjunto de todos los recorridos que salen de A y llegan a B. Entonces dados dos caminos $R, S \epsilon R(AB)$, definen un polígono cerrado a partir de las dos poligonales obtenidas uniendo los puntos de cada camino (ver Figura 4.5).

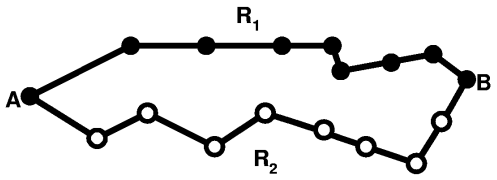

Figura 4.5: Ejemplo de dos caminos como un polígono.

De esta forma se da la siguiente definición:

Definición 22 (Equivalencia) *Dos caminos son equivalentes y se denota por* $R \equiv S$ *si definen el mismo polígono i.e. ambos caminos comparten los mismos puntos (igualdad entre conjuntos), y/o los puntos de cada recorrido definen las mismas poligonales.*

Claramente esta relación es una relación de equivalencia.

Definición 23 (Distancia y Similitud) *Dados dos caminos* R *y* S*, se define una distancia entre ellos, y se denota por* $darea(R, S)$*, como el área encerrada entre* R *y* S *(ver Figura 4.6).*

Se define también la similitud entre R *y* S*, y se denota por* $\Delta(R, S)$*, como sigue:*

$$\Delta(R, S) = \frac{1}{1 + darea(R, S)}. \tag{4.16}$$

Ciertamente, *darea* es una distancia dado que para cualquier $R, S \epsilon R(AB)$ se tiene que:

1. $darea(R, S) = 0$, $(\Delta(R, S) = 1) \Leftrightarrow R \equiv S$.
2. $darea(R, S) = darea(S, R)$, $(\Delta(R, S) = (\Delta(S, R))$.

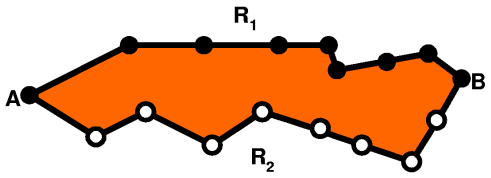

Figura 4.6: Distancia entre dos caminos como un área.

3. $darea(R, S) \geq 0$, $(0 \leq \Delta(R, S) \leq 1)$.

4. $darea(R, S) \leq darea(R, T) + darea(T, S)$ para cualquier $R, S, T \in R(AB)$.

Esto es claro sin más que pensar que se ha identificado cada camino con una trayectoria (se puede pensar en el plano o en el espacio) o una función. Por tanto $R = f_1(x)$ y $S = f_2(x)$ y se ha definido $darea(S, R) = \int_A^B \|f_1(x) - f_2(x)\|dx$.

El principal inconveniente que presenta en la práctica esta aproximación se encuentra cuando se pretende aplicar este tipo de distancia a dos caminos que aún empezando y finalizando en los mismos lugares, los puntos iniciales y finales validados no coinciden, con lo cual no se obtendría un polígono (las poligonales no se cierran). Sin embargo, esta situación puede ser salvada si previamente se decide, por convenio, tomar los lugares más frecuentados en los caminos como marcas de posiciones y de esta forma se cierra el polígono con la línea que une el anterior punto con el primer punto validado del camino.

4.3.5. Análisis de las diferentes medidas de similitud

Para poder analizar la adaptación de las diferentes similitudes/disimilitudes a nuestro problema, generamos la Tabla 4.1 en la que pueden verse las propuestas más destacadas con respecto a las características a analizar: precisión GPS, independencia del tamaño de los recorridos, recorridos de ida y vuelta, diferencia de velocidades y pequeñas desviaciones.

Tabla 4.1: Comparación de medidas de similitud entre recorridos

Característica	Froehlich	Hausdorff-GPS	Área	Jaccard-GPS
Precisión	No	Sí	No	Sí
Tamaño	No	Sí	No	Sí
Ida y vuelta	Sí	Sí	No	Sí
Velocidades	Sí	Sí	Sí	Sí
Desviaciones	No	No	No	Sí

Podemos observar que tanto en las medidas de Hausdorff-GPS, Froehlich y Área, no disponemos de una normalización, es decir, la medida depende de la magnitud de los recorridos comparados y no será sencillo escoger un umbral por encima del cuál decidir si ambos recorridos son parecidos o no. Imaginemos dos recorridos como los de la Figura 4.7. En ellos podemos intuir una similitud elevada, sin embargo, si los recorridos tienen muchos puntos o son muy largos, las similitudes de Froehlich y del área generarán valores que indicarán poca similitud o mucha distancia. En la medida propuesta por Torkkola o la nuestra de Hausdorff-GPS, dependerá de las distancias máximas entre puntos, por lo que si nos hemos desviado brevemente de nuestro recorrido, se alcanzarán esos máximos y se asumirá una disimilitud total. Sin embargo, con la propuesta de Jaccard-GPS, prácticamente todos los puntos están a una distancia suficiente como para considerarlos equivalentes y por tanto nuestra medida indicará una similitud muy elevada, es decir es robusta frente a pequeñas desviaciones mientras que las demás no lo son.

4.3.6. Similitud entre recorridos no finalizados

Las similitudes propuestas hasta ahora consideran que todos los recorridos han finalizado y su objetivo era el de clasificar las diferentes rutas. Sin embargo, ahora nuestro objetivo varía y lo que pretendemos es realizar predicciones. Para ello necesitamos comparar recorridos que no han acabado con aquellos que sí lo han hecho. Al contrario que en el anterior apartado, no existe un estudio sobre este tipo de

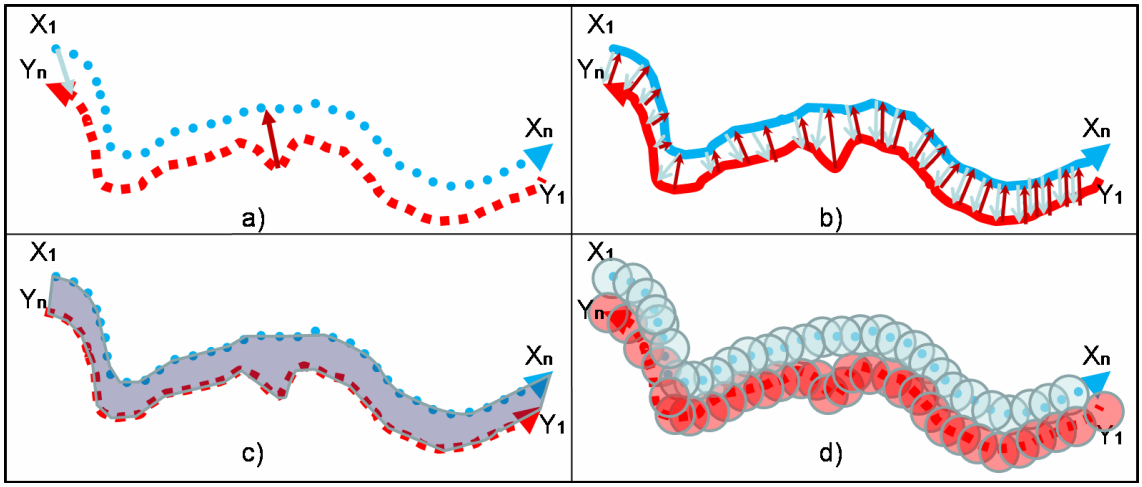

Figura 4.7: Ejemplo de dos recorridos comparados con las diferentes disimilitudes estudiadas: a) Hausdorff-GPS, b) Froehlich, c) áreas y d) Jaccard-GPS.

similitudes en el estado del arte por lo que plantearemos una medida de similitud que recoja las siguientes características:

Últimos datos más importantes. Es lógico considerar que la información más importante es la última recibida, dado que son los puntos GPS más próximos al destino.

Recorridos no clasificados. En algunos casos se realizan trayectos que provienen de lugares no indexados pero cuyos destinos sí lo son. Este tipo de trayectos, se suelen adaptar a segmentos de rutas conocidas para llegar al destino. Desde el punto de vista de un conductor o de un transeúnte es lógico que se aproveche el conocimiento de sus rutas pasadas para realizar una composición de éstas y llegar a un destino conocido.

Tiempo de fijado. Como ya hemos expuesto, el tiempo de fijado de los satélites por parte del receptor GPS puede ser de pocos segundos, si estamos en condiciones ideales, a varios minutos en caso contrario. En los datos reales detectamos que muchas veces el portador del receptor GPS no esperaba a tener fijados los satélites para comenzar su trayecto por lo que el primer punto vali-

dado distaba cientos de metros del origen real. Este aspecto práctico tendremos que tenerlo en cuenta para evitar problemas en nuestras similitudes.

Con el fin de alcanzar este objetivo, definiremos una medida con ponderaciones para considerar que la información más importante es la última recibida y una medida de similitud que la utilice.

Para conseguir ponderar los últimos puntos por encima de los primeros, definimos la función $Orden_\alpha$:

Definición 24 *Dado el recorrido sin finalizar* $R = \{R_1, R_2, \ldots, R_m\}$ *y el recorrido finalizado* $S = \{S_1, S_2, \ldots, S_n\}$*, definimos la función*

$$Orden_\alpha(R_i, S) = \begin{cases} i & si\ \exists\ S_j\ tal\ que\ R_i \equiv^\alpha S_j \\ 0 & e.o.c \end{cases}$$

A partir de esa función, podemos obtener la siguiente medida de similitud:

Definición 25 *Definimos la* α*-similitud deslizante entre* R *y* S *y notamos como* $SimD_\alpha(R, S)$*:*

$$SimD_\alpha(R, S) = \sum_{i=1}^{m} \lambda_i\, Orden_\alpha(R_i, S)$$

donde λ_i *son las ponderaciones.*

En la Figura 4.8 podemos ver dos rutas X e Y y cuatro regiones bola de un recorrido parcial Z. Obsérvese que Z_3 no es equivalente a ninguna región bola de Y y Z_4 no lo es con ninguna de X. Para esclarecer la representación hemos obviado las regiones bola de los recorridos finalizados. En este caso, el número de regiones bola de Z que intersectan con X son las mismas que el que intersectan con Y, que son 3. Sin embargo, $SimD_\alpha(Z, X) = 1+2+3+0 = 6$ y $SimD_\alpha(Z, Y) = 1+2+0+4 = 7$. En este ejemplo se ha considerado $\lambda_i = 1, \forall i$

Podemos observar que no existe simetría, aunque tampoco es algo importante ya que compararemos un recorrido parcial con todos los finalizados y no al revés.

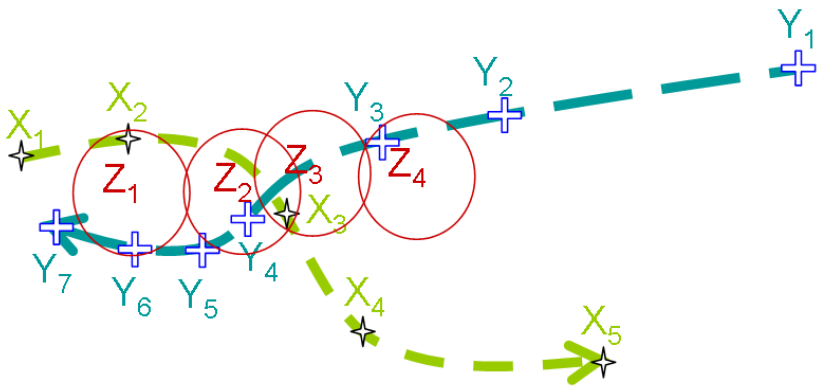

Figura 4.8: Ejemplo de ruta más cercana deslizante.

Si queremos conseguir una medida de similitud como una media aritmética ponderada (combinación convexa) hemos de conseguir que $\lambda_i \geq 0$ y $\sum_{i=1}^{m}\lambda_i = 1$. Para ello podemos hacer $\lambda_i = \frac{i}{\sum_{i=1}^{m} i}$ y como $\sum_{i=1}^{m} i = \frac{(m+1)m}{2}$, tenemos:

$$SimD_\alpha(R,S) = \frac{1}{\sum_{i=1}^{m} i}\sum_{i=1}^{m} i\, Orden_\alpha(R_i,S)$$

Con esta medida, damos más importancia cuanto mayor sea i, sin embargo la medida no está acotada. Para subsanar ese inconveniente, buscamos otra aproximación para tener la similitud acotada lo cual facilitará su interpretación. Teniendo en cuenta que $Orden_\alpha(R_i,S) = 0$ o i, se sigue que considerando los pesos constantes e igual a $\frac{2}{m(m+1)}$, se tiene que $0 \leq \sum_{i=1}^{m}\lambda_i Orden_\alpha(R_i,S) = \frac{2}{m(m+1)}\sum_{i=1}^{m} Orden_\alpha(R_i,S) \leq \frac{2}{m(m+1)}\sum_{i=1}^{m} i = 1$. De esta forma consideramos una nueva medida de similitud:

$$SimD^*_\alpha(R,S) = \frac{2}{m(m+1)}\sum_{i=1}^{m} Orden_\alpha(R_i,S).$$

Con esta medida de similitud, conseguimos destacar lo más importante de nuestro recorrido en curso es decir, la parte final que es la que se aproxima al destino, no a partir de los pesos, sino a través de la función $Orden$. Evidentemente los pesos

no suman 1 pero no se pierde la naturaleza de la similitud. Esto es lógico ya que se pueden considerar cambios de recorrido, con lo que rápidamente mejorarán aquellos que se parezcan más y empeorarán los que no se parezcan en los últimos puntos.

En la Figura 4.9 podemos ver el comportamiento de las similitudes considerando sólo el número de regiones bola que intersectan con los demás caminos y la similitud deslizante. En el primer caso aunque el recorrido parece avanzar tal y como lo hace la ruta B, como hay un mayor tramo de recorrido que se asemeja a la ruta punteada, la mayor similitud sigue siendo con la ruta A. Utilizando el algoritmo de similitud deslizante, cuando el recorrido parcial está ya avanzado, la mayor similitud pasa a ser la ruta B dado que se ponderan más los últimos puntos.

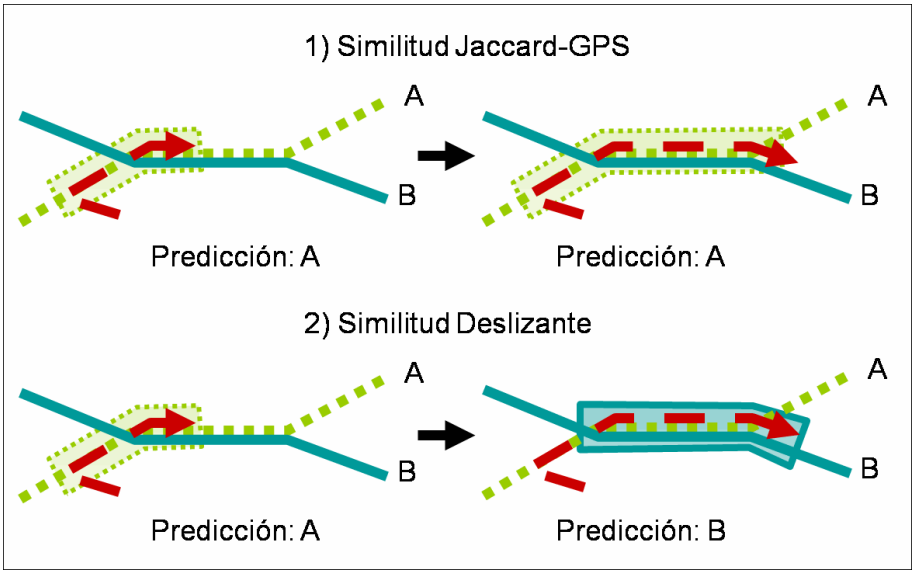

Figura 4.9: Ejemplo de funcionamiento de las similitudes Jaccard-GPS y la deslizante.

4.4. Resumen

En este capítulo hemos estudiado las medidas de similitud existentes para comparar recorridos. Al no existir ninguna que cubriese totalmente nuestros requisitos,

hemos definido tres similitudes (Hausdorff-GPS, Jaccard-GPS y áreas) adaptadas para los recorridos GPS. Tras comparar sus características comprobamos que la que denominamos Jaccard-GPS es la más apropiada para nuestro objetivo por lo que se seleccionó para aplicarla en el cluster de recorridos.

Además, se ha definido una medida de similitud deslizante, útil y necesaria para permitir la comparación de recorridos no finalizados con rutas ya terminadas. Esta similitud es la adecuada para aplicar en predicción debido a que se ponderan los últimos puntos del recorrido parcial en mayor medida que los iniciales, permitiendo determinar la ruta que mejor se adapta y por tanto conocer cuál puede ser su destino más probable.

Una vez analizadas las similitudes entre recorridos GPS, pasamos a analizar los resultados obtenidos tanto de extracción de conocimiento como de predicción en el Capítulo 5.

CAPÍTULO 5

RESULTADOS

En este capítulo mostraremos los resultados obtenidos tras aplicar la metodología propuesta en el Capítulo 2. Comenzaremos describiendo el proceso de recogida de datos y los usuarios que participaron en él. A continuación analizaremos los procesos que la componen y estudiaremos las predicciones obtenidas tanto para la técnica de combinación de HMM junto con puntos soporte como para la de aplicación directa de las similitud deslizante.

5.1. Usuarios estudiados

En primer lugar, estudiaremos a los participantes que nos proporcionaron los datos fuente. El número de usuarios que colaboraron en el estudio fueron ocho. De ellos, seis aportaron datos con más de una semana de recorridos consecutivos. Los otros dos realizaron muy pocos recorridos y en días sueltos.

Observamos que aunque algunos usuarios estuvieron interesados en llevar el GPS para así poder utilizarlo con programas de navegación en vehículos, otros lo consideraban invasivo y poco cómodo y hubiesen preferido integrar el sistema de seguimiento

en sus terminales móviles. Por otra parte, hay que tener en cuenta que la información suministrada invadía la privacidad por lo que podría haber cierta reticencia a colaborar. Entre los datos privados a los que podemos acceder, se encuentran las zonas que más frecuenta, como la vivienda del usuario o posibles tramos de carretera en los que infringió las normas de tráfico rebasando la velocidad permitida mientras conducía.

Deberíamos prestar atención a la legislación vigente[(1)] en la que se indica que quedan prohibidos los ficheros creados con la finalidad exclusiva de almacenar datos de carácter personal que revelen la ideología, afiliación sindical, religión, creencias, origen racial o étnico, o vida sexual. Prácticamente todos esos datos podrían extraerse de un fichero de las localizaciones si cotejamos los datos con bases de datos geo-localizadas o simplemente si conocemos donde se encuentra la sede de un partido político, lugar de culto, etc..

Dado que los usuarios eran conocidos por el autor no hubo problemas en compartir esa información, pero para evitar problemas, se eliminó cualquier rastro del nombre de los voluntarios para que no fuese posible relacionarlos con los ficheros. Además, existen estudios [Kru07] que demuestran la posibilidad de recuperar un alto porcentaje de los nombres y apellidos de los usuarios que participan en procesos de recogida de trazas GPS, simulando un ataque sobre una base de datos con recorridos. Por ello decidimos mantener la base de datos de manera personal y no publicarla para su uso por otros investigadores.

De los seis usuarios con datos suficientes, uno es el autor de este documento y los demás son personas de su entorno cercano. Estas personas tenían diferentes hábitos de desplazamiento tanto por el número de trayectos diarios como por sus horarios, las zonas por las que se desplazaban y los medios de transporte utilizados. Para no revelar sus identidades los nombraremos a partir de ahora con los números del 1 al 6. A continuación los describimos someramente:

- Usuario 1: Se trata de una persona soltera y con empleo. Sus recorridos más

[(1)] LEY ORGÁNICA 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal.

frecuentes son los trayectos de la vivienda de sus padres, con los que vivía, hacia su lugar de trabajo (y el camino inverso). Además suele incluir centros comerciales y la vivienda de su pareja, con la que compartió algunos recorridos en coche (ambos llevaban su receptor encendido por lo que sería posible encontrar los trayectos conjuntos).

- Usuario 2: Comercial casado con empleo. Su patrón de comportamiento es interesante para la predicción de destinos por la variedad de zonas que visita y el número de medios de transporte que utiliza.

- Usuario 3: Trabajador soltero con pareja. Sus destinos además de la casa de su pareja, son el lugar de trabajo y viviendas de familiares a los que visitaba los fines de semana o en vacaciones.

- Usuario 4: Trabajador casado. Sus desplazamientos incluían frecuentes situaciones de atasco entre su vivienda, el trabajo de su mujer a la que llevaba diariamente y su lugar de trabajo. Todos sus desplazamientos son en su vehículo e incluyó alguno a la vivienda de sus padres a más de 100 kilómetros de la suya.

- Usuario 5: Soltero con empleo. Entre sus recorridos se encuentra un periodo de vacaciones y múltiples viajes a diferentes ciudades por lo que resulta de especial interés para el proceso de extracción de destinos.

- Usuario 6: Estudiante universitario con pareja. Sus constantes desplazamientos a pie y con destinos cercanos entre sí supusieron un interesante desafío para el ajuste de parámetros. Entre sus recorridos aparecen los de sus vacaciones de verano.

En la Tabla 5.1 podemos ver algunas características de cada usuario. En ella además de observar el número de días aportados por cada uno, se indican otros datos, como los medios de transporte utilizados y el número de zonas, entendidas como ciudades o pueblos por los que se desplazaron.

Tabla 5.1: Características de los usuarios del estudio.

Usuario	Días	Transporte	Nº de ciudades
1	26	Autobús y coche	2
2	35	Tren, coche, bicicleta y a pie	6
3	37	Coche y bicicleta	8
4	45	Coche	3
5	82	Coche y a pie	13
6	98	Coche y a pie	10

Las ciudades donde se registraron desplazamientos de todos los usuarios fueron en su mayoría Sevilla capital y algunos pueblos cercanos aunque también se obtuvieron, en menor proporción, datos de la provincia de Almería, Cádiz, Granada, Huelva, Málaga, Madrid, Barcelona, Girona, Bilbao y Alicante. En la Figura 5.1 pueden verse a la izquierda los recorridos generados en España (donde se observan fácilmente los outliers como líneas rectas de gran longitud) y a la derecha el detalle de los obtenidos en Sevilla.

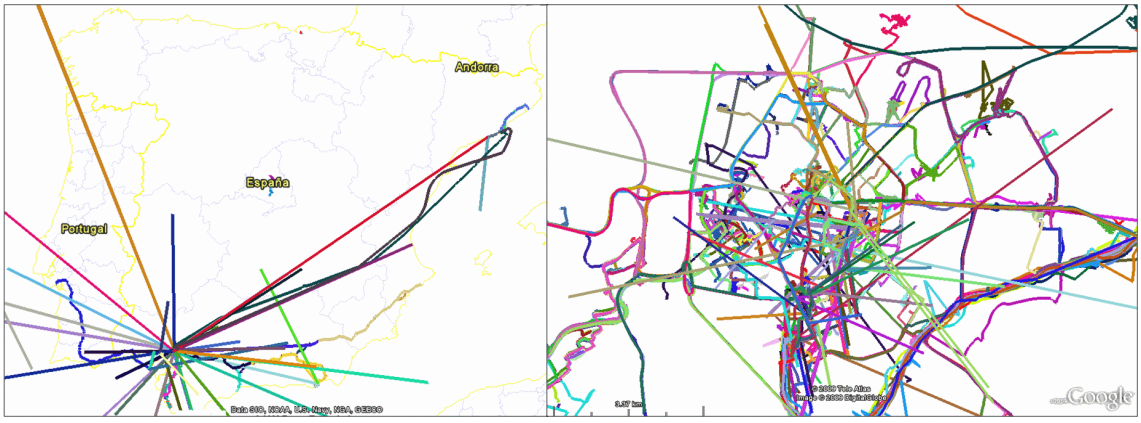

Figura 5.1: Visualización de recorridos obtenidos en España y detalle de los recuperados en Sevilla.

Tras recuperar todos los datos y almacenarlos en una base de datos (SQL Server 2005), obtuvimos 498091 puntos GPS de los ficheros de los seis usuarios. Es impor-

tante indicar que en todos los conjuntos de datos la frecuencia de muestreo se fijó en un Hertzio.

5.2. Procesado

A continuación describimos los resultados obtenidos tras aplicar los procesos de la metodología descritos en el Capítulo 2 sobre nuestro conjunto de datos.

5.2.1. Filtrado

El primer paso de la metodología consiste en filtrar los puntos de cada conjunto de datos, con el objeto de limpiar outliers y reducir el número de datos redundantes que no aportaban información significativa. Los resultados de este procedimiento se presentan en la Tabla 5.2, donde aparecen tres columnas; la primera muestra el número total de puntos para cada usuario, la segunda el número de puntos que restan tras realizar el proceso de filtrado y la tercera el porcentaje de reducción de los datos.

Tabla 5.2: Resultados del proceso de filtrado.

	Puntos		
Usuario	Total	Filtrados	% del Total
1	14884	7871	52.88 %
2	36161	10517	29.08 %
3	161426	57043	35.33 %
4	24099	24011	99.63 %
5	109617	36297	33.11 %
6	143701	61179	42.57 %
Total	498091	196837	39.52 %

Puede observarse como el filtrado de puntos reduce de manera considerable el número de datos. Destaca el bajo porcentaje de reducción del usuario 4. Este hecho se debe a que el conjunto de datos originales se perdió y trabajamos sobre un conjunto que había sido sometido al filtro de distancia (ver Sección 2.2.3). Al no haber podido recuperar los datos iniciales, trabajamos con los datos parcialmente filtrados. Este caso nos sirve para comprobar que el filtro por distancia es el que más datos elimina, mientras que los filtrados por análisis de precisión y velocidad extraen un pequeño porcentaje de outliers.

5.2.2. Segmentado temporal

A partir del conjunto de recorridos obtenidos después del proceso de filtrado, aplicamos el segmentado temporal. Como comentamos en la Sección 2.2.4, esta técnica es usada por diferentes autores con diferentes tiempos, normalmente entre 3 y 10 minutos. Debido a ello nosotros realizamos un estudio del número de candidatos a recorridos obtenidos según variamos su valor desde 1 a 10 minutos. En la Tabla 5.3 y en la Figura 5.2 podemos ver el número de recorridos obtenidos en función del tiempo de segmentado mínimo.

De los resultados obtenidos en la tabla podemos extraer qué tipo de paradas suele realizar cada usuario. Como indicamos en la Sección 2.2.4, se producen falsos positivos si utilizamos un tiempo de segmentado muy bajo (por ejemplo en semáforos) y falsos negativos con tiempos altos (por ejemplos en paradas cortas).

Podemos observar los falsos positivos, por ejemplo, en el usuario 5 en el que el número de candidatos para segmentar los recorridos con valores de 1 minuto es de 1708. Observando espacialmente los puntos candidatos de segmentado, comprobamos que existe una gran concentración de estos puntos en lugares cercanos a los edificios techados. Eso significa que el receptor GPS de este usuario, aún en el interior de un edificio, era capaz de seguir captando puntos válidos. Esto suele ocurrir si el receptor está cerca de una ventana, provocando capturas de puntos validados por el sistema pero aislados temporalmente. Un ejemplo de ello puede verse en la Figura

Tabla 5.3: Número de puntos candidatos para segmentar en recorridos en función del tiempo entre puntos.

	Usuarios					
Minutos	1	2	3	4	5	6
1	308	676	260	417	1708	785
2	112	403	137	161	862	514
3	89	303	124	129	645	462
4	80	256	116	114	550	425
5	77	233	116	102	498	405
6	74	216	115	101	448	392
7	72	205	111	100	421	379
8	72	189	110	99	400	371
9	69	183	107	97	375	363
10	67	177	105	96	363	355

5.3 en la que se generan 18 posibles puntos de segmentado entre las 16:28 y las 17:29 estando el usuario dentro del edificio. Cada punto está etiquetado con su día y hora de obtención. Además, tras la hora pueden verse los segundos que transcurren hasta el siguiente punto.

5.2.3. Extracción de destinos

En este proceso realizamos un cluster jerárquico modificado, a partir de los puntos finales de todos los recorridos, explicado en la sección 2.2.5. Recordamos que este algoritmo de clustering necesita los parámetros *minPtos* que indica el mínimo número de puntos por cluster y *radio* que establece el radio máximo del lugar. Dependiendo del parámetro *radio* elegido tendremos más o menos lugares frecuentes. Podemos ver la evolución del número de lugares finales con respecto al umbral elegido para los diferentes usuarios en la Figura 5.4. Aunque cabría esperar que según aumenta el radio de los destinos el número de éstos disminuyese por la agrupación

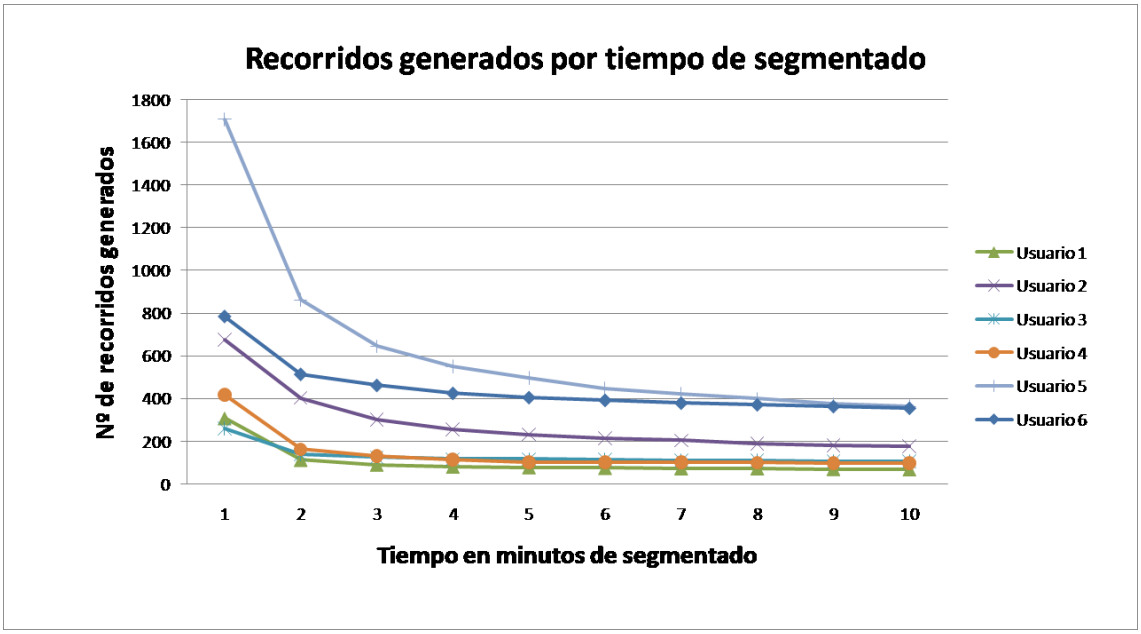

Figura 5.2: Puntos candidatos para segmentar en recorridos en función del tiempo entre puntos.

de varios en uno solo, observamos que no siempre se cumple. En algunos tramos, por ejemplo en los usuarios 1 y 2 entre los 100 y 200 metros, existe un crecimiento debido a que puntos finales con menos de tres elementos en su cluster incluyen un tercer elemento al aumentar el radio.

Al no existir un criterio válido para todos los usuarios en cuanto a la forma de las curvas, se decidió utilizar el criterio de antelación en la predicción de tres minutos que a pie suponían 200 metros.

En la Figura 5.5 podemos observar el dendrograma o representación del árbol binario generado en el cluster para el usuario 1. Marcados con elipses están las agrupaciones de más de tres puntos finales. Los elementos no marcados al no existir una densidad suficiente, no son considerados como elementos de ningún destino.

Figura 5.3: Puntos de segmentado generados en un edificio techado.

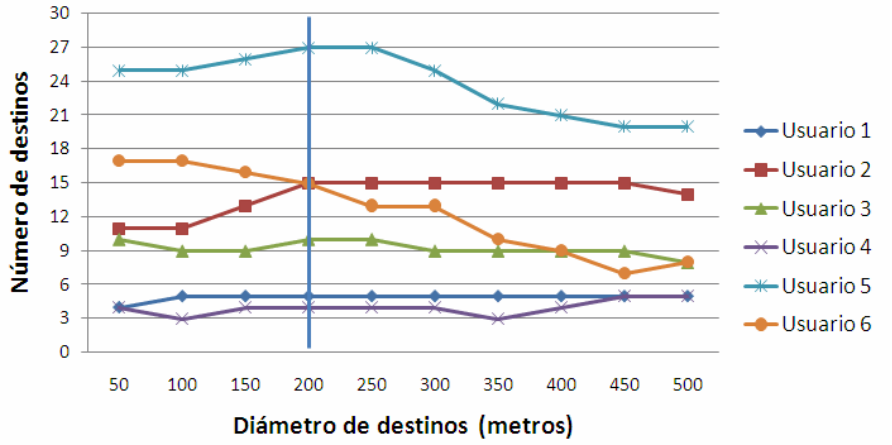

Figura 5.4: Número de destinos generados en función del umbral elegido.

5.2.4. Segmentado por cercanía a destinos conocidos

Una vez realizado el proceso anterior, podemos considerar el algoritmo de segmentado por cercanía a destinos. Recordamos que éste consiste en volver a segmentar los recorridos aplicando un umbral temporal menor en zonas cercanas a los destinos

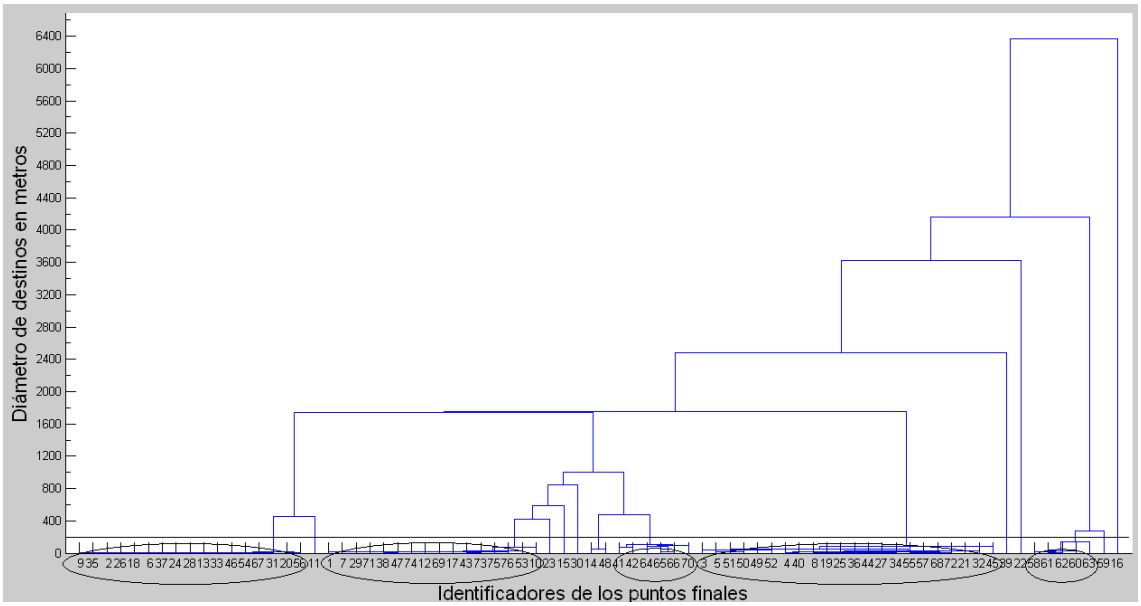

Figura 5.5: Representación del cluster jerárquico modificado aplicado a los puntos finales del usuario 1.

extraídos.

Para ver las mejoras introducidas por nuestra técnica con respecto a la comúnmente usada (segmentado temporal), mostramos los resultados comparativos de ambas en la Tabla 5.4. En ella para cada usuario se muestran el número de recorridos extraídos gracias al segmentado temporal (primera columna), los obtenidos tras aplicar el segmentado por cercanía a destinos (segunda), la diferencia entre éstos (tercera) y el porcentaje comparando ambos (última columna).

Podemos observar cómo el porcentaje de nuevos recorridos gracias al segmentado doble es muy significativo sobre todo en el usuario 4. En su caso se debe al uso común por parte del usuario y su pareja del mismo vehículo, de modo que ambos parten hacia sus respectivos trabajos desde su hogar dejando el primero a su acompañante en su oficina y continuando posteriormente hacia la suya. Si no se realizase el doble segmentado, las predicciones indicarían siempre que el destino es lugar de trabajo del usuario, sin embargo el primer destino es suficientemente significativo como para considerar la parada dado que altera la ruta inicial. Esta situación se repite en los

Tabla 5.4: Diferencia de recorridos generados mediante segmentado temporal y segmentado por cercanía a destinos.

	Segmentado		Nuevos recorridos	
Usuario	temporal	destinos	Diferencia	Porcentaje
1	77	79	2	102.60 %
2	233	236	3	101.29 %
3	116	120	4	103.45 %
4	102	157	55	153.92 %
5	498	589	91	118.27 %
6	405	445	40	109.88 %
Total	1431	1626	195	113.63 %

usuarios 5 y 6 aunque con menor intensidad.

Este resultado es significativo dado que se obtienen hasta un 13.63 % más de recorridos mediante una técnica simple pero novedosa. Esto es importante para aplicaciones de predicción de tráfico: Si predijésemos directamente el destino final, la consulta sobre la ruta hacia ese destino indicaría un camino diferente al que se usa para pasar previamente por un lugar de encuentro antes de dirigirse al final.

5.2.5. Filtrado de recorridos

En este proceso realizamos un filtrado de aquellos recorridos que no nos son útiles para la predicción. En primer lugar eliminamos los que generan un recorrido con inicio y fin en el mismo punto. Entre ellos se incluyen recorridos como paseos o carreras con el mismo origen y destino sin paradas intermedias (en nuestro caso no encontramos este tipo de recorridos) y falsos recorridos, es decir secuencias de puntos que parten y finalizan en el mismo lugar obtenidas al tener encendido el receptor en interiores. Un ejemplo puede verse en la Figura 5.6, donde aparece un falso recorrido por estar el receptor en un edificio techado pero cerca de una ventana.

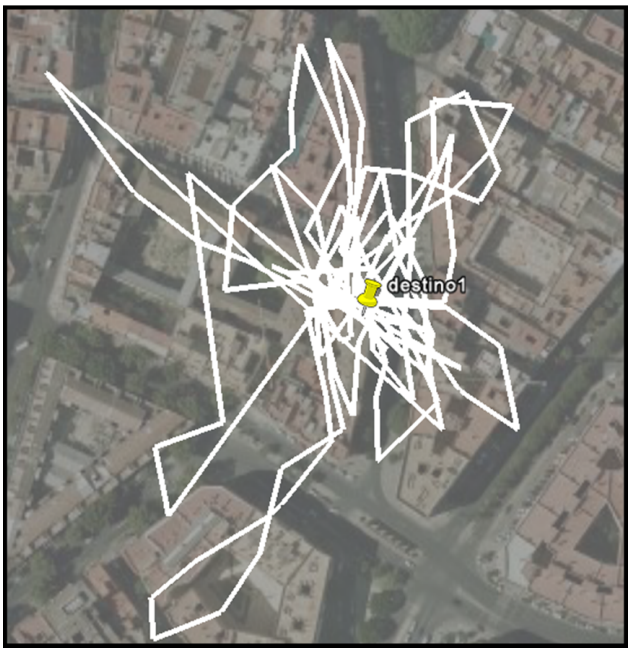

Figura 5.6: Ejemplo de falso recorrido generado al recuperar información desde un lugar techado.

Además, filtramos aquellos recorridos que no tienen sentido a la hora de las predicciones, es decir, aquellos que finalizan lejos de cualquier destino obtenido en el paso anterior. Este último filtrado puede darse en casos en los que la batería del dispositivo se agota o cuando se accede a lugares visitados menos de tres veces. Este último caso supone una pérdida de información con respecto al conjunto inicial de recorridos, pero debemos recordar que el objetivo es predecir un destino frecuente en un desplazamiento y éste se considerara a partir de tres visitas.

Tras filtrar los recorridos obtuvimos los datos que mostramos en la Tabla 5.5. En ella se indica para cada uno de los usuarios el número de recorridos obtenidos tras el doble segmentado, el número de recorridos útiles (columna "Finales"), el porcentaje de recorridos útiles con respecto a los de doble segmentado y los recorridos eliminados por lejanía a destinos extraídos (columna "Distancia") y los eliminados por comenzar y finalizar en el mismo lugar (columna "Inicio y Fin").

Tabla 5.5: Resultados de filtrado de recorridos.

				Filtrado	
Usuario	Recorridos	Finales	Porcentaje	Distancia	Inicio y Fin
1	79	53	67.09 %	11	15
2	236	48	20.34 %	70	118
3	120	84	70.00 %	22	14
4	157	126	80.25 %	20	11
5	589	308	52.29 %	77	204
6	445	305	68.54 %	49	91
Total	1626	924	56.83 %	249	453

Observando la última fila podemos destacar que aunque el porcentaje de recorridos finales es el 56.83 % del total, sólo 249 (un 15.31 %) son por pérdida de información, es decir, recorridos válidos pero descartados por no finalizar cerca de un destino. El mayor número de recorridos filtrados 453, el 27.86 % son falsos recorridos generados desde el interior de edificios.

También podemos observar como para el usuario 2 se pierde un número elevado de recorridos (70, un 29.66 %) por no finalizar cerca de los destinos. Tras analizar dichos trayectos filtrados se pudo observar que los había realizado durante periodos vacacionales (verano y navidades), por lo que había muchos recorridos que finalizaban en lugares que se visitaban menos de tres veces.

Por otra parte observamos que para los usuarios 2, 5 y 6 también existe un filtrado muy alto por recorridos que comienzan y finalizan en el mismo lugar. Esto se explica por su gestión del receptor GPS: una vez llegaban a un destino techado lo mantenían encendido.

5.2.6. Generación de rutas

Tras comprobar que otras distancias no eran aplicables a nuestro problema [JHP00, HS06, DGJS06] o era necesario especificar un umbral dependiente del tamaño medio de los recorridos a comparar [FK08] o no eran suficientemente flexibles para nuestro propósito [TZL+07, ZLTG07], para generar las rutas, utilizamos la separación Jaccard-GPS propuesta en la Sección 4.3.3. Con nuestra alternativa permitimos dos niveles de tolerancia: el primero a nivel de puntos y el segundo a nivel de recorridos.

A nivel de puntos establecimos el umbral, que indica cuando dos regiones bolas son equivalentes, en 40 metros. Escogimos esa distancia al considerar la situación más desfavorable: dos vehículos que viajen en sentidos opuestos a la máxima velocidad permitida, por los carriles más distantes de una autopista. Esa distancia permite que en los tramos en los que ambos sentidos sean paralelos, se generen regiones bolas equivalentes. En la Figura 5.7 podemos observar los cálculos para una distancia entre carriles de autopistas de 20 metros con una frecuencia de muestreo de 1 Hertzio a máxima velocidad y en el caso de instante de muestreo más desfavorable. Como vemos hasta los 40 metros previstos, existe margen suficiente.

A nivel de recorridos, indicamos lo parecido que debían ser dos trayectos para ser considerados como una única ruta. Para permitir una reducción significativa de éstos en rutas escogimos un porcentaje de similitud del 70 % de modo que recorridos que utilizan pequeños desvíos pero comparten gran parte del recorrido siguen perteneciendo a la misma ruta. Además comprobamos que todos los recorridos de un mismo cluster tenían los mismos destinos como extremos.

Para comprobar la corrección de la elección de los parámetros también probamos otros valores. En la Figura 5.8 podemos ver cómo evolucionan el número de rutas generadas según varía el parámetro de equivalencia de separación entre recorridos. Se puede observar que con una similitud del 85 % (15 % de separación) y 100 metros de distancia entre regiones bola equivalentes, obtenemos el mismo número de rutas que para el 70 % de similitud (30 % de separación) y 40 metros.

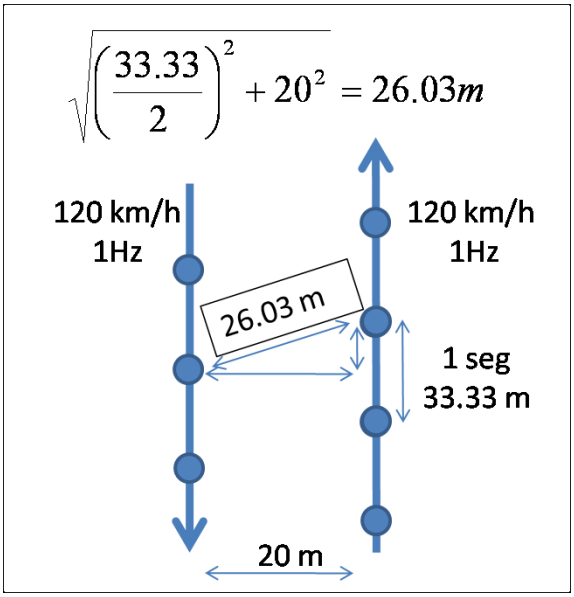

Figura 5.7: Cálculo de distancia umbral para la equivalencia de regiones bola.

En la Tabla 5.6 mostramos el número de rutas obtenidas utilizando una distancia entre regiones bola de 40 metros y realizando agrupación a partir de una similitud superior al 70 %. Además incluimos una columna que indica el porcentaje de reducción. Podemos observar que la reducción de recorridos a sus rutas representantes es muy importante (60.71 % de media), permitiendo resumir la información almacenada y mejorando su integración en dispositivos móviles.

Destacan los resultados extremos de los usuarios 2 y 4. En el primero observamos que no hay una gran reducción de recorridos en rutas mientras que en el segundo podemos agrupar sus recorridos en sólo 17 rutas siendo sólo el 13.49 % del total de recorridos. Para comprobar cómo son sus desplazamientos, comparamos los dendrogramas del usuario 2 y el usuario 1 que tiene un porcentaje de reducción también elevado y un número de recorridos menor que el 4 de modo que su representación gráfica es más clara) y el usuario 2. Podemos verlos en la Figura 5.9 y observamos cómo el primero recorre repetidas veces caminos muy parecidos mientras que el segundo varía bastante más su comportamiento. En el eje de ordenadas se observan los identificadores de cada uno de los recorridos de cada usuario y en el de abscisas

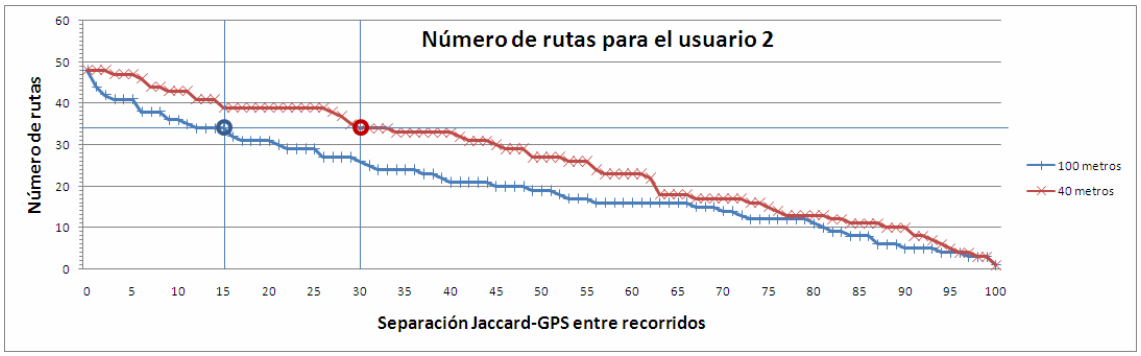

Figura 5.8: Número de rutas generadas al realizar el cluster jerárquico de recorridos.

el índice de separación entre ellos.

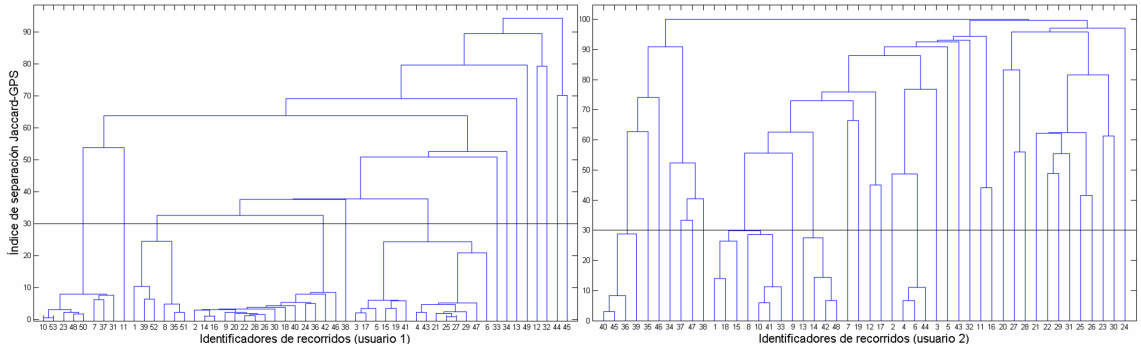

Figura 5.9: Dendrograma de rutas para el usuario 1 (izquierda) con una reducción del 73.58 % en y para el 2 (derecha) con una reducción del 29.17 %.

5.3. Predicción

Tras el preprocesado de los datos, nos disponemos a analizar los resultados de predicción obtenidos mediante las dos técnicas descritas en los capítulos 3 y 4. Por una parte la predicción mediante el mapa soporte (Sección 3.3.1) y la aplicación del modelo de Markov Oculto (Sección 3.3 y por otra, la aplicación directa de las

Tabla 5.6: Resultados de generación de rutas por cluster jerárquico de recorridos con regiones bola de radio 40 metros y agrupando con similitud superior al 70 %.

Usuario	Recorridos	Rutas	Reducción
1	53	14	73.58 %
2	48	34	29.17 %
3	84	34	59.52 %
4	126	17	86.51 %
5	308	161	47.73 %
6	305	103	66.23 %
Total	924	363	60.71 %

similitudes deslizantes (Sección 4.3.6).

5.3.1. Predicción mediante mapa soporte y HMM

En primer lugar, estudiamos el porcentaje de reducción de puntos obtenidos mediante la generación de puntos soporte con respecto al total de puntos tras el preprocesado. Para ello se ha elaborado la Tabla 5.7, en la cual podemos observar la alta reducción de puntos desde la información inicial hasta los puntos soporte, los cuales, recordamos, permiten la generación de un mapa personal.

Este es uno de los avances más importantes de nuestras aportaciones dado que en el peor de los casos (Usuario 1) obtenemos una reducción del 96.14 % (3.86 % del número de puntos filtrados), lo cual permite generar un mapa personal muy ligero y útil para el usuario en cuestión.

En nuestro análisis de las predicciones el criterio usado para validar la predicción de un destino final dado un recorrido parcial, fue una validación cruzada de M-capas sobre el conjunto completo de datos, donde M se escogió de acuerdo al tamaño de conjunto de datos. El procedimiento se repitió 50 veces para poder asegurar un

Tabla 5.7: Resultados de la obtención de puntos soporte.

	Puntos		Puntos soporte		
Usuarios	Total	Filtrados	Número	% del total	% de los filtrados
1	14884	7871	327	2.20 %	3.86 %
2	36161	10517	230	0.64 %	2.19 %
3	161426	57043	397	0.24 %	0.96 %
4	24099	24011	545	2.26 %	2.27 %
5	117820	36216	1054	0.89 %	2.91 %
6	143701	61179	1024	0.71 %	1.67 %

buen comportamiento estadístico y de esta forma evitar comportamientos extraños al considerar valores promedios. En cada recorrido de test, se extrajeron los puntos soporte generados para el 10, 25, 50, 75 y 90 por ciento del trayecto recorrido sobre el total. Los datos de entrenamiento nos permitieron generar un modelo de Markov Oculto y poste

riormente, utilizando el Algoritmo de Viterbi [Vit67], obtener el estado más probable (destino) para cada uno de los datos del conjunto de test. Los resultados se muestran en la Tabla 5.8 y en la Figura 5.10.

Hemos de señalar que existen dos posibles situaciones de que no ocurran predicciones correctas: predicción incorrecta o imposibilidad de predicción. La primera se da cuando se predice un destino que finalmente no es el correcto. La segunda cuando el recorrido no atraviesa ninguno de los puntos soporte generados en el conjunto de entrenamiento. Así, si en la zona de la predicción existen múltiples cruces de rutas, entonces prácticamente todos los recorridos contendrán puntos soporte, sin embargo, si el transeúnte o conductor realiza un recorrido por una zona en la que sus rutas no se cruzan con otras, puede que su recorrido no pase cerca de ningún punto soporte generado. Esta situación suele solventarse cuando el número de recorridos es mayor ya que es común que se generen más rutas y éstas se traduzcan en puntos soporte. También suele corregirse al aumentar el porcentaje de recorrido parcial atravesado.

Tabla 5.8: Resultados del experimento. M indica el número de capas en la validación cruzada. Los valores denotan el porcentaje de predicciones correctas.

	% Recorrido parcial atravesado				
Usuario$_M$	10 %	25 %	50 %	75 %	90 %
1_{10}	55.85 %	66.60 %	73.87 %	82.55 %	85.09 %
2_{10}	26.98 %	47.6 %	53.75 %	62.5 %	66.25 %
3_{10}	35.89 %	50.71 %	69.58 %	80.89 %	85.71 %
4_{10}	64.8 %	80.52 %	89.68 %	93.21 %	93.57 %
5_5	21.98 %	34.40 %	47.26 %	62.06 %	72.66 %
6_5	21.70 %	41.82 %	54.31 %	65.85 %	74.75 %

Como cabe esperar, podemos ver que según el porcentaje de recorrido atravesado se incrementa, el porcentaje de predicciones correctas mejora considerablemente. Esto resulta lógico debido a que el número de observaciones y cruces es mayor y las probabilidades de predicciones imposibles decrecen. Es importante señalar que aunque los resultados no están todo lo cercano al 100 % que nosotros desearíamos de corrección, los últimos puntos soporte obtenidos en el 90 % de los recorridos atravesados están lejos del 90 % del recorrido completo, en algunos casos varios kilómetros. Esto es debido al método con el que se han seleccionado los puntos soporte. Esta característica de nuestra aproximación nos permite realizar la predicción con suficiente antelación a la llegada al destino final.

Con respecto a los resultados, se puede observar que varían de un usuario a otro. Esto se explica debido a los diferentes patrones de movimientos, la zona en la que se desplazan y el número de lugares frecuentemente visitados por cada usuario. Por ejemplo, aunque existen bastantes más recorridos del usuario 5 y el 6, éstos visitan lugares cercanos entre sí, lo que implica que el número de predicciones correctas se reduzca. También puede observarse que el usuario 4 es el más predecible debido a que utiliza las mismas rutas y el número de destinos finales es pequeño.

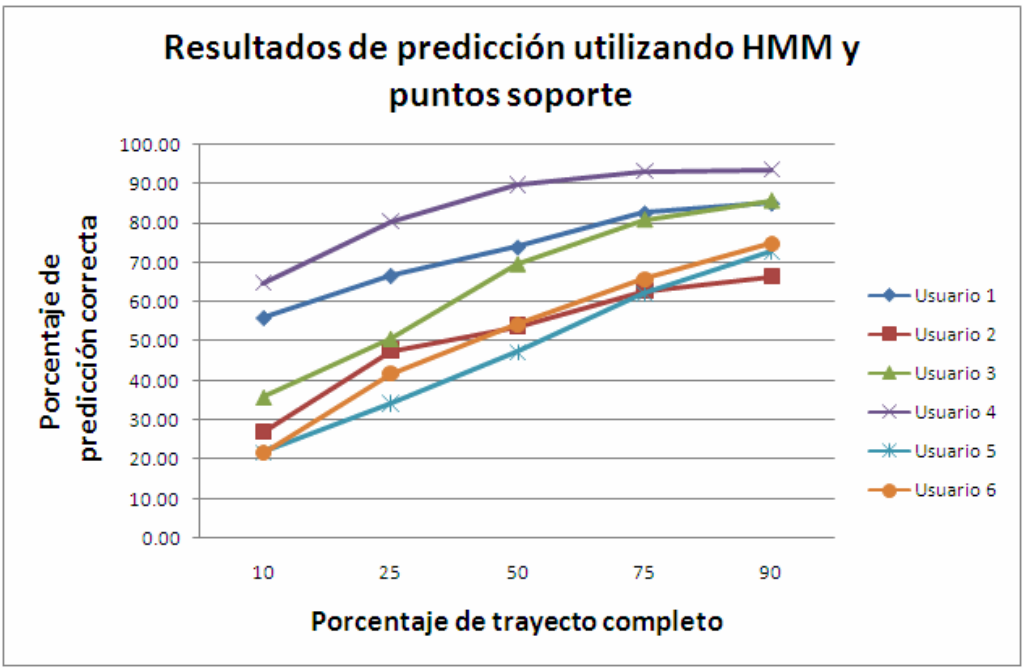

Figura 5.10: Resultados de predicciones correctas utilizando el mapa soporte y HMM.

Los resultados del modelo HMM basado en puntos soporte son esperanzadores teniendo en cuenta el número reducido de recorridos y la variabilidad de rutas de los diferentes usuarios.

5.3.2. Predicción a partir de similitud deslizante

El uso de similitudes tiene como principal ventaja que no es necesario entrenar al sistema, sin embargo el inconveniente resulta del número de datos que se almacenan ya que en esta aproximación se utilizan todos los puntos de cada ruta. Al igual que en la predicción a partir del HMM, en nuestro experimento también se utilizó una validación cruzada de M-capas sobre el conjunto completo de datos. Los resultados obtenidos con esta metodología son presentados en la Tabla 5.9 y en la Figura 5.11, para cada uno de los seis usuarios estudiados a diferentes niveles de trayecto atravesado.

Tabla 5.9: Resultados del experimento con similitudes. M indica el número de capas en la validación cruzada. Los valores denotan el porcentaje de predicciones correctas.

	% Recorrido parcial atravesado				
Usuarios_M	10 %	25 %	50 %	75 %	90 %
1_{10}	55.47 %	61.98 %	68.30 %	79.81 %	80.56 %
2_{10}	44.04 %	55.19 %	66.28 %	74.80 %	76.30 %
3_{10}	50.73 %	62.03 %	70.11 %	78.78 %	80.84 %
4_{10}	61.33 %	69.37 %	75.00 %	82.54 %	84.27 %
5_5	49.96 %	60.21 %	68.04 %	75.30 %	77.91 %
6_5	47.34 %	58.23 %	67.75 %	73.99 %	76.26 %

De estos resultados destaca el valor máximo de predicciones correctas para todos los usuarios, alcanzado por el usuario 4 en el 90 % de sus recorridos atravesados con un 84.27 %. Al estar lejos del 100 % estudiamos las predicciones incorrectas y encontramos que existían una serie de rutas que representaban únicamente a un recorrido. Si esta ruta no se parecía en nada a ninguna otra, cuando el recorrido estaba en el conjunto de test, incluso con éste finalizado, no era posible emitir una predicción correcta. De ese modo, todos los usuarios tienen una cota máxima de acierto por debajo del 100 %. Podemos ver esa situación en la Figura 5.12 donde el recorrido número 66 accede al destino 4 por una ruta totalmente diferente a la que lo hacen los demás recorridos que sí comparten tramos con los demás. Este fenómeno de rutas aisladas explica la lejanía del 100 % de aciertos.

Por otra parte, podemos observar cómo las pendientes de todas las curvas son pronunciadas hasta el 75 % del recorrido atravesado y más suaves en el 90 % ya que el tramo final el recorrido aporta menos información que los primeros puntos.

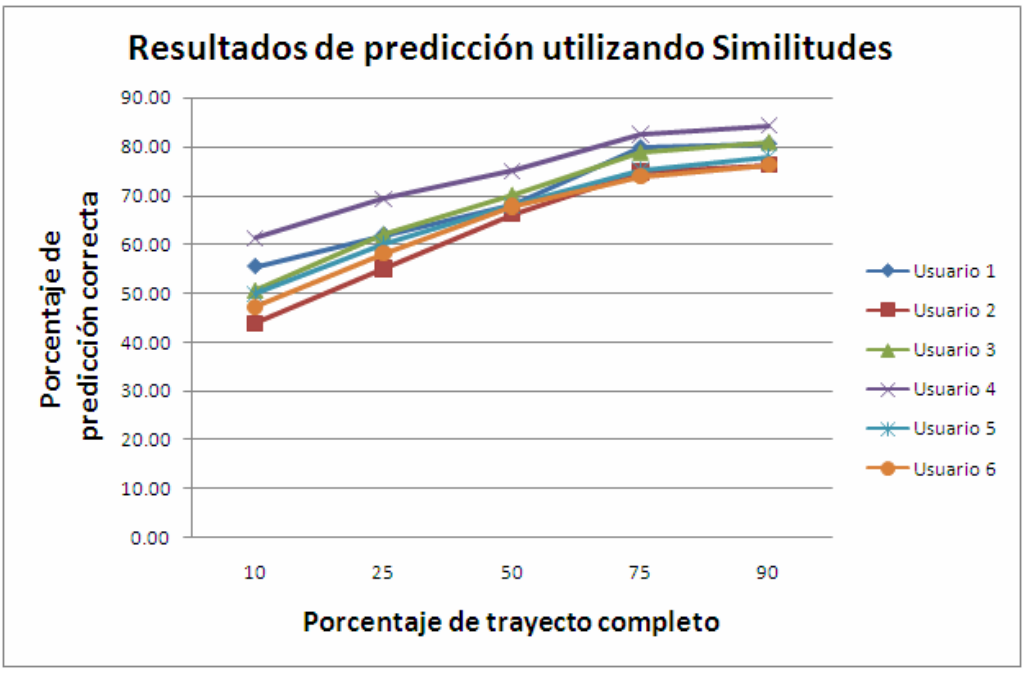

Figura 5.11: Resultados de predicciones correctas utilizando la similitud Jaccard-GPS.

5.3.3. Comparación de resultados

Mostramos a continuación en la Tabla 5.10 y en la Figura 5.13 los resultados de las predicciones anteriores de forma conjunta, así como las gráficas para cada uno de los usuarios con la escala personalizada para observar mejor las diferencias.

Al comparar los resultados de ambas técnicas debemos tener en cuenta varios factores críticos:

Número de recorridos "únicos". Las rutas poco frecuentes, concretamente las que representa únicamente a un recorrido, provocan errores inevitables en cuando estos recorridos se encuentran en el subconjunto de test. Esta situación influye en las cotas máximas que pueden alcanzar las predicciones de ambas técnicas.

Densidad de cruces entre rutas. Este factor afecta de manera capital a la ge-

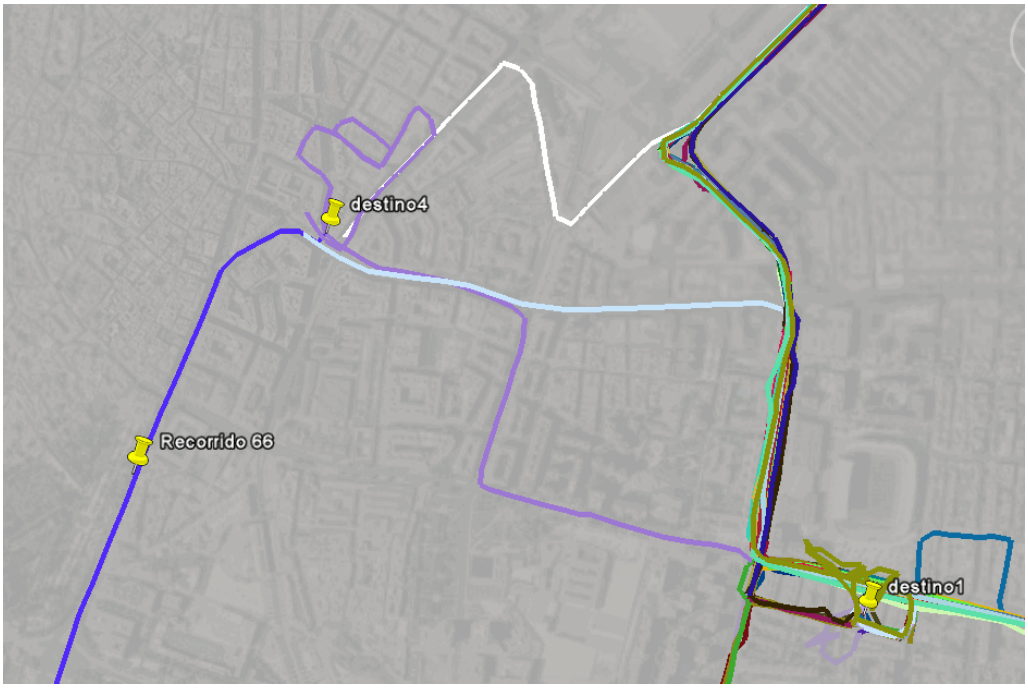

Figura 5.12: Ejemplo de recorrido con poca similitud con los demás.

neración del mapa soporte. Si la densidad de cruces entre las rutas de un determinado usuario es baja, sus predicciones serán pobres al no tener un mapa con información suficiente.

Frecuencia de toma de rutas. El modelo HMM implícitamente incluye información frecuencial, algo que el modelo de similitudes no hace. Eso significa que conocemos que el punto soporte por el que pasamos se da más cuando queremos alcanzar un destino que otro, además de conocer la probabilidad de hacer una transición entre estados. Esta característica hace que los usuarios con hábitos de desplazamientos más estables tengan mejores predicciones mediante el método de HMM.

Teniendo en cuenta esos factores, podemos observar de manera general como el modelo HMM comienza con predicciones bastante pobres con respecto al método de similitud deslizante. Esto se debe a que la densidad de puntos soporte es baja en los inicios de los recorridos. Sin embargo, esta densidad aumenta según avanza el

Tabla 5.10: Comparación de resultados entre HMM+Mapa soporte y Similitud entre recorridos.

Usuario	Tipo	10 %	25 %	50 %	75 %	90 %
1	HMM	55.85 %	66.60 %	73.87 %	82.55 %	85.09 %
	Similitud	55.47 %	61.98 %	68.30 %	79.81 %	80.56 %
2	HMM	26.98 %	47.60 %	53.75 %	62.50 %	66.25 %
	Similitud	44.04 %	55.19 %	66.28 %	74.80 %	76.30 %
3	HMM	35.89 %	50.71 %	69.58 %	80.89 %	85.71 %
	Similitud	50.73 %	62.03 %	70.11 %	78.78 %	80.84 %
4	HMM	64.80 %	80.52 %	89.68 %	93.21 %	93.57 %
	Similitud	61.33 %	69.37 %	75.00 %	82.54 %	84.27 %
5	HMM	21.98 %	34.40 %	47.26 %	62.06 %	72.66 %
	Similitud	49.96 %	60.21 %	68.04 %	75.30 %	77.91 %
6	HMM	21.70 %	41.82 %	54.31 %	65.85 %	74.75 %
	Similitud	47.34 %	58.23 %	67.75 %	73.99 %	76.26 %
Media	HMM	37.87 %	53.61 %	64.74 %	74.51 %	79.67 %
	Similitud	51.48 %	61.17 %	69.25 %	77.54 %	79.36 %

recorrido, por lo que su pendiente es más pronunciada que la del método de similitud.

Podemos observar, como cabe esperar, que en ambos procesos de predicción, a medida que aumenta el porcentaje de recorrido parcial realizado aumenta el porcentaje de aciertos. Sin embargo, es de destacar que ninguna de las dos alternativa es superior a la otra para todos los usuarios. Veamos esto como más detalles.

Si comparamos uno a uno cada uno de los resultados de predicción (por ejemplo después del 10 % del recorrido realizado por el usuario 1 el método HMM alcanza un 55,85 % mientras que el basado en similitudes alcanza el 55,47 % podemos declarar que HMM gana un punto) podemos constatar que el método basado en similitud deslizante es superior al método basado en HMM con un tanteo de 18 a 12 predicciones. Sin embargo, esto no es suficiente para declarar que uno es mejor que otro

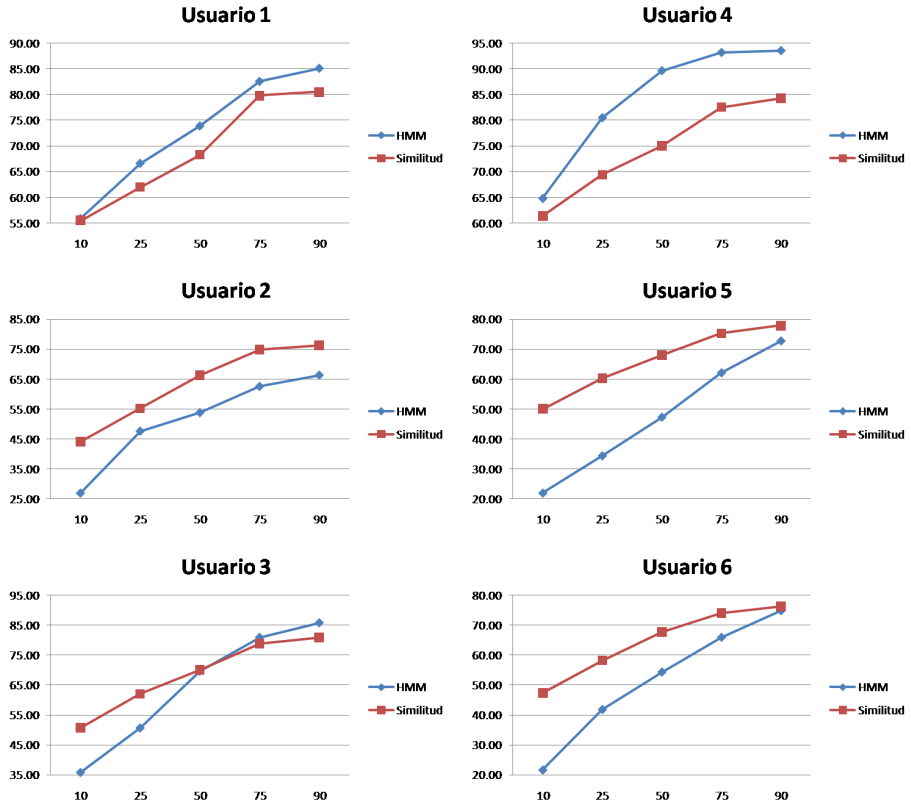

Figura 5.13: Comparación de los resultados de predicción obtenidos para cada usuario.

ya que estudiando particularmente cada usuario se tiene claramente que el método HMM es superior al de similitudes para los usuarios 1, y 4, y el de similitudes al HMM en los demás.

Estudiando a los usuarios 1 y 4, descubrimos que eran los que menos rutas utilizaban (sus recorridos podían reducirse a 14 y 17 rutas según vimos en la Tabla 5.6) y menos destinos visitaban (5 y 4 respectivamente según la Figura 5.4). Comparando esos datos con respecto a los de los usuarios 2, 3, 5 y 6 (34, 34, 161 y 103 rutas y 15, 10, 27 y 15 destinos, respectivamente), vemos como los patrones de comportamiento de los primeros eran mucho más repetitivos y por tanto más adecuados para trabajarlos mediante un modelo estadístico como el HMM que recupera información frecuencial. Los segundos resultaban menos previsibles por lo que utilizar la técnica

Tabla 5.11: Índice de bondad de las predicciones.

Usuario	Similitud	HMM
1	0.736	0.770
2	0.690	0.584
3	0.733	0.713
4	0.782	0.879
5	0.711	0.552
6	0.699	0.593
Media	0.725	0.682

de similitudes permitía incluir más información espacial (en el HMM sólo incluíamos puntos soporte mientras que en el otro incluíamos todos los puntos de las rutas) y no importaba tanto los datos frecuenciales.

Utilizando el índice IP propuesto (ver Sección 2.2.12) obtenemos la Tabla 5.11, donde se puede constatar lo apuntado anteriormente referente a que el método HMM es superior para los usuarios 1 y 4. En la tabla también se ha añadido una fila donde se indica cual es el porcentaje de acierto medio para cada uno de los métodos y se observa que el método de similitud deslizante mejora al HMM.

5.4. Resumen

En este capítulo hemos comprobado los resultados de aplicar la metodología off-line propuesta a un conjunto de datos de casi medio millón de puntos geo-posicionados mediante el sistema de localización GPS.

En cuanto al proceso de extracción de información, es importante resaltar la mejora en el segmentado de recorridos aplicando el método por cercanía a destinos con una mejora media del 13 % con respecto a la técnica comúnmente aceptada del segmentado temporal.

Por otra parte, cabe destacar los resultados obtenidos mediante el uso de la combinación de los mapas personales (rutas y puntos soporte) con las dos técnicas de predicción (similitud deslizante y HMM). Concretamente la técnica de similitud utilizando rutas permite una corrección media superior al 50 % habiendo recorrido sólo el 10 % del recorrido. Aunque los resultados del HMM junto con puntos soporte son relativamente peores en el conjunto de los seis usuarios, observamos que en aquellos usuarios con los hábitos más repetitivos (usuarios 1 y 4) esta regla también se cumple.

Una vez obtenidos estos resultados, estamos en disposición de adaptar esta metodología a entornos on-line, es decir entornos en los que tengamos que extraer la información de los mapas personales y realizar predicciones en tiempo real. En el Capítulo 6 veremos esta adaptación a terminales móviles cuyo mayor problema es el reducido tiempo de duración de sus baterías.

CAPÍTULO 6

PREDICCIÓN ON-LINE

En los desarrollos anteriores hemos trabajado con resultados *off-line*, es decir, primero recuperábamos los datos mediante un receptor GPS y posteriormente los tratábamos en un ordenador. Sin embargo, la utilidad de las aplicaciones para las que están diseñadas nuestras aproximaciones radica en una predicción *on-line*. En este capítulo explicaremos el proceso seguido para llevar a la práctica esta predicción en un dispositivo móvil e incidiremos sobre las técnicas utilizadas para evitar la interacción con el usuario y reducir el consumo energético del terminal. De ese modo permitiremos utilizar la predicción de destinos de manera transparente y durante el máximo tiempo posible entre cada carga de baterías.

6.1. Estado del arte

Para conseguir predicciones en tiempo real es necesario considerar la combinación de un receptor GPS para recoger la información, una CPU capaz de procesar la información y ejecutar el motor de predicciones y una base de datos que almacene la base de conocimiento compuesta por rutas y destinos. Frente a estas restricciones, existen dos soluciones según el lugar en el que se realice la predicción:

1. **Predicción en el dispositivo.** El dispositivo móvil integra tanto el receptor GPS como la CPU y la base de datos. Las características de esta solución son:

   SGBD poco maduros. La madurez de los sistemas gestores de bases de datos en terminales móviles dista mucho de la de los servidores. La sencillez con la que se implementaban las relaciones entre tablas, los reducidos tipos de datos aceptados y la baja precisión de éstos complica el almacenamiento y gestión de datos referentes al posicionamiento.

   Capacidad de cálculo limitada: Este aspecto es cada vez menos relevante por el rápido avance de la computación móvil. Sin embargo, el uso intensivo de la CPU del dispositivo incide negativamente en el siguiente punto.

   Baterías del dispositivo limitadas: Mientras las dos primeras características evolucionan rápidamente, esta tercera no lo hace tanto. Aunque ya han aparecido en el mercado los primeros móviles solares[(1)], el consumo de baterías del chip GPS incluido en los teléfonos es tal que hace que en pocas horas se agote en prácticamente todos los terminales.

2. **Predicción en un servidor.** El dispositivo personal integra el receptor GPS y un emisor/receptor inalámbrico para el envío y recepción de la información a un servidor. Éste, almacena la base de conocimiento y ejecuta su motor de predicciones con los datos recibidos. Posteriormente envía al dispositivo la predicción realizada. Los aspectos más relevantes de esta solución son los siguientes:

   Consumo de baterías. Además del coste obligatorio de obtener los puntos GPS, tal y como tenemos en la primera solución, las baterías del dispositivo se reducen rápidamente debido a las numerosas transmisiones de datos entre dispositivo y servidor, influyendo también su duración.

   Coste económico. Transmitir cada cierto número de segundos la posición espacial por parte del cliente y recibir la respuesta del servidor supone

[(1)] http://www.reghardware.co.uk/2009/04/20/waterproof_solar_kddi/

transmisiones durante todo el día, lo cual supone evidentemente un coste monetario. Aunque el coste de las transmisiones a través de la red telefónica se ha abaratado en los últimos años[(2)], puede que los usuarios no estén dispuestos a pagar por este tipo de servicios.

Desconfianza sobre la privacidad. El envío de información de localización a un servidor supone dos posibles amenazas a la privacidad: la captura de datos a través del canal inalámbrico y el uso indebido de los datos almacenados en un servidor ajeno al usuario. Aunque estas amenazas pueden prevenirse con una gestión segura de la comunicación y el almacenamiento, el usuario puede desconfiar del tratamiento de su información de localización por parte de terceros.

La segunda opción resulta útil en aplicaciones multiusuario como los sistemas de gestión de tráfico o en sistemas que necesitan de CPUs muy potentes. Por ejemplo podemos ver esta alternativa en el trabajo de Patterson [PLG+04] donde el servidor utiliza servicios de localización en tiempo real de los autobuses y de un GIS de la ciudad de Washington para ejecutar el motor de inferencia de predicciones. También podemos comprobarlo en el estudio de Brakatsoulas [BPT04] donde se describe un sistema de gestión de tráfico capaz de detectar situaciones problemáticas en la red de carreteras en Atenas. En el caso particular de la gestión de tráfico, la información del estado de la circulación de vehículos se obtiene de servicios de transportes públicos, coches de policía o vehículos que recorren la ciudad expresamente para medir la densidad del tráfico en tiempo real. Estos vehículos no son privados, por lo que no existen problemas de privacidad cuando se transmiten los datos al servidor dado que no son datos personales. Sin embargo, si consideramos que en vez de vehículos de trabajo, monitorizamos personas en cualquier instante y lugar, la invasión de la intimidad es un problema importante.

[(2)] Yoigo y Orange (en verano de 2009) ofrecen una tarifa plana diaria de 1.2 euros más IVA.

6.2. Nuestra propuesta

Para conseguir nuestro objetivo, predecir destinos de usuarios, es necesario realizar una monitorización continua de éstos, por lo que aunque el problema del consumo de baterías es un aspecto que debemos asumir, el coste económico y fundamentalmente la falta de privacidad no son aceptables.

Dado que deseamos que los usuarios confíen en el servicio de predicción de destinos y que de esa manera lo adopten sin ninguna objeción, la opción elegida en nuestra investigación fue la de integrar todo en el dispositivo móvil preservando la intimidad y reduciendo el gasto económico y energético de las transmisiones inalámbricas. Para comprobar su viabilidad, construimos un prototipo, comprobamos su debilidad en cuanto a duración de baterías y gestionamos el consumo energético basado en el diseño del GPS y el comportamiento en exteriores de los usuarios.

6.2.1. Prototipo

Para desarrollar un prototipo capaz de llevar a cabo la predicción de destinos *on-line*, se utilizó un terminal HTC P3300 que podemos ver en la Figura 6.1.

Al intentar implementar las dos técnicas de predicción utilizados en la fase *off-line*, HMM con mapas soporte y similitudes, analizamos dos indicadores para cada uno de ellas: la base de conocimiento inicial necesaria para poder realizar predicciones y el procesado necesario tras finalizar un recorrido.

La base de conocimiento inicial nos indica el número de días necesarios para que el sistema comience a ser útil. En la técnica HMM, al estar basada en un mapa soporte, era necesario crearlo previamente. Eso supone tener un número de recorridos elevado para poder generar múltiples cruces entre rutas. El número medio de recorridos para conseguir las primeras predicciones correctas era de unos diez días.

Sin embargo, utilizando la técnica de similitudes, una vez que definidos cuales son los destinos frecuentes, con pocos recorridos podríamos predecir los extremos de

Figura 6.1: Terminal HTC P3300 utilizado para el prototipo de predicción *on-line*.

éstos. Normalmente a partir del tercer día de desplazamientos ya se comenzaban a tener predicciones correctas.

El segundo indicador, el procesado tras el fin de cada recorrido, nos informa de la cantidad de procesamiento necesario que debe realizar la CPU del dispositivo. Para la técnica de HMM y mapas soporte en primer lugar es necesario comprobar si el recorrido pertenece a una ruta antigua o no. A continuación debemos regenerar el mapa soporte revisando cada uno de los nuevos cruces con rutas anteriores y por último volver a crear las matrices de transición y emisión ya que suelen aparecer

nuevas observaciones o estados. El proceso realizado en un portátil Intel Core 2 Duo P8400 a 2.26GHz y 3GB de memoria RAM utilizando un SGBD SQL Server 2005 tardaba entre 7 y 15 minutos por lo que aunque el procesador del dispositivo fuese potente ese cálculo podría tardar mucho más.

Utilizando la técnica de similitud deslizante únicamente era necesario comprobar si el recorrido coincidía con una ruta antigua o era necesario crear otra para representarlo, por lo que el tiempo de procesado era mínimo.

Al ser tan desfavorables las condiciones para el modelo de mapas soporte *on-line*, nos decidimos por el uso de la técnica de similitudes. Desarrollamos una aplicación prototipo que era activada y desactivada por el usuario cada vez que comenzaba un nuevo desplazamiento. La base de datos que utilizaba la aplicación se cargaba previamente utilizando un conjunto de recorridos extraídos de la fase *off-line*. De ese modo la aplicación simplemente utilizaba el motor de inferencia basado en la base de conocimiento existente para comprobar el tiempo de respuesta del procesador y la corrección de las predicciones. A partir de este trabajo observamos los siguientes hechos:

Carga de recorridos. Depender de una recuperación *off-line* de recorridos para cargarlos en un dispositivo móvil no era una tarea sencilla ni intuitiva, por lo que en la aplicación final, el proceso de recogida y extracción de recorridos y destinos finales debía estar integrado de manera transparente.

Interacción por parte del usuario. La activación y desactivación de la aplicación por parte del usuario no era aceptable. Al igual que en el proceso de recuperación de datos de la fase *off-line*, se producían olvidos. Además, la molestia para el usuario era importante.

Respuesta del procesador. Aunque pensamos que podría haber retardo en la aplicación de la técnica de similitudes no fue así y aún recuperando datos GPS a la frecuencia máxima (1Hz) no hubo problemas, la predicción se realizaba cada segundo y se notificaba visualmente al usuario. En este prototipo no nos

preocupamos de cuando realizar la notificación pero consideramos que para la aplicación final sería interesante buscar intervalos de predicciones estables (que indicasen el mismo destino) durante al menos un minuto.

De lo anterior concluimos que nuestro sistema debía ser transparente al usuario y de ejecución continua, con lo que el problema de baterías se acrecentaría. Para ello en primer lugar definimos un diagrama de procesos para evitar que el usuario interactúase con el sistema y en segundo estudiamos las técnicas de reducción de baterías.

6.2.2. Diagrama de procesos

En la metodología estudiada, trabajamos sobre una base de conocimiento estática de recorridos y destinos que no variaba con el tiempo. El reto al que nos enfrentamos ahora es que esa base de conocimiento se incrementa a medida que el usuario realiza sus desplazamientos. Así mismo, desde el primer instante en el que sea posible, se deben emitir predicciones de ruta y destino.

A continuación pasamos a describir los aspectos más interesantes del diagrama de procesos que difiere en algunos puntos de la versión *off-line*.

En la Figura 6.2 podemos ver el esquema seguido. Hemos numerado cada uno de los bloques para poder explicarlos fácilmente.

Consideramos como punto de partida el estado de fin de recorrido (bloque 1), es decir, el instante en el que el usuario se encuentra en un destino o acaba de llegar a él. El objetivo es detectar un nuevo desplazamiento (bloque 2). El evento que lo dispara es la obtención durante un cierto tiempo (en la implementación se usaron 45 segundos) de un porcentaje suficiente (nosotros escogimos el 90 %) de puntos válidos con HDOP menor que 6 que supusiesen una distancia recorrida de 50 metros. Una vez detectado, se van almacenando todos los puntos válidos tras aplicar los filtrados de distancia, velocidad y precisión. Al mismo tiempo, se aplican las técnicas de predicción para emitir conjeturas sobre la futura ruta seguida y el destino final

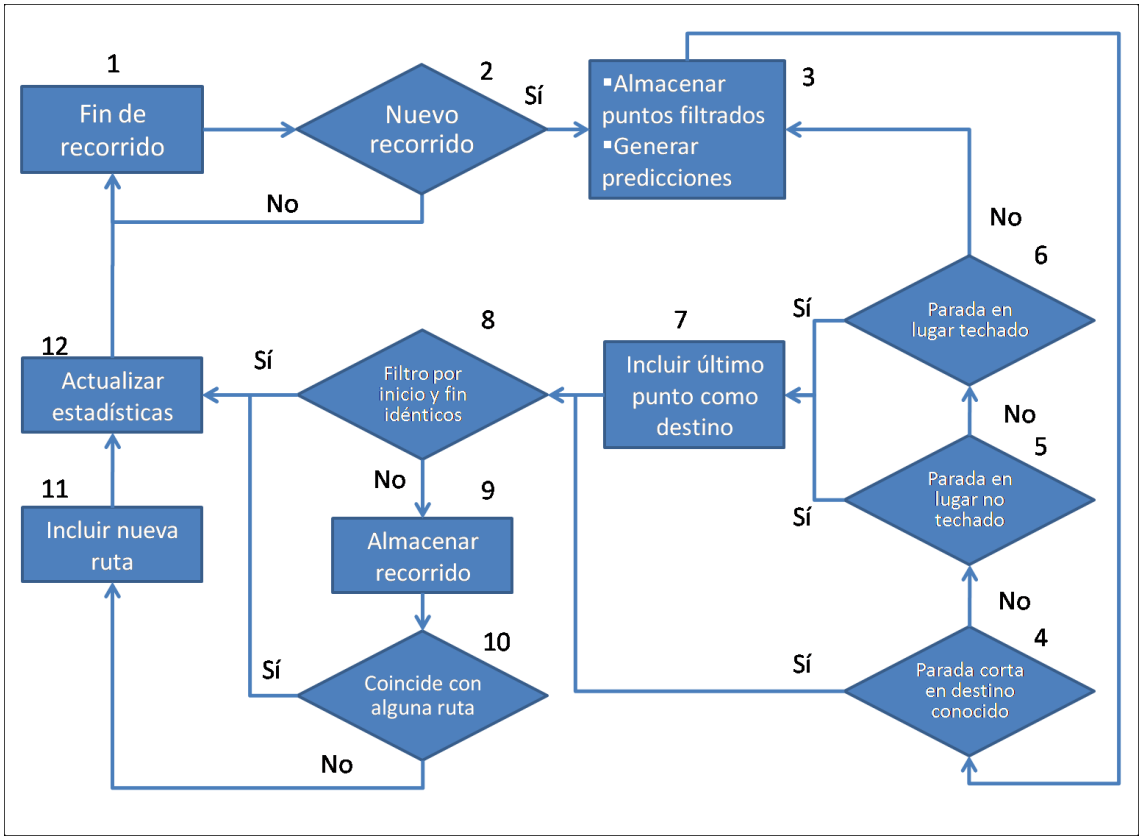

Figura 6.2: Esquema seguido en el prototipo desarrollado en el terminal móvil para realizar una predicción *on-line*.

(bloque 3). Tras cada nuevo punto válido, se comprueban las condiciones de fin de recorrido o segmentado, es decir:

- Parada corta en destino conocido (bloque 4). El segmentado por distancia a destinos simplemente comprueba si el usuario pasa más de un umbral temporal en un destino conocido. El tiempo de parada considerado fue de 40 segundos tal y como vimos en la Sección 2.2.6.
- Parada en lugar no techado (bloque 5). Si no perdemos la señal GPS pero pasamos más de 5 minutos en un lugar en el que no variemos significativamente nuestra posición, consideraremos el final del recorrido.
- Parada en lugar techado (bloque 6). La pérdida de la señal GPS suele significar

la entrada en edificios, sin embargo a veces también supone la entrada en un túnel y su consiguiente tiempo de espera para volver a conseguir un punto válido. 5 minutos fue el umbral considerado para nuevos destinos (si el destino ya había sido analizado nos quedaríamos en el bloque 4).

Las condiciones de los bloques 5 y 6, suponen el descubrimiento de nuevos destinos, por lo que es necesario aplicar el algoritmo de agrupamiento de puntos finales o cluster de destinos estudiado en la Sección 2.2.5. Aunque en la propuesta *off-line*, realizamos un cluster de todos los puntos finales, en on-line, debemos considerar que cuantos más recorridos realice el usuario, mayor será el conjunto de todos los puntos finales. Para no ir realizando agrupamientos de destinos incrementalmente con cada recorrido finalizado, decidimos considerar dos conjuntos: el de destinos con una única repetición (D) y el de destinos con tres o más repeticiones ($D3$). De ese modo con cada nuevo punto final, se computan las distancias Haverseno con respecto a todos los elementos de $D3$. Si dista menos de 200 metros con alguno de sus componentes, directamente sin realizar el proceso de cluster se etiqueta el destino como el del componente de $D3$. Si no se cumple la anterior condición, se inserta en el conjunto D y se aplica el cluster jerárquico por distancia. Si existen agrupaciones con 3 elementos, se eliminan esas componentes del conjunto D y se inserta como un único elemento en $D3$. De ese modo ni D ni $D3$ crecen de manera excesiva, reduciendo así los cálculos más costosos y repetitivos.

Una vez considerado el último punto destino, debemos contemplar que el recorrido tenga como inicio el mismo lugar que como fin, es decir recorridos circulares (bloque 8). Aunque este tipo de recorridos puede darse, creemos que no son importantes para el sistema por lo que se obvian.

Si el recorrido no ha sido filtrado por inicio y fin idénticos, pasamos a incluir el recorrido en nuestra base de conocimiento (bloque 9). Eso supone que los puntos acumulados desde la detección del inicio hasta su finalización deben ser considerados como un recorrido y pasa a ser comparado con las rutas existentes (bloque 10). En esta comparación se aplica la similitud Jaccard-GPS (ver Sección 4.3.3). Si el

recorrido se parece lo suficiente a alguna de las rutas, no es necesario hacer el cluster, sólo elegir cuál será el recorrido representante. Si la similitud no es suficiente, pasa a ser una nueva ruta (bloque 11).

Tras haber identificado correctamente el nuevo recorrido (flecha desde bloque 11) o haberlo filtrado (flecha desde el bloque 8), es necesario actualizar las estadísticas que servirán para llevar un histórico de los desplazamientos. Entre los datos que se añaden a las estadísticas, se incluyen:

- Resumen del recorrido, entendido como lugar origen y destino, fecha y hora de comienzo y fin.
- En caso de que proceda, es decir, si no hemos filtrado el recorrido, ruta seguida, indicada por el identificador de la ruta almacenada.
- Medio de transporte utilizado a partir de velocidad media y la velocidad máxima. Los únicos considerados fueron: vehículo motorizado, bicicleta y a pie.
- Distancia recorrida y desnivel entre punto inicial y final.

Tras esa actualización volvemos al bloque 1 donde se buscará el comienzo de un nuevo recorrido. En este punto cabe destacar el artículo [BLP+08], en el que ante la pérdida de señal GPS, disminuyen la frecuencia de muestreo de puntos, llegando hasta una toma GPS cada ocho minutos. En nuestro caso consideramos también los lugares no techados por lo que no se consideró esta reducción.

6.2.3. Reducción de baterías

El anterior diagrama permitía una ejecución continua de la predicción de destinos pero la duración de las baterías hacía que este ciclo no se completase durante más de siete horas.

Una de las opciones barajadas fue la de utilizar un receptor GPS externo al dispositivo móvil para evitar el consumo extra del chip GPS del teléfono, tal y

como hicimos con la combinación de la PDA Dell X30 y el receptor GPS RoyalTek. Aunque se comprobó que existen trabajos en los que siguen esta alternativa como [YCcC09], (teléfono Nokia N70 y receptor GPS HOLUX 1000) y [ZLTG07] (Motorola A1200/Ming junto con receptor GPS no especificado), nuestra experiencia nos indicó que tener que llevar y recargar dos dispositivos diferentes duplica las posibilidades de olvido. De ese modo decidimos utilizar un único dispositivo móvil.

Como hemos comentado, la tecnología GPS es la más precisa en exteriores y sobre todo, es global. Sin embargo su coste en baterías y su inutilidad en interiores hizo que buscáramos una técnica que nos permitiese activar el GPS sólo cuando comenzásemos un nuevo recorrido y desactivarlo cuando llegásemos a un destino. A continuación estudiamos el comportamiento de los usuarios participantes para comprobar el tiempo que estuvieron en exteriores durante la fase de recogida de datos.

6.2.3.1. Comportamiento de los usuarios

Tras observar los recorridos de los diferentes participantes, comprobamos que de media los 6 usuarios permanecían diariamente en exteriores 62.5 minutos, es decir, el 4.34 % del día. Este dato es similar al obtenido en el trabajo PlaceLab [LCC+05] (de media el 4.5 %) y algo inferior comparado con el estudio de Toole [THPP05] de 82.3 minutos (5.71 %). Esta información nos permitió corroborar que nuestra muestra de trabajo se podía considerar como una muestra estándar.

De ese modo la batería del dispositivo móvil podía durar mucho más tiempo si durante casi el 95 % del día no estaba activo el chip receptor GPS.

6.2.3.2. Estudio de consumo energético

Durante el periodo de tiempo en el que el chip GPS está inactivo, es necesario utilizar otra tecnología que permita la detección de comienzo de un desplazamiento. Las alternativas estudiadas fueron varias:

Sensores de luminancia. El paso de interiores a exteriores, que suele determinar el comienzo de un recorrido puede ser detectado mediante un cambio de luz. Sin embargo, normalmente el terminal móvil se lleva en un bolsillo, además hay situaciones en las que el destino es un lugar no techado con lo que no existe esa transición. Por todo lo anterior esta opción se descartó desde un principio.

Huellas Wifi y antenas GSM. Tras estudiar los trabajos de Hightower [HCL+05] y Kang [KWSB04] observamos cómo la localización basada en estaciones Wifi y antenas GSM geo-localizadas era una buena opción para posicionar a un usuario sin necesidad de GPS.

Acelerometría. Los trabajos [APAO07, PAlGO07] en los que participamos para la monitorización del movimiento de personas en interiores y la integración de acelerómetros tri-axiales en dispositivos móviles también fueron motivos para su consideración.

Tras estas posibilidades estudiamos también la repercusión energética de las diferentes tecnologías en la batería del dispositivo móvil. Para hacer las medidas utilizamos el terminal que podemos ver en la Figura 6.3, un móvil Samsung Omnia ya que integraba radio WiFi y acelerómetro tri-axial, esto último no disponible en nuestro HTC P3300.

En la Figura 6.4 podemos ver la evolución de la batería del terminal manteniéndolo continuamente activo y con cada una de las tecnologías en funcionamiento.

Las pruebas se realizaron partiendo de una carga completa del dispositivo y finalizaban cuando éste se apagaba debido a la falta de batería. Para ver la evolución, desarrollamos un programa que muestreaba cada minuto el nivel de batería. En la gráfica se destacan únicamente los puntos que suponen un cambio del 10 % con respecto al nivel anterior. Como podemos ver, la tecnología que más batería consume es la GPS, con un tiempo medio de vida de unas 7.5 horas. Le sigue de cerca la radio WiFi con unas 10.5 horas. Mucho menos costosa es la detección de antenas GSM con 46.5 horas y el acelerómetro con algo más de 47 horas. En la gráfica se superponen los puntos de GSM, acelerometría y sensor de luminancia.

Figura 6.3: Terminal Samsung Omnia utilizado para la mejora de tiempo de baterías.

Tras ver las tecnologías y los costes de éstas, estudiaremos cómo pueden ayudarnos a detectar el comienzo de un nuevo recorrido modificando el bloque 2 del diagrama de procesos.

6.2.3.3. Antenas GSM y puntos de acceso Wifi geo-localizados

En el trabajo de Intel liderado por Lamarca [LCC+05], se utilizan bases de datos geo-localizadas de puntos de acceso WiFi y antenas GSM, por lo que pueden servirse de estas señales para localizar a un usuario. Los puntos de acceso WiFi dan una

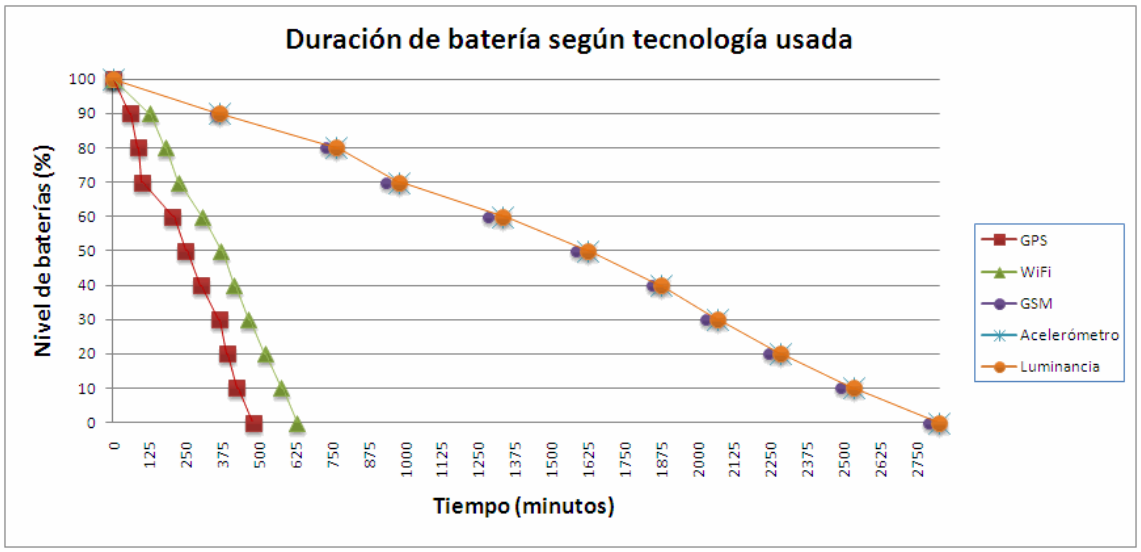

Figura 6.4: Comparación de la duración de baterías para un Samsung Omnia según el tipo de sensor activado.

precisión de aproximadamente 25 metros y si la densidad es suficiente[3] permiten el seguimiento de un usuario. Las antenas GSM dan un radio de cobertura que varía desde 100 metros (en ciudades con alta densidad de población) hasta varios kilómetros (en zonas rurales). Aunque son mucho menos precisas, se estima[4] que para el 2010 darán cobertura al 90 % de la tierra (sin incluir océanos) por lo que pueden servir para situar al usuario en lugares donde no hay redes WiFi.

Llevando al límite esta línea, Jun Rekimoto de Sony [RMI07] comprobó que era posible realizar seguimiento preciso de la localización de un usuario utilizando una gran base de datos de más de medio millón de puntos WiFi geo-localizados en las grandes ciudades de Japón donde su densidad es muy elevada. Un ejemplo puede verse en la Figura 6.5 donde se comparan los recorridos obtenidos por GPS y por puntos de acceso WiFi. Se observa como tanto en interiores como en exteriores la segunda opción es mejor que la primera dado que en ciudades como Tokio hay

[3] En Seattle, donde hacen el estudio se observa una densidad de 1200 puntos de acceso por kilómetro cuadrado.

[4] http://www.zdnetasia.com/news/communications/0,39044192,61960272,00.htm

muchos rascacielos y el efecto “cañón urbano” se acentúa.

Figura 6.5: Comparación de la precisión de seguimiento utilizando GPS y puntos de acceso WiFi geo-localizados.

Utilizar esta técnica de localización mientras el chip GPS está inactivo reduciría algo el consumo, pero el problema es que en países como España, aunque la densidad de puntos de acceso WiFi es elevada, no existen bases de datos tan detalladas como las usadas en Estados Unidos y Japón [LCC+05, RMI07]. Esto puede observarse en la Web de Wigle.net y en la de PlaceEngine de las que se extrajo la Figura 6.6. En ella puede observarse con colores más intensos las zonas con mayor densidad de puntos de acceso WiFi geo-localizados.

Aunque podríamos utilizar las redes WiFi sin geo-posicionar como marcadores para el lugar donde se desconectó el chip GPS, finalmente optamos por la opción de acelerometría que suponía un ahorro energético mayor.

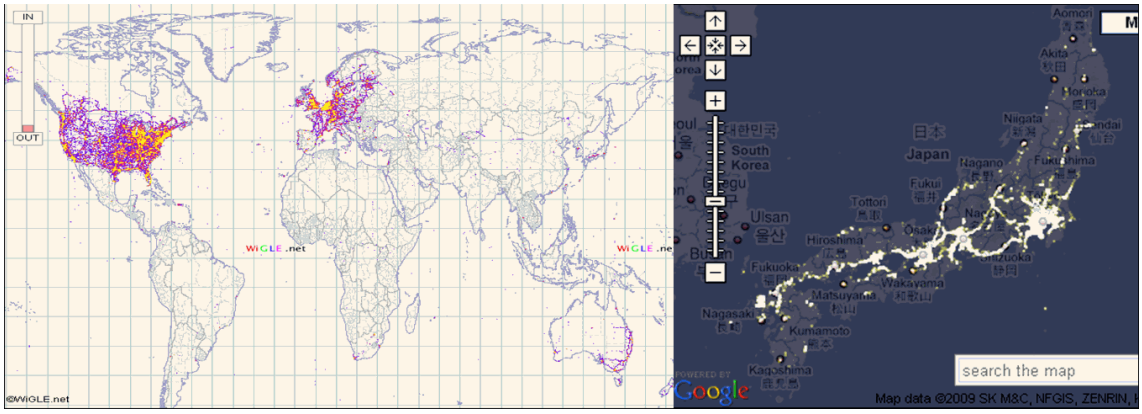

Figura 6.6: Puntos de acceso WiFi geo-localizados en las bases de datos de Wigle.net con cobertura global (izquierda) y PlaceEngine con cobertura para Japón (derecha).

6.2.3.4. Acelerometría

El 28 de julio de 2009, Kanw ar Chadha, fundador de SiRF[5], empresa líder en chips y módulos de GPS (recientemente adquirida por CSR[6]) anunció[7] el nuevo chip SiRFstarIV para principios de 2010 cuyas especificaciones dividen entre dos el consumo y añaden un modo de hibernación inexistente en el anterior chip SIRFstarIII con el objetivo de mantenerse continuamente activo en exteriores. Éste incluye una interfaz que permite utilizar otros sensores disponibles en un terminal móvil como un acelerómetro.

Parece obvio que la solución a la detección de nuevos recorridos viene con la fusión del chip GPS y la acelerometría. Aunque existen aproximaciones hardware como [RG07], la alternativa más cercana la propone Rahmati [RZ07] dónde se conecta un acelerómetro externo a las baterías de diferentes móviles para demostrar las posibilidades del uso de la acelerometría.

[5] http://www.sirf.com
[6] http://www.csr.com
[7] http://www.gpsworld.com/consumer-oem/news/sirfstariv-debuts-with-promise-always-on-location-awareness-8583

El acelerómetro del Omnia tiene una resolución de 0.004 g, es capaz de medir aceleraciones de ±2 g y la frecuencia de muestreo es de aproximadamente 6 Hz. Aunque otros dispositivos con los que hemos trabajado como el Alive Heart Monitor[(8)] o el KneeMeasurer [CMJ+07] trabajan a frecuencias mucho más elevadas (75 y 30 Hz respectivamente), la frecuencia que ofrece el móvil es suficiente como para detectar movimientos y clasificarlos.

Para la detección de comienzo de un recorrido supusimos que el usuario debía desplazarse desde el interior del edificio en el que se encontraba, durante un breve periodo de tiempo, hasta llegar al exterior. Consideramos que este desplazamiento se realiza siempre andando[(9)], ya sea hasta el garaje donde se encuentra su vehículo o hasta la parada del medio de transporte más cercano. Comprobamos que el desplazamiento a pie, suponía cambios de aceleración bruscos sin importar la posición del terminal (en el bolsillo, en la cintura, en un bolso, etc.) por lo que utilizamos el módulo de la aceleración de los tres ejes del acelerómetro y evaluamos ventanas temporales de cinco segundos. En cada una de ellas extrajimos la diferencia de valores máximos y mínimos tras obtener varias muestras por diferentes usuarios, establecimos un valor umbral que indicaba la existencia de un movimiento brusco en esa ventana. Para detectar un comienzo de recorrido, se consideró que debían darse al menos nueve ventanas con movimiento (45 segundos). En la Figura 6.7 podemos ver un ejemplo de transición de interior hacia exterior andando.

Cabe destacar que para el reconocimiento de cinco actividades (parado, andando, corriendo, en vehículo, en bicicleta) realizamos pruebas en las que usamos un clasificador bayesiano (Naive Bayes) sobre varias características (rango, varianza, coeficiente de Pearson y suma de las componentes FFT entre 1 y 5 Hz del módulo de la señal) y para modelar un efecto de inercia entre transiciones, evitando así saltos bruscos entre actividades, se aplicó tras la clasificación un modelo de Markov de segundo orden. Sin embargo, dado que en nuestro trabajo no buscamos clasificar

[(8)] http://www.alivetec.com/products.htm

[(9)] Queda fuera del ámbito de este trabajo las personas que se desplacen en silla de ruedas por las grandes diferencias existentes en cuanto a acelerometría.

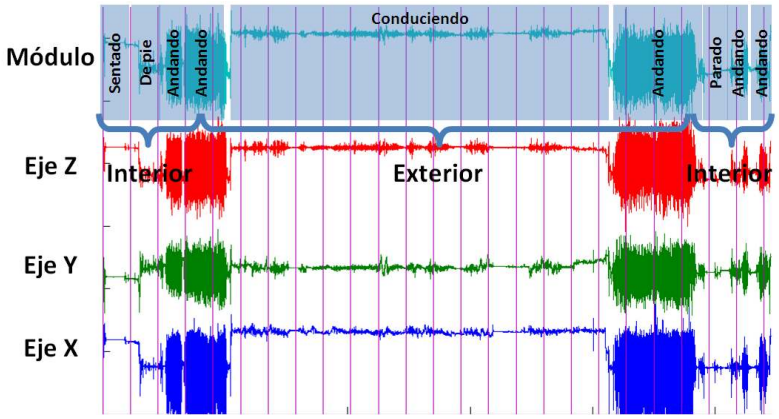

Figura 6.7: Estudio de transiciones de interiores a exteriores basada en la acelerometría.

diferentes actividades sino encontrar únicamente cuando el usuario está andando, el proceso era mucho más simple.

Las pruebas indicaron que el umbral era el adecuado porque conversaciones por el teléfono mayores que el umbral de los 45 segundos eran ignoradas como desplazamientos a pie (a no ser que éstas se realizasen andando).

Los resultados fueron muy importantes por el aumento de la autonomía del dispositivo aún con falsos positivos por movimientos en interiores, pasando de 7.5 horas a cerca de 22 (dependiendo del tiempo que estuviese desplazándose el usuario en exteriores), suficientes como para completar prácticamente un día de movimientos.

Tras finalizar el desarrollo, volvimos a revisar el estado del arte encontrando un par de trabajos de 2009 [FBAT09, RLRE09] donde hacen estudios similares, el primero con un estudio teórico en el que podrían llegar a registrar movimientos con un Nokia N95 durante 22 horas y el segundo con una aproximación parecida a la nuestra en la que consiguen que un móvil con sistema operativo Android (no especifican qué modelo) pase de 6.3 horas a 9 horas enviando además datos a través de GSM a un servidor.

Con estos resultados modificamos el diagrama de procesos descrito en la sección

6.2.2. En el bloque 2 en vez de utilizar la recuperación de puntos válidos por parte del GPS, utilizamos la detección continuada de movimiento del acelerómetro. Además, en el bloque 1 modificamos también que al detectar el fin del recorrido, el chip GPS se desactivaba para economizar baterías.

6.3. Resumen

En este capítulo hemos abordado la migración del sistema de predicción off-line propuesto en el Capítulo 2 a un terminal móvil de última generación. Aunque los procesadores y memorias de estos dispositivos son cada vez más potentes, se trata de un sistema con grandes restricciones de batería y por tanto de vida útil entre cada recarga. Dado que el chip GPS que procesa las señales de localización tiene un elevado consumo, el objetivo primordial, una vez probado que el tiempo de respuesta del sistema era el adecuado, consistía en reducir el consumo del chip en situaciones en las que no permitía recuperar la información adecuada, es decir cuando éste estaba en interiores. Esto lo abordamos estudiando el comportamiento de los usuarios en su vida diaria y buscando otras tecnologías presentes en los móviles que permitiesen monitorizar la actividad del dispositivo para activar el chip GPS en la salida a exteriores. Tras analizar el consumo de diversos sensores, nos decidimos por el de acelerometría que permite detectar situaciones de movimientos bruscos y por tanto de transiciones a pie hacia el exterior. Esto supuso una mejora cercana al 300 % con respecto al uso ininterrumpido del chip GPS y además evitó la interacción del usuario con el terminal para activar y desactivar el sistema GPS.

En el Capítulo 7 estudiaremos diversas aplicaciones diseñadas para utilizar las predicciones realizadas por nuestro sistema en un terminal móvil y mostrar las posibilidades que abre este trabajo.

CAPÍTULO 7

APLICACIONES

En este capítulo revisaremos las aplicaciones de los sistemas de predicción de destinos y explicaremos nuestras propuestas.

7.1. Aplicaciones existentes

Tras estudiar las diferentes aplicaciones del estado del arte, las clasificamos en cuatro tipos:

Conducción de vehículos. Podemos distinguir investigaciones que buscan la ayuda al conductor o a la gestión de los motores del vehículo. Entre los primeros, destacan los trabajos de la Universidad de Osaka junto a la de Kobe y Mitsubishi Electric [TMK+06, TKTN09] y los de Motorola [TZL+07, ZLTG07, ZTL+07]. En los primeros desarrollan un sistema de navegación que predice el destino de los conductores indicando mediante interfaces amigables información sobre atascos en su ruta actual. En los segundos plantean un sistema de aviso automático de atascos basado en la ruta predicha.

En los trabajos de John Krumm [Kru08, Kru09] se proponen otro tipo de

ayudas al conductor como la activación automática de intermitentes al predecir giros o incluso a la mejora del rendimiento de los motores de coches híbridos [Deg04] donde proponen un sistema de control conectado a la información de navegación de la ruta futura. El sistema controla la carga o descarga de la batería de acuerdo a las condiciones de tráfico e inclinación de la ruta: sabiendo el perfil de alturas que se realizará, se regulan los momentos de carga en cuestas hacia abajo de los motores.

Pérdidas y reorientación: Patterson implementa un sistema de ayuda a personas con problemas mentales en "Opportunity Knocks" [PLG+04]. Su objetivo principal es ayudar a los usuarios a encontrar su destino en caso de que se sientan desorientados. Tras predecir el destino del usuario por su trayecto, detecta posibles equivocaciones de recorridos utilizando autobuses y lo reorienta para encontrar el lugar final dándole instrucciones de donde debe bajarse y coger el próximo autobús. Cabe destacar que el prototipo propuesto de tiempo real tenía una duración de cuatro horas por lo que recuperaron recorridos de un usuario durante treinta días de manera *off-line* para posteriormente con el modelo predictivo propuesto realizar las pruebas *on-line*.

Comunicación: Aunque hay pocos trabajos referentes a comunicación, en su tesis Marmasse [Mar04] desarrolla un sistema de comunicación visual a través de iconos entre los miembros de una red familiar y de amistades. Aunque el sistema era algo escueto, mostraba iconos de una maleta y una vivienda para indicar que el usuario estaba en el trabajo o en su vivienda, las posibilidades de ampliación son muchas como la propuesta en el escenario inicial (Sección 1.2).

Servicios basados en localización predictivos: Los conocidos como servicios basados en localización pueden ampliarse utilizando las técnicas de predicción. Entre éstos encontramos diferentes propuestas aunque ningún desarrollo real. Sobresale por original una lista de tareas geo-posicionada predictiva. Ésta introducida por Marmasse [MS00] y transformada a predictiva por Ashbrook [AS03] permitiría recordar tareas pendientes asociadas a un lugar. En vez de

emitir los avisos una vez que el usuario esté en él, éstos se lanzarían una vez que se prediga el destino. Esto permitiría por ejemplo que si tenemos que devolver un libro en la biblioteca y vamos a pasar cerca, el sistema emitiría un recordatorio en cuanto se predijese el destino (idealmente antes de comenzar el desplazamiento).

Como vemos el número de aplicaciones reales no es muy elevado y las más estables son las relacionadas con vehículos. La posibilidad de cargar dispositivos móviles a través de las baterías de éstos permite crear aplicaciones de ejecución continua. Nosotros intentaremos aportar algunas ideas y desarrollos así como comentar aspectos interesantes sobre este tipo de aplicaciones.

7.2. Aplicaciones desarrolladas sin predicción

En esta sección comentamos las aplicaciones desarrolladas relacionadas con la generación de recorridos y mapas personales sin utilizar una predicción *on-line*.

7.2.1. Generación de rutas y destinos ficticios

Uno de los mayores problemas de los modelos predictivos es la inutilidad de éstos si partimos desde una base de conocimiento vacía. Aunque con la técnica de similitudes podemos comenzar a tener predicciones válidas a partir del tercer día, normalmente necesitamos un número mayor de días, en torno a los recogidos en un mes, para que el modelo consiga realizar predicciones bastante precisas. Para evitar esa situación y permitir que un usuario consiga resultados desde el primer instante en el que salga a exteriores, realizamos una aplicación Web que permite generar recorridos ficticios indicando simplemente el inicio y fin del trayecto. Podemos ver la pantalla principal en la Figura 7.1. Como se observa utilizamos la API de Google Maps, obteniendo una representación gráfica del recorrido propuesto por el sistema de planificación de Google. Aunque la representación de la ruta para Google Maps

Figura 7.1: Aplicación Web para generar recorridos ficticios.

se compone de los puntos iniciales y finales de cada calle que se atraviesa, nosotros generamos puntos intermedios cada 40 metros. Aunque inicialmente puede que la ruta propuesta no se adapte a la que el usuario haga realmente, damos la posibilidad al usuario de que mueva a su voluntad sus puntos para adaptarla según le interese. Una vez que ésta está preparada, el sistema guarda las coordenadas generadas como si fuesen las de una ruta. Además, también almacena en el sistema los orígenes y destinos introducidos así como sus 'alias', de modo que se está realizando un etiquetado manual que transforma coordenadas o nombres de calles en lugares con significado semántico para el usuario. Aunque siempre sería recomendable utilizar los

recorridos reales en vez de los generados por Google Maps y editados por el usuario, la aplicación permite extraer la información fundamental para un funcionamiento aceptable en menos de media hora tras sincronizar el dispositivo con la máquina en la que se han hecho las consultas, mientras que la otra opción sería la de registrar todos los recorridos durante cerca de un mes sin obtener buenos resultados predictivos.

Otra utilidad importante que se consigue con esta aplicación es la de generar un modelo "semi-abierto". Si recordamos, en el trabajo de Krumm "Predestination" [Kru06, KH06], se obtenía un modelo que permitía predicciones sobre lugares a los que nunca se había dirigido el usuario dentro de la ciudad de Seattle. Gracias a la información del tipo de suelo, la intuición de conducción eficiente y la distribución de la duración media de los viajes, ellos conseguían obtener los destinos con un error medio de 10 kilómetros a falta de la mitad del trayecto. Nosotros consideramos que es posible conocer la localización de un destino no visitado previamente y una aproximación de la ruta que seguiremos si analizamos el comportamiento del usuario cuando debe informarse de cómo llegar a dicho destino. Tras realizar una encuesta entre 30 voluntarios que consideramos como usuarios potenciales de este tipo de aplicaciones (entre 25 y 35 años con estudios de ingeniería informática, telecomunicaciones, electrónica, arquitectura o industriales) obtuvimos los resultados que se muestran en la Figura 7.2. De los resultados se extrae que la búsqueda de informa-

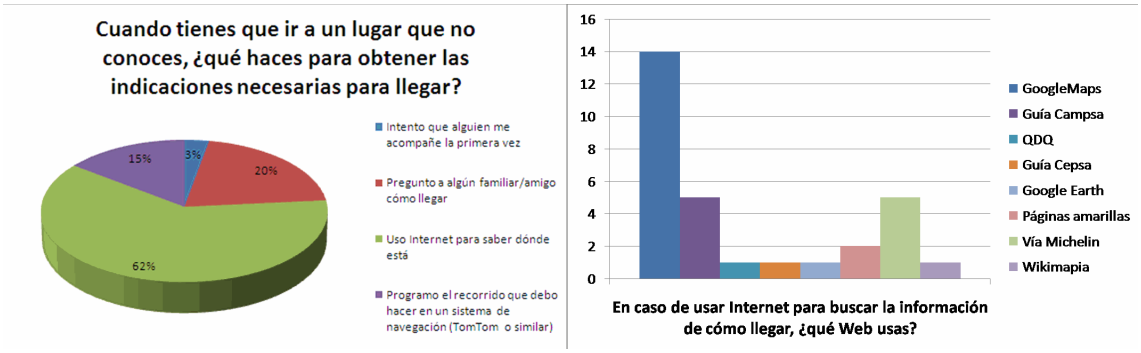

Figura 7.2: Resultado de preguntas sobre la búsqueda de indicaciones para llegar a un lugar desconocido.

ción de indicaciones se realiza de manera mayoritaria en Internet y que Google Maps

es la interfaz más utilizada. De lo anterior, se puede concluir que la sincronización de los resultados de nuestra aplicación con el dispositivo móvil permitiría generar rutas y destinos no conocidos previamente de modo que tendríamos datos sobre los que emitir predicciones sin haber realizado físicamente el recorrido, generando así un modelo "semi-abierto".

7.2.2. Generación de lugares de encuentro

La experiencia en la generación de los mapa soporte, nos permitió darnos cuenta de las posibilidades de combinar más de un mapa personal para encontrar zonas de encuentro entre dos o varios usuarios. Al contrario que cuando generamos los puntos soporte, en vez de encontrar cruces entre rutas, nos interesaba obtener zonas comunes por las que pasasen diferentes usuarios que no se conociesen lo suficiente como para tener un lugar de reunión o que el motivo del encuentro fuese tan urgente que interesase quedar en un lugar conocido por todos y que redujese la distancia de trayecto. Las restricciones podían ser muy variadas: mínimo desplazamiento de todos los participantes, o de un conjunto de ellos, desplazamiento antes de una determinada hora del día (de manera que se eliminen las rutas que se realicen normalmente después de esa hora), etc.. Desarrollamos un prototipo básico que permite indicar el punto de encuentro más cercano entre dos o más usuarios de los que almacenamos su información en la fase de recuperación de la información. Aunque puede resultar bastante útil, este tipo de aplicaciones suponen un servidor que comparte los mapas personales de diferentes usuarios y permite realizar cálculos espaciales con ellos por lo que habría que tomar medidas de seguridad importantes.

7.2.3. Generación de diarios automáticos personales

Aunque Rekimoto [RMI07] realiza una propuesta muy parecida, nosotros podemos crear un diario de desplazamientos *on-line* de modo que al terminar un recorrido incluimos el destino alcanzado como una anotación textual en el diario. Un ejemplo

de este tipo de diario es el mostrado en la Tabla 7.2.3.

Tabla 7.1: Diario automático

Salida	Desde	Llegada	Hasta
08:14 27-03-2008	Casa	08:25 27-03-2008	Despacho
14:05 27-03-2008	Despacho	14:30 27-03-2008	Casa
17:09 27-03-2008	Casa	17:29 27-03-2008	Despacho
19:40 27-03-2008	Despacho	20:14 27-03-2008	Casa padres
21:02 27-03-2008	Casa padres	21:12 27-03-2008	Cine
00:02 28-03-2008	Cine	00:21 28-03-2008	Casa

Sobre este diario personal estamos desarrollando una aplicación para entrenar la memoria. En ella, está previsto que se hagan dos tipos de preguntas al usuario del terminal: lugar en el que estaba a una hora determinada un día particular o fecha de última visita a un determinado lugar. Además se utilizará la frecuencia de los destinos visitados para buscar trayectos y estancias relevantes (suponemos que es más interesante preguntar en qué lugar estaba el usuario una tarde de vacaciones que un lunes laborable a las cinco de la mañana).

7.3. Aplicaciones de predicción

Aunque en las anteriores aplicaciones no era necesario una predicción *on-line*, las que describimos a partir de ahora sí que hacen uso de ella.

7.3.1. Consultas anticipadas sobre medios de transporte y tráfico

Uno de los prototipos que se ha desarrollado para demostrar las posibilidades de la predicción de destinos tiene como objetivo mejorar el uso del servicio de bicicletas

compartidas Sevici[1] de Sevilla. Este servicio permite a los abonados (cuota anual de 13 euros en el año 2009) utilizar de manera gratuita las bicicletas de sus estaciones durante media hora. De ese modo, si un cliente coge una bicicleta y la devuelve en otra estación en ese intervalo de tiempo, el servicio es gratuito. Uno de los mayores problemas de este útil servicio de bajo coste es la poca disponibilidad de huecos para poder aparcar la bicicleta en horas punta en lugares concurridos. Un ejemplo típico de este problema se produce al intentar estacionar una bicicleta a partir de las 9 de la mañana en las estaciones cercanas a la Universidad, produciéndose colas de espera a que alguien saque una bicicleta o búsquedas infructuosas de huecos por las estaciones cercanas. Para evitar estos problemas, nuestra aplicación, una vez detectado el medio de transporte, el destino y la proximidad a éste (menos de 500 metros), consulta dos direcciones web mediante la conexión 3G del dispositivo. La primera[2] indica la posición de cada una de las estaciones de Sevici. La segunda[3] el número de bicicletas y huecos disponibles de cada estación. El aviso se realiza mediante vibración y se muestra una pantalla con las cinco estaciones más próximas al destino con disponibilidad para dejar el vehículo.

Al ser las estaciones bastante dinámicas en cuanto a la entrada y salida de bicicletas, no interesa hacer una predicción muy temprana ya que variaría al llegar, por eso se escogió una distancia de 500 metros que se puede recorrer en menos de dos minutos a una velocidad media de 15 kms/h. De ese modo si es necesario, el ciclista puede variar su dirección para acercarse más rápidamente a la que tenga más huecos. Al estar tan cerca del destino las predicciones tienen un índice de acierto muy elevado y la aplicación resulta bastante útil. El punto débil y donde estamos trabajando es en el modo de comunicación, pretendemos que si el ciclista está en movimiento y lleva conectado los auriculares a su dispositivo, el aviso sea sonoro mediante un mensaje leído. El contenido de éste debe ser lo más orientativo posible en cuanto a la localización de las estaciones. Por otra parte, si detectamos que el usuario está parado (en un semáforo por ejemplo), el aviso se debería hacer mediante

[1] www.sevici.es

[2] www.sevici.es/services/carto

[3] www.sevici.es/services/stationdetails/XX donde XX es el número de estación.

la interfaz visual.

Aunque el anterior prototipo se desarrolló en Windows Mobile, también hemos trabajado en otra aplicación que permite consultas geo-posicionada (utilizando la ubicación del usuario) sobre las estaciones más cercanas con disponibilidad de bicicletas. Ésta se desarrolló como un proyecto de fin de carrera en un móvil IPhone 3G de la que pueden verse una serie de pantallas en la Figura 7.3.

Figura 7.3: Aplicación de consulta interactiva de estaciones de Sevici.

Las aplicaciones con otros servicios de transporte como autobuses o trenes también son directas, siempre que ofrezcan información a través de la Web de tiempos de llegada y localización de las estaciones (en el caso de los autobuses urbanos existen miles de paradas y estamos trabajando en su geo-localización para varias ciudades). Lo que sí observamos es que aunque las técnicas de predicción propuestas son globales, los servicios que se pueden ofrecer son locales normalmente a nivel de ciudades y que el nivel de desarrollo de éstas influye enormemente ya que no es lo mismo

disponer de una gran red de transporte accesible de manera ubicua[4] que si esa información es solamente accesible a través de los puntos de información físicos de sus estaciones.

En el caso del tráfico también podemos obtener datos en tiempo real si lo suministra la entidad que gestiona el tráfico. En el caso de España, la DGT[5] proporciona datos relativamente dispersos sobre los problemas relativos a tramos de carreteras en obras y accidentes. Dentro de cada una de las grandes ciudades españolas encontramos información más precisa sobre el estado del tráfico por las vías principales de una ciudad. En el caso de Sevilla, se utiliza la información de las cámaras de tráfico y las espiras electromagnéticas situadas bajo el asfalto en gran cantidad de puntos de la ciudad para generar información *on-line* en su Web[6]. Aunque la información está preparada para que el usuario acceda a ella a través de un navegador, no resulta complejo parsearla desde un terminal móvil y obtener únicamente la información relativa a las calles que se atravesarán por la ruta que se ha detectado que se va a realizar. Si alguna de las calles que atraviesa el usuario normalmente tiene tráfico intenso, podría permitir al usuario utilizar otra ruta alternativa o incluso cambiar de medio de transporte. El problema que nos hemos encontrado es que a veces el usuario atraviesa calles no monitorizadas por lo que la información puede ser de escaso valor. Otras veces, el usuario atraviesa sólo pequeños tramos de una calle con información de tráfico por lo que no podemos prever la influencia de ésta en la duración de sus trayectos.

Observamos que la tendencia de la obtención de datos de tráfico ya no son las cámaras y espiras que ofrecen información a través de la Web (infraestructura fija y cara pero accesible por todos), sino las flotas de vehículos que tienen un determinado sistema de navegación personal instalado (infraestructura móvil, barata pero inaccesible para los que no pagan el servicio) y que refrescan en tiempo real una base de datos central con los datos de velocidad con la que atraviesan las calles.

[4] La información ubicua ha terminado siendo un sinónimo de información accesible a través de internet.

[5] http://www.dgt.es

[6] http://www.trajano.com

7.3.2. Consultas anticipadas sobre personas

Al igual que puede consultarse el estado de la infraestructura que se piensa atravesar o acceder con suficiente antelación, se puede pensar en consultas sobre personas geo-localizadas. Ya existen redes sociales[7] en las que se puede especificar donde se encuentran una serie de amigos. Sin embargo, pensamos que este tipo de información es muy delicada y exponerla en un mapa supone peligros de privacidad y uso indebido. Por ello proponemos un modelo de red geo-posicionada por descripciones textuales, es decir, en vez de indicar en un mapa la posición exacta de un usuario, ésta se indicará mediante una etiqueta en la que se muestre la descripción que ha asociado ese usuario al lugar en el que se encuentra. Esto provoca que una vez detectado un nuevo destino, se solicite al usuario una descripción textual por ejemplo "Casa", "despacho", "el cine de siempre", "restaurante donde vamos los sábados" etc., que no desvelan la posición exacta del usuario para personas que no sean del mismo círculo social o familiar.

Estas descripciones nos permiten generar nuevas explicaciones bastante esclarecedoras de manera automática cuando el usuario está desplazándose utilizando la predicción de destinos. Un ejemplo de este tipo de indicaciones automáticas puede ser el siguiente mensaje: "Voy **andando** hacia **el cine de siempre** y me quedan **50 metros** para llegar".

Para probar la utilidad de la propuesta, se implementó un emisor/receptor de mensajes cortos en Windows Mobile que permite identificar un mensaje corto con un formato determinado (la primera y única palabra del mensaje debe ser "`__DndStas`"), comprobar que el número de teléfono del emisor está en la lista de miembros de la red familiar del receptor (que previamente ha configurado) y si todo es correcto, generar un mensaje corto respondiendo al receptor con dos posibles formatos, el que hemos puesto más arriba si el usuario se está desplazando o uno que simplemente indica donde se encuentra éste si no hay desplazamiento alguno, siendo las palabras

[7] Whrrl o Latitude permiten buscar amistades por lugares de interés o conocer en tiempo real la localización de esas amistades.

en negrita las que debe completar el sistema utilizando el sistema GPS y la predicción de destinos. Cabe destacar que se incluyeron tres etiquetas de desplazamiento ("andando", "en coche" y "en bicicleta") que se asignaban según la velocidad máxima y media del recorrido desde su inicio.

El medio de comunicación elegido fueron los mensajes cortos de móvil que permiten asegurar una identificación única y una respuesta al usuario adecuado.

De ese modo, podemos hacer realidad parte del escenario que comentamos en la introducción, en el que un matrimonio podía tomar diferentes decisiones según hacia donde se estuviese dirigiendo su pareja.

7.3.3. Detección de pérdidas

Una última aplicación desarrollada es la de detección de pérdidas para personas con problemas de orientación. Aunque inicialmente se consideró trabajar con personas en etapas tempranas de la enfermedad de Alzheimer, se descartó tras mantener una entrevista con el responsable de "Alzheimer Sevilla"[8] una asociación que se dedica al cuidado y seguimiento de este tipo de personas, en la que nos indicaron que los familiares de estas personas no solían colocar dispositivos de seguimiento, sino que directamente evitaban que saliesen a la calle solas si ya se había producido una pérdida. De ese modo aunque podía utilizarse con personas que tuviesen esta enfermedad en una fase inicial, lo aconsejable era tener como público objetivo personas con dificultades en la orientación como ancianos o niños.

Aunque ya existen algunas herramientas que gestionan las pérdidas[9],[10], éstas actúan definiendo zonas de seguridad estáticas. En nuestro caso, tras ser entrenada, la aplicación detectaba la ruta y el destino y establecía una zona de seguridad alrededor de esa ruta de modo que en caso de que atravesase esa zona, se enviaba un SMS con un enlace a Google Maps con la localización exacta de la persona

[8] http://www.alzheimersevilla.com/
[9] http://www.simapglobal.com
[10] http://www.keruve.com

con necesidad de seguimiento. La posibilidad de utilizar el modelo incremental de entrenamiento por similitudes permite más flexibilidad al usuario que lo lleva.

7.3.4. Otras aplicaciones propuestas

Aunque no se llegaron a llevar a la práctica se plantearon las siguientes aplicaciones:

Intercambio de mercancías monitorizada. Para mejorar el la información sobre intercambio de mercancías entre empresas suministradoras y consumidoras, de modo que la flota de camiones de las primeras, fuese enviando la predicción de llegada a las segundas de manera autónoma y con estimaciones de tiempo de llegada para mejorar la planificación e información.

Acciones a distancia automáticas. Las posibilidades de control remoto y de acceso a elementos físicos dentro de espacios asistidos y domotizados también permitiría que el terminal ofreciese la posibilidad de encender el aire acondicionado del lugar al que nos dirigimos, o de enviar mensajes de retardo en caso de que no lleguemos a una determinada cita.

Consultas sobre lugares de interés. Al avance que supone disponer de los datos de transporte y tráfico en tiempo real, se une el trabajo realizado por empresas de cartografía que recogen buscadores como Google y Yahoo y hacen accesibles a través de sus mapas en la Web. De ese modo, se pueden ampliar las aplicaciones que utilicen la predicción de destinos a búsqueda de lugares de interés cercanos a la ruta que vamos a seguir. Combinando información que manejamos diariamente con los sistemas de predicción podríamos conseguir que se nos indicase los lugares de interés más cercanos a la ruta que seguiremos como una gasolinera si tenemos el depósito del vehículo vacío, los supermercados si tenemos que hacer la compra, los cajeros de nuestra entidad bancaria si tenemos que sacar dinero, etc.. Otra de las fuentes de información importantes son los portales Web con objetivos concretos por ejemplo de venta o alquiler

de inmuebles que en caso de estar buscando vivienda puede ser muy útil para visitar pisos y casas cercanas a los trayectos habituales.

7.4. Resumen

En este capítulo hemos estudiado y clasificado las aplicaciones existentes con conocimiento de la localización futura para posteriormente detallar nuestros desarrollos. Las aplicaciones las hemos dividido en las que hacen uso de los datos provenientes del conocimiento extraído en el proceso de recuperación de la información y aquellos que utilizan el sistema de predicción.

Entre las primeros, destaca la aplicación Web que complementa al sistema del terminal móvil al sincronizar éste con el equipo dónde se realiza una consulta de rutas a través de la API de Google Maps. Esto permite que incluyamos recorridos virtuales en nuestra base de datos y por tanto extraigamos conocimiento como la nueva ruta o el nuevo destino. Además generamos un diario personal en tiempo real que consideramos bastante novedoso (otros autores lo consiguen de manera off-line).

Entre las aplicaciones que usan el sistema de predicción incluimos una que busca la ayuda para ciclistas que utilicen sistemas de bicicletas compartidas (existentes en diferentes ciudades europeas), ésta permite la consulta sobre los huecos para aparcar la bicicleta en las estaciones más cercanas al destino predicho por el sistema. Además también incluimos un sistema de consulta sobre la posición de los miembros de una comunidad de una manera segura y respetando la privacidad. Aunque inicialmente se pensó en un sistema de comunicación familiar, hemos ampliado la idea para formar una red social geo-posicionada segura.

Tras ver algunas aplicaciones de nuestro sistema, pasamos a enumerar las aportaciones y líneas de trabajo futuro en el Capítulo 8.

CAPÍTULO 8

CONCLUSIONES Y TRABAJO FUTURO

En esta tesis hemos abordado una línea de investigación relativamente nueva, nacida en 2002: la predicción de destinos en desplazamientos basándonos en un conjunto de recorridos pasados. Ante este reto analizamos las propuestas existentes y detectamos una serie de carencias que hemos intentado subsanar. En este capítulo, enumeraremos los objetivos conseguidos y las principales aportaciones de esta tesis, así como las futuras líneas de trabajo abiertas.

8.1. Principales aportaciones

Aunque el objetivo final era desarrollar un sistema que permitiese predecir destinos *on-line*, en primer lugar realizamos un estudio completo de los procesos existentes para el análisis y extracción de información a partir de los ficheros log generados por diferentes usuarios. Para ello definimos una metodología *off-line* que nos permitió extraer conclusiones tanto a nivel práctico como teórico. Además está orientada a todo tipo de recorridos desde los realizados en vehículos como los hechos a pie y permite la aplicación de diferentes técnicas de predicción.

Esta metodología encapsula los procesos de filtrado, extracción de destinos y extracción de recorridos o segmentado, de modo que obtenemos como producto intermedio un conjunto de recorridos y destinos que servirán de entrada para los algoritmos de predicción.

En esta primera parte de extracción de información hemos aportado dos nuevas técnicas:

Recuperación de destinos frecuentes. Esta técnica nos permite recuperar los destinos frecuentes mediante un cluster jerárquico con corte por distancia y número mínimo de elementos ayudando a reducir el número de falsos lugares con respecto a otras técnicas propuestas anteriormente.

Segmentación de recorridos por distancia y tiempo. Además de aplicar el segmentado temporal utilizado por diferentes autores, desarrollamos un algoritmo que permite detectar paradas cortas pero significativas para los usuarios, aumentando en más de un 13 % de media los nuevos recorridos obtenidos.

Una vez obtenidos los recorridos, generamos mapas personales que permitiesen que el sistema fuese escalable, de modo que aunque fuese utilizado durante años, el almacenamiento de la información de desplazamientos en exteriores permitiese almacenarse en relativamente poco espacio. Para ello realizamos dos aproximaciones. La primera basada en representaciones atemporales de los recorridos a las que llamamos rutas. La segunda, basada en cruces y bifurcaciones de rutas a la que llamamos mapas soporte.

Para generar las rutas, fue necesario agrupar los recorridos por proximidad por lo que propusimos varias medidas de similitud entre recorridos de las que finalmente utilizamos la similitud Jaccard-GPS. Su comportamiento demostró ser más flexible y robusto ante pequeñas desviaciones entre recorridos que otras aproximaciones. Esto permitió que redujésemos en aproximadamente un 60 % el número de recorridos totales.

La parte más novedosa la introducimos en la forma de generar el mapa soporte,

compuesto por puntos significativos sin tener una representación de la red de calles y carreteras que utilizan otros autores a través de su GIS local. Esto nos permitió conseguir un mapa soporte adaptado a las zonas recorridas por el usuario, consiguiendo una reducción superior al 95 % de puntos. Aunque la reducción es superior a las rutas, el aspecto crítico para que el mapa contenga puntos significativamente importantes para el usuario es la densidad de cruces entre las rutas de ese usuario.

A nivel de predicción, desarrollamos dos alternativas interesantes. La primera basada en mapas soporte y HMM resultó ser una aproximación bastante fiable para los usuarios que tenían un comportamiento repetitivo en cuanto a sus desplazamientos diarios, sin embargo se comportó algo peor con usuarios más variables.

En la segunda alternativa en vez de usar puntos significativos, se usaron directamente las rutas y una nueva similitud ponderada que llamamos similitud deslizante que permite comparar un recorrido no finalizado con una ruta. A esta aproximación la llamamos predicción por similitudes. Esto permitió mejorar en la mayoría de los casos el porcentaje de éxito de predicción con respecto a la anterior técnica de predicción. Aunque no es posible comparar los resultados de esta técnica con respecto a los de otros autores (ya que utilizan GIS), conseguimos que de media al recorrer el 10 % de un recorrido, se consiguiese un porcentaje de éxito en la detección de la ruta y el destino final superior al 50 %.

Una vez evaluadas ambas técnicas, resultó que la de similitudes era mucho más rápida y adecuada para utilizarse en tiempo real por lo que se implementó en un dispositivo móvil. Aunque los resultados eran los esperados, nos encontramos con el problema de la duración de las baterías. Éste se solventó utilizando un dispositivo móvil que integraba además de un chip receptor GPS un acelerómetro que permitió detectar situaciones de movimiento que permitían parar el chip GPS cuando el dispositivo permanecía quieto y reactivarlo cuando se identificaban patrones de desplazamiento continuados. Además de conseguir un elevado porcentaje de aumento de la vida del dispositivo, se adaptó la metodología *off-line* para aplicarla *on-line*.

El enfoque de esta metodología de trabajo ha sido novedoso por el estudio de

técnicas de predicción de destinos sin utilizar ningún tipo de GIS. La potencia que aportan los GIS tienen el inconveniente de que evitan que el sistema sea global y haya que adaptarlo al tipo de GIS existente para una zona o zonas concretas. Nosotros evitamos esa dependencia y utilizamos datos generados por el propio usuario estudiado para generar mapas soporte y rutas personales. Esta diferencia fundamental hizo que nuestros resultados no fuesen comparables a los obtenidos por otros autores.

Por último una vez que el enfoque *on-line* se completó, se desarrollaron varios prototipos novedosos que utilizaban la información de las predicciones:

Ayudante para ciclistas de Sevici. Esta herramienta permite acceder rápidamente a estaciones de Sevici con huecos libres.

Avisos automáticos de tráfico para conductores. Esta aplicación permite advertir sin necesidad de interacción a conductores de la situación del tráfico de la rutas que suelen seguir cuando comienzan a recorrerlas.

Sistema de comunicación por SMS. Hacía sencillo y seguro las consultas sobre la situación geográfica de personas dentro de un círculo social determinado previamente. La predicción permitía responder de manera autónoma mensajes cortos en caso de que el usuario consultado estuviese desplazándose. De esa manera no era molestado teniendo que consultar su terminal mientras por ejemplo conducía un coche.

Detección de pérdidas. Se desarrolló un prototipo que permitía detectar salidas de una zona de seguridad delimitada para una ruta determinada.

8.2. Líneas de trabajo futuro

Predicciones previas al desplazamiento. La detección de comienzos de desplazamientos puede suponer un buen momento para avisar al usuario de tareas

pendientes, por lo que podríamos combinar trabajos previos como el de Ashbrook [AS03] para hacer predicciones anteriores al desplazamiento.

Uso de información sobre el medio de transporte. La detección del medio de transporte utilizado puede eliminar una serie de destinos de todos los posibles según la distancia. No sería lógico considerar como posible destino uno que diste más de 200 kilómetros si nos desplazamos en bicicleta o uno que diste más de 20 kilómetros si nos desplazamos a pie.

Uso de información personal del dispositivo. Los dispositivos móviles cada vez condensan más información personal como la agenda, el calendario, la lista de tareas, etc.. Podríamos utilizar los datos del calendario por ejemplo para permitir asociar esos eventos con destinos futuros.

Uso de información temporal. Aunque no entró en nuestra hipótesis inicial, el uso de esta información mejoraría en muchos casos la predicción debido al carácter rutinario tanto en espacio como en tiempo de la mayoría de nuestras acciones.

Al trabajar sólo con información espacial, los resultados indican el mejor destino sin importar el instante en que se realiza el recorrido. Es decir no importa si el recorrido que se está prediciendo se realiza un lunes por la mañana o un sábado por la tarde.

Interfaces de usuario. Aunque no se ha tratado la manera de suministrar la información al usuario, resulta crucial el modo de comunicar avisos o pedir su interacción cuando las predicciones se realizan en movimiento. Por ejemplo, a un ciclista le puede resultar bastante complejo coger su dispositivo móvil y leer un mensaje de texto mientras pedalea, sin embargo a un peatón sí que podría hacerlo sin problemas. La detección del medio de transporte o si el terminal tiene conectado auriculares serían datos importantes tanto para realizar el aviso mediante las diversas alternativas (sonora, visual o vibración) como para comunicar la información (mensaje textual, mensaje leído o incluso comunicación por vibración como si fuera un lenguaje Morse).

Mejoras en la detección de movimientos. En la detección de desplazamientos a pie, comprobamos que movimientos de un usuario en una vivienda con el móvil en el bolsillo (por ejemplo poniendo la mesa, tendiendo, etc.) podían generar falsos positivos y provocar la activación innecesaria del chip GPS del terminal. Para evitar esos errores, estamos estudiando el uso de huellas WiFi que permitirían marcar la entrada en edificios para que al detectar que el usuario ande, activar primero la radio WiFi y si se encuentra una huella parecida a la de la entrada, activar el GPS.

APÉNDICE A

SISTEMA GPS

En este anexo describiremos brevemente el sistema GPS, su funcionamiento, así como las sentencias de control NMEA utilizadas para gestionar la comunicación con otros dispositivos.

A.1. Funcionamiento

El sistema GPS está formado por tres segmentos o áreas:

Espacial. Engloba los satélites del sistema: la constelación NAVSTAR (*Navigation Satellite Timing and Ranging*), formada por 24 satélites activos distribuidos en seis planos diferentes tal y como se puede ver en la Figura A.1 más seis de reserva. Está mantenida por el gobierno estadounidense que periódicamente los repara o sustituye (el último se lanzó en abril de 2009).

Control. Está constituido por cinco estaciones terrestres y tres antenas con coordenadas bien conocidas en un sistema terrestre de referencia internacionalmente aceptado. Su misión es la de rastrear a todos los satélites para calcular las órbitas (efemérides) y controlar sus relojes. La información sobre efemérides y

relojes son periódicamente transmitidas en forma de mensajes de navegación a los satélites desde las antenas en la Tierra, para su transmisión posterior desde los satélites a los usuarios.

Usuario. Corresponde con los equipos receptores. Estos equipos constan básicamente de una antena, un receptor, capacidad para procesamiento de señales y almacenamiento de datos. La señal de radio transmitida por cada satélite es recepcionada por el equipo conociendo el código de la señal PRN (ruido pseudoaleatorio), obteniendo de esta manera la información de la pseudodistancia y detectando el mensaje de navegación.

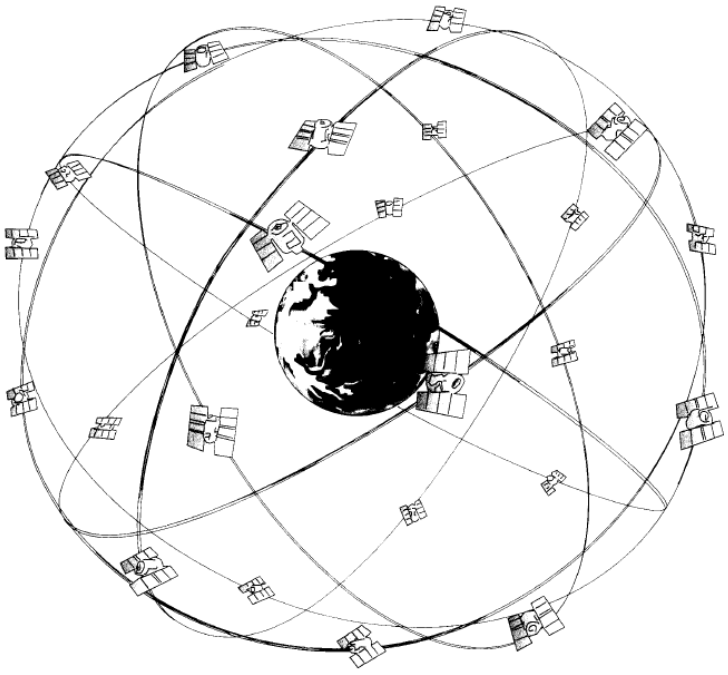

Figura A.1: Distribución de los 24 satélites activos de NAVSTAR.

Para conseguir un posicionamiento correcto, cada satélite contiene relojes atómicos (de Cesio o Rubidio) de gran precisión sincronizado con todos los demás. La frecuencia de estos relojes es de 10.23 MHz. Cada satélite suministra dos señales portadoras: La L1 a 1575.42 MHz y la L2 a 1227.6 MHz (multiplicando la salida del reloj atómico por 154 y 128 respectivamente). Estas señales portadoras, utilizando una modulación CDMA (multiplexación por división de código) se combinan

con un código de precisión llamado código PRN. Estas señales son recuperadas por el receptor GPS que también contiene un reloj, en este caso de cuarzo, de mucha menos precisión y coste que los anteriores. Los receptores usan la interferometría, retrasando una replica del código PRN de los satélites almacenados en la memoria y comparándolo con el código de entrada. En una sincronización precisa, el código desaparece dejando sólo la onda portadora. De ese modo, el receptor puede medir la diferencia de tiempo entre el envío y la recepción. Este proceso puede observarse en la Figura **??** extraída del artículo de uno de los diseñadores del sistema (Ivan Getting [Get93]). Multiplicando esa diferencia de tiempo por la velocidad de la luz

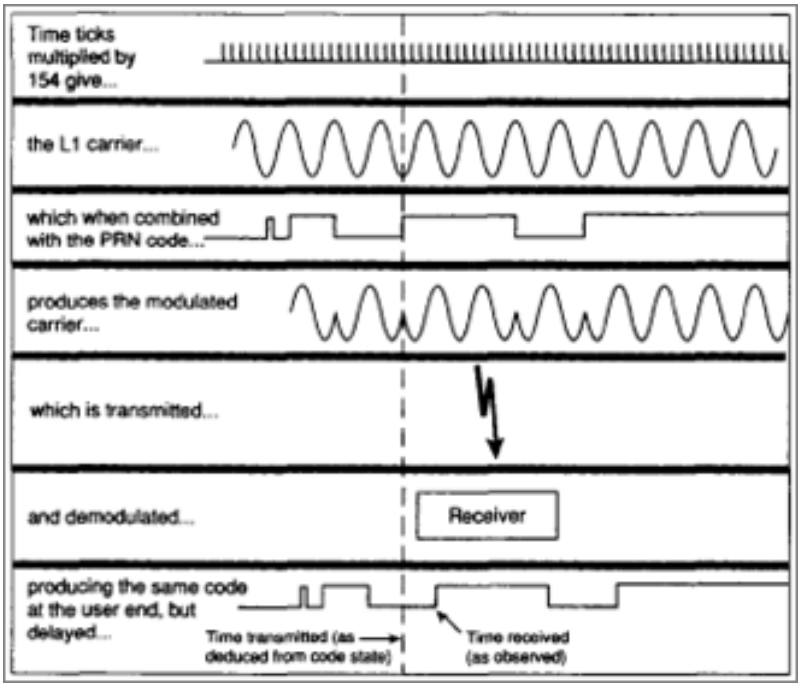

Figura A.2: Proceso de recuperación del intervalo de tiempo entre el envío de la señal desde un satélite GPS y la recepción en tierra.

se obtiene la distancia entre el receptor y el satélite. A esta distancia se le conoce como medición de pseudodistancia ya que tiene un margen de error, principalmente debido al error en la sincronización del reloj del receptor (el de cuarzo).

Para obtener su localización exacta, el receptor debe resolver un sistema de cuatro ecuaciones (basadas en las pseudodistancias recuperadas de cuatro satélites) con

cuatro incógnitas: latitud, longitud, altura y error en la precisión del reloj de cuarzo con respecto a los de satelitales. Si el receptor recibe señales de al menos cuatro satélites, este sistema podrá resolverse[1]. La solución de esta ecuación permite obtener la posición del receptor GPS con una alta precisión (dependiendo del tipo de receptor puede ser del orden de centímetros hasta de varios metros).

A.2. Sentencias NMEA

Una vez se resuelve el sistema de ecuaciones comentado, el receptor produce una secuencia de sentencias NMEA 0183[2] que permite comunicar la información de posicionamiento con otros dispositivos a través de bluetooth (receptores GPS dedicados) o en caso de estar integrado en otro dispositivo mediante señales eléctricas (como en los terminales móviles que integra chips GPS). Un ejemplo de estas secuencias puede verse en la Tabla A.1.

Tabla A.1: Extracto de sentencias NMEA.

$GPGSA,A,3,24,08,02,10,04,27,,,,,,,3.1,1.6,2.6*3F
$GPGLL,3722.7295,N,00558.7754,W,094452.504,A*2D
$GPGSV,3,1,10,08,66,153,44,27,63,069,39,02,54,238,41,10,43,315,30*7D

Cada tipo de sentencia (identificada por la primera cadena de línea) aporta diferente información. A continuación mostramos las más comunes:

GPRMC. Sentencia que contiene latitud, longitud, velocidad, orientación, fecha, validez del dato.

[1] También puede resolverse un sistema de tres ecuaciones con las señales de tres satélites. En este caso, no se obtiene la altura.

[2] Especificación eléctrica y de datos entre aparatos electrónicos marinos (receptor GPS, sonar, anemómetro, etc.) definida y controlada por la organización estadounidense National Marine Electronics Association.

GPGSA. Informa sobre los satélites sobre los que se ha obtenido su señal y la precisión de la información suministrada de localización. Esta última información llamada *"Dilution of precision"*, indica la pérdida de precisión de la señal y viene suministrado por 3 sentencias: HDOP, VDOP y PDOP. HDOP indica la pérdida de precisión en horizontal (relativa a latitud y longitud), VDOP indica lo mismo en vertical (relativo a la altura) y PDOP indica un índice para las 3 componentes. De ese modo, cuanto más altos sean los valores de HDOP, VDOP o PDOP, menor será la precisión de la medida obtenida.

GPGGA. Indica la latitud, longitud, hora, validez del dato de localización, número de satélites visibles, precisión del dato de posicionamiento horizontal (HDOP) y altura.

GPGSV. Suministra información sobre cada uno de los satélites de los que se capta información. Para cada satélite se incluye el número de identificación, su inclinación, su azimuth y su relación señal-ruido (SNR). Cada sentencia puede almacenar la información de cuatro satélites.

GPGLL. Información sobre la posición. Contiene latitud, longitud, hora y validez del dato.

De ese modo, nuestro trabajo parte de la interpretación de las sentencias NMEA dado que el proceso previo resulta transparente al usuario.

APÉNDICE B

CURRICULUM

B.1. Publicaciones relacionadas

Durante los últimos años hemos desarrollado un conjunto de aportaciones, tanto en el campo de middleware para entornos estructurados y con capacidades de comunicación inalámbricas así como de procesado y almacenamiento, como en la reciente línea de investigación de la predicción de destinos espaciales a través de dispositivos GPS y estudio de recorridos y lugares frecuentes. A continuación detallamos las contribuciones realizadas.

[**2005**] : Durante los años 2004 y 2005 se desarrolló el proyecto Domoweb. El objetivo del mismo era el diseño de una arquitectura que permitiese el control domótico de una vivienda a través de protocolos de comunicación como HTTP, Bluetooth o X-10; y la definición de las metodologías necesarias para poder aplicar los conocimientos adquiridos en los espacios teleasistenciales.

Así mismo comenzamos a analizar las interacciones inter-personales en la teleasistencia, así como las preferencias de los asistidos y las posibilidades que ofrecía la tecnología con la que trabajamos para mejorarlas.

- **Título:** Experiencias en entornos de computación ubicua mediante arquitecturas orientadas a servicios.

  Autores: J. A. Álvarez, M.D. Cruz, A. Fernández, J. A. Ortega y J. Torres.

  Publicado en: Jornadas Científico-Técnicas de Servicios Web, JSWEB 2005 (W3C) durante el I Congreso Español de Informática, Granada. Septiembre 2005. ISBN: 84-9732-455-2.

- **Título:** Soluciones a problemas de comunicación e interacción en entornos de computación ubicua con OSGi.

  Autores: J. A. Álvarez, J. A. Ortega, A. Fernández-Montes y M.D. Cruz.

  Publicado en: Simposio de Computación Ubicua e Inteligencia Ambiental, UCAmI'2005, Granada. Septiembre 2005. ISBN: 84-9732-442-0.

- **Título:** Creación de entornos de teleasistencia mediante computación ubicua.

  Autores: J. A. Álvarez, J. A. Ortega y J. Torres.

  Publicado en: Actas de la Conferencia IADIS Ibero-Americana WWW Internet, Lisboa. Junio 2005. ISBN: 972-8924-03-8

- **Título:** Aplicación del razonamiento cualitativo al hogar digital.

  Autores: J. A. Álvarez, J. A. Ortega, J. Torres, A. Fernández-Montes, M. D. Cruz, C. Angulo y F. Velasco

  Publicado en: Actas de las VII Jornadas de trabajo ARCA, Benalmádena (Málaga). Junio 2005. ISBN: 84-689-3357-0.

- **Título:** CUCA Project: Cooperative system of ubiquitous computing in welfare contexts.

  Autores: J. A. Álvarez, C. Angulo, J. A. Ortega, M. D. Cruz y A. Fernández-Montes.

  Publicado en: Monet Newsletter, April 2005. ISSN: 1464-9276.

[**2006**]: Los trabajos desarrollados durante este año se centran en resolver el problema de la detección de destinos futuros y plantear la metodología off-line correspondiente al Capítulo 2 de este documento.

También se aportan nuevas ideas para comunicar el sistema utilizado para la gestión domótica de los entornos asistenciales con el sistema de predicción, permitiendo nuevas aplicaciones conscientes del contexto futuro.

- **Título:** Extended sensations on interactive telecommunication.

  Autores: J. A. Álvarez, J. A. Ortega, A. Fernández-Montes, M. D. Cruz, y P. Castilla.

  Publicado en: I International Conference on Ubiquitous Computing. June 2006. ISBN: 84-8138-704-5.

- **Título:** Where do we go? OnTheWay: A prediction system for spatial locations.

  Autores: J. A. Álvarez, J. A. Ortega, L. González, F. Velasco y F. J. Cuberos.

  Publicado en: I International Conference on Ubiquitous Computing, Alcalá de Henares, June 2006, ISBN: 84-8138-704-5.

- **Título:** OnTheWay: Sistema de predicción de destinos espaciales.

  Autores: J. A. Álvarez, J. A. Ortega, J. Torres, L. González, F. Velasco y F. J. Cuberos.

  Publicado en: Actas de las VIII Jornadas de trabajo ARCA, ISBN:84-611-1401-9.

- **Título:** OnTheWay: A prediction system for spatial locations.

  Autores: J. A. Álvarez, J. A. Ortega, L. González, F. Velasco y F. J. Cuberos.

  Publicado en: Proceedings of the International Conference on Wireless Information Networks and Systems 2006, ISBN: 972-8865-65-1. Publicado posteriormente en la revista electrónica "CEUR Workshop Proceedings", ISSN: 1613-0073.

[**2007**]**:** Combinando las aportaciones de los años anteriores, comienzan a ofrecerse aplicaciones que hacen uso tanto del control de entornos con infraestructura de Computación Ubicua como del sistema desarrollado de detección de destinos. Entre éstas destacan los soluciones para empresas de transporte.

Además se prepara el proyecto InCare en el que se pretende mejorar los servicios existentes para personas con enfermedades neuro-degenerativas como el Alzheimer. De hecho contactamos con la Asociación "Alzheimer Sevilla".

Asimismo, realizo una estancia muy enriquecedora en la Universidad Politécnica de Cataluña con el grupo GREC-UPC en el Centro de Estudios Tecnológicos para la Dependencia (CETpD) en la que implementamos un sistema de detección de patrones de movimiento a partir de sensores acelerómetros adaptados a una rodillera para personas con problemas de movilidad y participo en un proyecto en el que trabajo con RFID y Text-to-Speech.

- **Título:** Interoperability for transport companies.

  Autores: J. A. Álvarez, J. A. Ortega, L. González, F. Velasco y F. J. Cuberos

  Publicado en: Enterprise Interoperability II. New Challenges and Approaches. 2007 ISBN: 978-1-84628-857-9.

- **Título:** Sistema asistencial experimental de monitorización de movimiento y comportamiento.

  Autores: C. Pérez, C. Angulo, J.A. Álvarez, J.A. Ortega

  Publicado en: 2nd International Symposium on Ubiquitous Computing and Ambient Intelligence - 2007.

- **Título:** InMyOneWay: Un sistema de reorientación y navegación personal para personas con deterioro cognitivo leve y Alzheimer

  Autores: J. A. Álvarez, J. A. Ortega, L. González-, F. Velasco

  Publicado en: 2nd International Symposium on Ubiquitous Computing and Ambient Intelligence - 2007.

- **Título:** Combining smart tags and body fixed sensors for disabled people assistance.

  Autores: J. A. Álvarez, C. Pérez, C. Angulo y J. A. Ortega.

  Publicado en: KES2007 11th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems. Lecture Notes in AI series.

- **Título:** An orientation service for dependent people based on an open service architecture.

  Autores: A. Fernández-Montes, J. A. Álvarez, J. A. Ortega, N. Martínez Madrid, R. Seepold

  Publicado en: HCI and Usability for Medicine and Health Care, Springer LNCS, Volume 4799, ISSN 0302-9743, 2007.

[**2008**]**:** Durante este año dedicamos nuestro esfuerzo a los proyectos nacional InCare y al andaluz CUBICO, donde se enmarca el grueso de nuestra tesis doctoral.

- **Título:** Delivery improvement for transport companies.

  Autores: J. A. Álvarez, A. Fernández-Montes, J. Moreno, J. A. Ortega, L. González y F. Velasco.

  Publicado en: ICCBSS 2008.

- **Título:** Smart environment vectorization. An approach to learning of user lighting preferences.

  Autores: Alejandro Fernández-Montes González, J. A. Ortega Ramirez, L. González , J. A. Álvarez

  Publicado en: Lecture Notes In Computer Science 765-772 2008

- **Título:** Capítulo de libro: A home e-health system for dependent people based on OSGi.

  Autores: J. Martín, R. Seepold, N. Martínez, J. A. Álvarez , A. Fernández-Montes y J. A. Ortega.

Publicado en: Lecture Notes in Electrical Engineering Intelligent Technical Systems 10,1007/978 − 1 − 4020 − 9823 − 9_9 117-130 2009

[**2009**]**:** Finalizamos los desarrollos derivados de esta tesis, concretamente los relacionados con la implantación del sistema de predicción en el terminal móvil. Por ello se envían los trabajos a revistas con índice de impacto importante y esperamos su pronta resolución.

- **Título:** Service-oriented device integration for ubiquitous ambient assisted living environments.

  Autores: J. Andreu, J. A. Álvarez , A. Fernández-Montes y J. A. Ortega.

  Publicado en: Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assited Living ISBN: 978-3-642-02480-1. Congreso incluido en el índice "Computer Science Conference Ranking" con ranking 0.55 en el apartado de "Artificial Intelligence / Machine Learning".

- **Título:** Trip destination prediction based on past GPS log using a Hidden Markov Model.

  Autores: J. A. Álvarez, J. A. Ortega, L. González y F. Velasco.

  En revisión: Expert Systems and Applications (JCR en 2008). Enviado en mayo de 2009.

- **Título:** Trojan horses in mobile devices

  Autores: D. Fuentes, J. A. Álvarez, J. Ortega, L. González y F. Velasco.

  En revisión: Computer Science and Information Systems (Science Citation Index Expanded). Aceptado con revisiones el 2 de diciembre de 2009.

- **Título:** Outdoor exit detection using combined techniques to increase GPS efficiency.

  Autores: J. A. Álvarez, L. M. Soria, J. A. Ortega y L. González.

  En revisión: IEEE Network (JCR en 2008). Enviado en noviembre de 2009.

- **Título:** A study of thematic areas in economy from a measure of similarities based on a kernel.

  Autores: F. Velasco, L. González, J. A. Ortega y J. A. Álvarez.

  En revisión: Interciencia (JCR en 2008). Enviado en marzo de 2009.

- **Título:** A Novel Approach to Trojan Horses Detection in Mobile Devices.

  Autores: D. Fuentes, J. A. Ortega, L. González, F. Velasco.

  En revisión: IEEE Security & Privacy (JCR en 2008). Enviado en septiembre de 2009.

B.2. Proyectos de investigación

Esta tesis doctoral se ha desarrollado en el marco de los siguientes proyectos de investigación:

- **Nombre:** DOMOWEB: metodologías para el diseño y desarrollo de sistemas domóticos controlados vía Web.

  Investigador principal: Juan Antonio Ortega Ramírez.

  Organismo financiador: Junta de Andalucía.

  Período de duración: $2003 - 2005$.

- **Nombre:** E-TAO: Sistema de telemedicina asíncrona basado en estándares médicos para control de pacientes que siguen la terapia de anticoagulante oral.

  Investigador principal: Francisco José Moriana Garia.

  Organismo financiador: Junta de Andalucía.

  Período de duración: 2006.

- **Nombre:** Navegación e Interacción con el Usuario en el Desarrollo de Sistemas de Información Web: Métodos, Técnicas y Herramientas (TIC 2003-369).

Investigador principal: Jesús Torres Valderrama.

Organismo financiador: Ministerio de Ciencia y Tecnología y fondos FEDER.

Período de duración: 2003 – 2006.

- **Nombre:** InCare: Plataforma abierta para la integración en el hogar de servicios cooperativos de teleasistencia y telemedicina (TSI2006-13390-C02-02).

  Investigador principal: Ralf E.D. Seepold.

  Organismo financiador: Ministerio de Educación y Ciencia.

  Período de duración: 2007 – 2009.

- **Nombre:** : Sistema de cuidados ubicuos y asistencia controlado por familiares y centros médicos para personas con dependencias - CUBICO (TIC2141).

  Investigador principal: Juan Antonio Ortega Ramírez.

  Organismo financiador: Junta de Andalucía.

  Período de duración: 2007 – 2010.

- **Nombre:** : Arquitectura para la eficiencia energética y sostenibilidad en entornos residenciales (TIN2009-14378-C02-01).

  Investigador principal: Juan Antonio Ortega Ramírez.

  Organismo financiador: Ministerio de Ciencia e Innovación.

  Período de duración: 2009 – 2012.

BIBLIOGRAFÍA

[AGOV07] Juan A. Alvarez, Luis González, Juan A. Ortega, and Francisco Velasco. Medidas de similitud entre caminos: Una propuesta. *XI Jornadas JARCA*, jun 2007.

[APAO07] Juan A. Alvarez, Carlos Pérez, Cecilio Angulo, and Juan Antonio Ortega. Combining smart tags and smart sensors for disabled people assistance. *KES*, mar 2007.

[AS02] Daniel Ashbrook and Thad Starner. Learning significant locations and predicting user movement with GPS. In *International Symposium on Wearable Computers*, 2002.

[AS03] Daniel Ashbrook and Thad Starner. Using GPS to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing*, 7:275–286, Oct 2003. doi:10.1007/s00779-003-0240-0.

[Ata83] Mikhail J. Atallah. A linear time algorithm for the Hausdorff distance between convex polygons. *Information Processing Letters*, 17:207–209, 1983.

[Bau72] L. E. Baum. An inequality and associated maximization technique in statistical estimation of probabilistic functions of a markov processes. *Inequalities*, 3:1–8, 1972.

[BBT02] Maren Bennewitz, Wolfram Burgard, and Sebastian Thrun. Learning motion patterns of persons for mobile service robots. In *International Conference on Robotics and Automation*, 2002.

[BE67] L. E. Baum and J.A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of a Markov process and to a model for ecology. *Bulletin of the American Mathematical Society*, 73(3):360–363, 1967.

[BJ07] Agne Brilingaite and Christian S. Jensen. Enabling routes of road network constrained movements as mobile service context. *GeoInformatica*, 11(1):55–102, 2007.

[BLP+08] Sean Barbeau, Miguel A. Labrador, Alfredo Perez, Philip Winters, Nevine Georggi, David Aguilar, and Rafael Perez. Dynamic management of real-time location data on gps-enabled mobile phones. In *The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (Ubicomm 2008)*, 2008.

[BP66] Leonard E. Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, 1966.

[BPT04] Sotiris Brakatsoulas, Dieter Pfoser, and Nectaria Tryfona. Modeling, storing, and mining moving object databases. *International Database Engineering and Applications Symposium*, pages 68–77, 2004.

[Cha07] Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1:300–306, 2007.

[CJP05] A. Civilis, C.S. Jensen, and S. Pakalnis. Techniques for efficient road-network-based tracking of moving objects. *IEEE Transactions on Knowledge and Data Engineering*, 17(5):698–712, 2005.

[CMJ+07] Raya Cristobal, Torrent Marc, Parera Jordi, Angulo Cecilio, and Català Andreu. Kneemeasurer. a wearable interface for joint angle measurements. In *II International Congress on Domotics, Robotics and Remote-Assistance for All*, 2007.

[Deg04] Yoshitaka Deguchi. Hev charge/discharge control system based on navigation information. In *International Congress and Exposition On Transportation Electronics*, 2004.

[Dem68] A.P. Dempster. A generalization of bayesian inference. *J. Royal Statistics Soc. Series*, 30:205–247, 1968.

[DGJS06] Aiden R. Doherty, Cathal Gurrin, Gareth J. F. Jones, and Alan F. Smeaton. Retrieval of similar travel routes using gps tracklog place names. In *The 3rd Workshop on Geographic Information Retrieval.*, 2006.

[DR93] Huttenlocher D.P. and W.J. Rucklidge. A multi-resolution technique for comparing images using the hausdorff distance. In *Computer Vision and Pattern Recognition*, 1993.

[FBAT09] Abdesslem Fehmi Ben, Phillips Andrew, and Henderson Tristan. Less is more: Energy-efficient mobile sensing with senseless. In *ACM Mobiheld, Barcelona, Spain, August 2009.*, 2009.

[fCB02] The Center for Conservation Biology. Vafalcons proyect, 2002.

[FK08] Jon Froehlich and Jhon Krumm. Route prediction from trip observations. In *Society of Automotive Engineers (SAE) World Congress*, 2008.

[FM09] Derek Fagan and René Meier. Using context and behavioral patterns for intelligent traffic management. In *CAMS '09: Proceedings of the 1st International Workshop on Context-Aware Middleware and Services*, pages 61–66, New York, NY, USA, 2009. ACM.

[GDB+05] Vibhav Gogate, Rina Dechter, Bozhena Bidyuk, Craig Rindt, and James Marca. Modeling transportation routines using hybrid dynamic mixed networks. In *Proceedings of the 21th Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, page 217, Arlington, Virginia, 2005. AUAI Press.

[Get93] Ivan Getting. The global positioning system. *IEEE Spectrum*, 30(12):36–47, 1993.

[HCL+05] Jeffrey Hightower, Sunny Consolvo, Anthony LaMarca, Ian Smith, and Jeff Hughes. Learning and recognizing the places we go. volume 3660, pages 159–176, Aug 2005.

[HS06] Ningning Hu and Peter Steenkiste. Quantifying internet end-to-end route similarity. In *Passive and Active Measurement Conference*, 2006.

[HT04] Ramaswamy Hariharan and Kentaro Toyama. Project lachesis: Parsing and modeling location histories. In *Geographic Information Science: Third International Conference, GIScience 2004. Adelphi, MD, USA: Springer-Verlag GmbH*, 2004.

[JHP00] Oliver Jan, Alan J. Horowitz, and Zhong-Ren Peng. Using gps data to understand variations in path choice. *Transportation Research Record Journal*, 1725:37–44, 2000.

[KB03] Abdolreza Karbassi and Matthew Barth. Vehicle route prediction and time of arrival estimation techniques for improved transportation system management. In *International Vehicle Symposium*, 2003.

[KH05] Jhon Krumm and Eric Horvitz. The microsoft multiperson location survey. Technical report, Microsoft, 2005.

[KH06] John Krumm and Eric Horvitz. Predestination: Inferring destinations from partial trajectories. In *Ubicomp*, pages 243–260, 2006.

[Kru06] John Krumm. Real time destination prediction based on efficient routes. In *Society of Automotive Engenieers 2006 World Congress*, 2006.

[Kru07] John Krumm. Inference attacks on location tracks. In *International Conference on Pervasive Computing (Pervasive)*, pages 127–143, 2007.

[Kru08] John Krumm. A Markov model for driver turn prediction. In *Society of Automotive Engineers (SAE) World Congress*, 2008.

[Kru09] John Krumm. Where will they turn: predicting turn proportions at intersections. *Personal and Ubiquitous Computing*, 2009.

[KWSB04] J. Kang, W. Welbourne, B. Stewart, and G. Borriello. Extracting places from traces of locations. *In Proceedings of the Second ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots (WMASH 2004)*, pages 110–118, 2004.

[KWSB05] Jong Hee Kang, William Welbourne, Benjamin Stewart, and Gaetano Borriello. Extracting places from traces of locations. *SIGMOBILE Mob. Comput. Commun. Rev.*, 9(3):58–68, 2005.

[Laa05] Kari Laasonen. *Knowledge Discovery in Databases: PKDD 2005*, volume ?, chapter Clustering and Prediction of Mobile User Routes from Cellular Data, pages 569–576. Springer Berlin / Heidelberg, 2005.

[LBC98] T. Liu, P. Bahl, and I. Chlamtac. Mobility modeling, location tracking, and trajectory prediction in wireless atm networks. *IEEE Journal on Selected Areas in Communications*, 16:922–936, 1998.

[LCC+05] Anthony LaMarca, Yatin Chawathe, Sunny Consolvo, Jeffrey Hightower, Ian E. Smith, James Scott, Timothy Sohn, James Howard, Jeff Hughes, Fred Potter, Jason Tabert, Pauline Powledge, Gaetano Borriello, and BillÑ. Schilit. Place lab: Device positioning using radio beacons in the wild. In *Pervasive*, pages 116–133, 2005.

[LFK04] Lin Liao, D. Fox, and H. Kautz. Learning and inferring transportation routines. In *19th National Conference on Artificial Intelligence (AAAI)*, 2004.

[LFK05] Lin Liao, D. Fox, and H. Kautz. Location-based activity recognition. In *Advances in Neural Information Processing Systems 19*, 2005.

[LFK07] Lin Liao, Dieter Fox, and Henry Kautz. Extracting places and activities from gps traces using hierarchical conditional random fields. *Int. J. Rob. Res.*, 26(1):119–134, 2007.

[Lin98] Dekang Lin. An information-theoretic definition of similarity. In *In Proceedings of the 15th International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann, 1998.

[Mar04] Natalia Marmasse. *Providing Lightweight Telepresence in mobile communication to enhance collaborative living.* PhD thesis, Massachusetts Institute of Technology, 2004.

[MB08] Scott Morris and Kobus Barnard. Finding trails. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*, 2008.

[MS00] Natalia Marmasse and Christopher Schmandt. Location-aware information delivery with commotion. *Second International Symposium on Handheld and Ubiquitous Computing (HUC)*, pages 151–171, 2000.

[MS02] Natalia Marmasse and Christopher Schmandt. A user-centered location model. *Personal and Ubiquitous Computing*, pages 318–321, 2002.

[MW06] Stefan Michaelis and Christian Wietfeld. Comparison of user mobility pattern prediction algorithms to increase handover trigger accuracy. *IEEE Vehicular Technology Conference*, 2006.

[oML02] Census of Marine Life. Post: Pacific ocean salmon tracking project, 2002.

[OMM02] S. Osentoski, V. Manfredi, and S. Mahadevan. Learning hierarchical models of activity. In *IEEE/RSJ International Conference on Intelligents Robots and Systems*, 2002.

[PAlGO07] Carlos Pérez, Cecilio Angulo, Juan A. Álvarez García, and Juan A. Ortega. Sistema asistencial experimental de monitorización de movimiento y comportamiento. In *UCAMI*, 2007.

[PLFK03] Don Patterson, Lin Liao, Dieter Fox, and Henry Kautz. Inferring high level behavior from low level sensors. In *Fifth Annual Conference on Ubiquitous Computing (UBICOMP)*, 2003.

[PLG+04] Donald J. Patterson, Lin Liao, Krysztof Gajos, Michael Collier, NIk Livic, Katherine Olson, Shiaokai Wang, Dieter Fox, and Henry Kautz. Opportunity knocks: A system to provide cognitive assistance with transportation services. In *Ubicomp 2004: Ubiquitous Computing*, 2004.

[Rab89] Lawrence R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. In *proceedings of the IEEE, vol 77, no. 2*, February 1989.

[RG07] D. Raskovic and D. Giessel. Battery-aware embedded gps receiver node. In *International Conference onMobile and Ubiquitous Systems: Networking and Services, 2007. MobiQuitous 2007.*, pages 1–6, Aug. 2007.

[RLRE09] Jason Ryder, Brent Longstaff, Sasank Reddy, and Deborah Estrin. Ambulation: A tool for monitoring mobile patterns over time using mobile phones. In *Proceedings of the Workshop on Social Computing with Mobile Phones & Sensors: Modeling, Sensing and Sharing (SCMPS09)*, August 2009.

[RMI07] Jun Rekimoto, Takashi Miyaki, and Takaaki Ishizawa. Lifetag: Wifi-based continuous location logging for life pattern analysis. In *LoCA*, pages 35–49, 2007.

[RSB+09] Sasank Reddy, Katie Shilton, Jeff Burke, Deborah Estrin, Mark H. Hansen, and Mani B. Srivastava. Using context annotated mobility profiles to recruit data collectors in participatory sensing. In *LoCA*, pages 52–69, 2009.

[RZ07] Ahmad Rahmati and Lin Zhong. Context-for-wireless: context-sensitive energy-efficient wireless data transfer. In *MobiSys '07: Proceedings of the 5th international conference on Mobile systems, applications and services*, pages 165–178, New York, NY, USA, 2007. ACM.

[SAG+93] BillÑ. Schilit, Norman Adams, Rich Gold, Michael Tso, and Roy Want. The PARCTAB mobile computing system. In *Proceedings Fourth Workshop on Workstation Operating Systems (WWOS-IV)*, pages 34–39. IEEE, October 1993.

[SBZS06] Reid Simmons, Brett Browning, Yilu Zhang, and Varsh Sadekar. Learning to predict driver route and destination intent. In *IEEE Intelligent Transportation Systems Conference*, sep 2006.

[Sen02] K. Sentz. *Combination of Evidence in Dempster-Shafer Theory*. PhD thesis, Binghamton University, 2002.

[Sho89] R. Shonkwiler. An image algorithm for computing the Hausdorff distance efficiently in linear time. *Information Processing Letters*, 30:87–89, 1989.

[Sho91] R. Shonkwiler. Computing the Hausdorff set distance in linear time for any lp point distance. *Information Processing Letters*, 38:201–207, 1991.

[SK05] Nancy Samaan and Ahmed Karmouch. A mobility prediction architecture based on contextual knowledge and spatial conceptual maps. *IEEE Transactions on Mobile Computing*, 4(6):537–551, 2005.

[SKJH06] Libo Song, D. Kotz, Ravi Jain, and Xiaoning He. Evaluating next-cell predictors with extensive wi-fi mobility data. *IEEE Transactions on Mobile Computing*, 5(12):1633–1649, 2006.

[SLBB04] Linda Sweanor, Ken Logan, Jim Bauer, and Walter Boyce. Puma project, 2004.

[THPP05] L. Toole-Holt, S. Polzin, and R. Pendyala. Two minutes per person per day each year: Exploration of growth in travel time expenditures. *Transportation Research Record*, 1917:45–53, 2005.

[TKTN09] Kohei Tanaka, Yasue Kishino, Tsutomu Terada, and Shojiro Nishio. A destination prediction method using driving contexts and trajectory for car navigation systems. In *SAC '09: Proceedings of the 2009 ACM symposium on Applied Computing*, pages 190–195, New York, NY, USA, 2009. ACM.

[TMK+06] Tsutomu Terada, Masakazu Miyamae, Yasue Kishino, Kohei Tanaka, Shojiro Nishio, Takashi Nakagawa, and Yoshihisa Yamaguchi. Design of a car navigation system that predicts user destination. In *IEEE 7th International Conference on Mobile Data Management (MDM'06)*, 2006.

[TZL+07] Kari Torkkola, Keshu Zhang, Haifeng Li, Harry Zhang, Christopher Schreiner, and Mike Gardner. Traffic advisories based on route prediction. In *Workshop on Mobile Interaction with the Real World (MIRW 2007)*, 2007.

[VF04] Dizan Vasquez and Th. Fraichard. Motion prediction for moving objects: a statistical approach. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 3931–3936, New Orleans, LA (US), April 2004.

[VFAL05] Dizan Vasquez, Thierry Fraichard, Olivier Aycard, and Christian Laugier. Intentional motion on-line learning and prediction. In *IEEE International Conference on robotics and automation*, 2005.

[Vin75] Thaddeus Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review*, XXII no 176:88–93, 1975.

[Vit67] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13 no 2:260–269, 1967.

[VKG02] Michail Vlachos, George Kollios, and Dimitrios Gunopulos. Discovering similar multidimensional trajectories. In *18th International Conference on Data Engineering (ICDE'02). pp 673-684*, 2002.

[Wei91] Mark Weiser. The computer for the twenty-first century. *Scientific American*, pages 94–10, sep 1991.

[WSA+95] Roy Want, Bill Schilit, Norman Adams, Rich Gold, Karin Petersen, John Ellis, David Goldberg, and Mark Weiser. The PARCTAB ubiquitous computing experiment. Technical Report CSL-95-1, Xerox Palo Alto Research Center, March 1995.

[YCcC09] Qian YE, Ling CHEN, and Gen cai CHEN. Personal continuous route pattern mining. *Journal of Zhejiang University Science A*, 10(2):221–231, 2009.

[ZFL+04] Changqing Zhou, Dan Frankowski, Pamela J. Ludford, Shashi Shekhar, and Loren G. Terveen. Discovering personal gazetteers: an interactive clustering approach. In *GIS*, pages 266–273, 2004.

[ZLTG07] Keshu Zhang, Haifeng Li, Kari Torkkola, and Mike Gardner. Adaptive learning of semantic locations and routes. In *LoCA*, 2007.

[ZSML04] P. Zhang, C. Sadler, M. Martonosi, and S. Lyon. Hardware design experiences in zebranet. *In Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004.

[ZTL+07] Keshu Zhang, K. Torkkola, Haifeng Li, C. Schreiner, H. Zhang, M. Gardner, and Zheng Zhao. A context aware automatic traffic notification system for cell phones. In *Distributed Computing Systems Workshops, 2007. ICDCSW '07. 27th International Conference on*, pages 48–48, June 2007.

11 de diciembre de 2009