UNIVERSIDAD NACIONAL DEL SUR

TESIS DE DOCTOR EN CONTROL DE SISTEMAS

Inferencia de intención de cruce de peatones utilizando la dinámica del cuerpo

Santiago Gerling Konrad

BAHÍA BLANCA ARGENTINA

2018

Prefacio

Esta tesis se presenta como parte de los requisitos para acceder al grado académico de Doctor en Control de Sistemas, de la Universidad Nacional del Sur y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otra. La misma contiene los resultados en investigaciones llevadas a cabo en el Departamento de Ingeniería Eléctrica y de Computadoras (DIEC) de la Universidad Nacional del Sur, durante el período comprendido entre el 25 de Noviembre de 2014 al 11 de Diciembre de 2018, bajo la dirección del Dr. Favio R. Masson, de la Universidad Nacional del Sur; y la dirección del Dr. Eduardo Nebot, de la Universidad de Sídney, Australia.

Ing. Santiago Gerling Konrad

UNIVERSIDAD NACIONAL DEL SUR

Secretaría General de Posgrado y Educación Continua

La presente tesis ha sido aprobada el ...../...../.....,

mereciendo la calificación de ..... (.....)

Dedicada a mis papás y a mi hermana.

Resumen

Se define la intención como la voluntad de una persona a realizar una acción, previo a desarrollarla, y denotada por su movimiento y actitud. En un escenario realista, el peatón podría tomar decisiones de riesgo para cruzar o no frente a un vehículo. La detección de caminar en la vereda y detenerse o caminar cerca de la calle no es suficiente para reconocer que el peatón pasará por delante del vehículo. La presente tesis aborda esta problemática para poder brindar a los vehículos que interactúan con peatones, información fundamental para mejorar los sistemas de seguridad en la búsqueda de la prevención de accidentes.

El principal desafío es determinar el tipo de información obtenida de los peatones, que debe ser medida y comunicada para que, a partir de la implementación de estrategias de aprendizaje automático, esta intención pueda ser determinada. En esta tesis se evalúa en primer lugar el desempeño del uso de información egocéntrica, con datos provistos y generados por el propio peatón. Las aceleraciones y velocidades de las extremidades del cuerpo se obtienen con sensores montados en diferentes partes del cuerpo. Estos datos dan indicios del tipo de actividad que realizará un peatón al cruzar o no frente a un vehículo. Sin embargo, esta aproximación tiene algunas desventajas, especialmente porque todos los peatones que se acercan al vehículo deben estar instrumentados. Aunque hoy es posible depender de teléfonos o pulseras inteligentes, estos dispositivos no siempre tienen la capacidad de comunicarse efectivamente con un vehículo.

Las cámaras, sensores que todos los vehículos inteligentes ya poseen, permiten reemplazar la información provista por los sensores montados en los peatones, extrayendo información dinámica por medio del procesamiento de imágenes. Para esto se extrae un esqueleto virtual del cuerpo de la persona en línea de vista del vehículo y se procesan varios cuadros consecutivos del video. De este modo se demuestra que la calidad de la información dinámica extraída por este medio es comparable con la provista por los sensores montados en las personas.

Con esta información se presenta un análisis del desempeño de algoritmos de estimación de la intención basados en redes neuronales. Los resultados demuestran que el objetivo de determinar la intención de que una persona cruce frente a un vehículo es posible de lograr y de hacerlo en forma confiable ante situaciones de tránsito diversas y reales.

Abstract

The intention is defined as the will of a person to perform an action, prior to developing it, and denoted by its movement and attitude. In a realistic scenario, the pedestrian can take risky decisions to cross or not in front of a vehicle. The detection of walking in the sidewalk and stopping or walking near the street is not enough to recognize that the pedestrian will cross in front of the vehicle. This thesis addresses this problem in order to provide vehicles that interact with pedestrians fundamental information to improve safety systems in search of accident prevention.

The main challenge is to determine the kind of information obtained from pedestrians, which must be measured and communicated so that, based on the implementation of automatic learning strategies, the intention can be determined. In this thesis, the performance of the use of egocentric information is evaluated with data provided and generated by the pedestrian himself. Accelerations and velocities of the limbs are obtained with sensors mounted on different parts of the body. These data gives indications of the kind of activity that a pedestrian will perform when crossing or not in front of a vehicle. However, this approach has some disadvantages, especially because all pedestrians approaching the vehicle must be instrumented. Although today it is possible to rely on smartphones or wristbands, these devices do not always have the ability to communicate effectively with a vehicle.

The cameras and sensors that all intelligent vehicles already have, allow to replace the information provided by the sensors mounted on pedestrians, extracting dynamic information through the processing of images. For this, a virtual skeleton of the body of the person in the line of sight of the vehicle is extracted and several consecutive pictures of the video are processed. In this way, it is demonstrated that the quality of the dynamic information extracted by this means is comparable with the information provided by the sensors mounted on people.

With this information, an analysis of the performance of intention estimation algorithms based on neural networks is presented. The results demonstrate that the objective of determining the intention of a person crossing in front of a vehicle is possible to achieve and to do it in a reliable way before diverse and real traffic situations.

Índice

Lista de figuras **15**

Lista de tablas **21**

Acrónimos **23**

1. Introducción **27**

1.1. Problemática y estado actual del arte 27

1.2. Objetivo 31

1.3. Resúmen de los aportes 31

1.4. Organización 33

2. Arquitecturas para simulación y adquisición de datos en ITS[1] **35**

2.1. La importancia de los simuladores 36

2.2. Sistema de adquisición y almacenamiento de datos reales de conducción 37

2.2.1. Paradigma Publicadores-Suscriptores 38

2.2.2. Comunicación entre procesos 39

2.2.3. Registro de datos 40

2.2.4. Comparación con otros sistemas similares 41

2.3. Plataforma de visualización y simulación de datos 44

2.3.1. Virtualización del ambiente 44

2.3.2. Motor de videojuegos para simulación 47

2.3.2.1. Potencialidades 48

2.3.3. Simulador para aplicaciones de ITS 51

[1]Sistemas de Transporte Inteligentes -*Intelligent Transportation Systems*-

2.4. Conclusión del capítulo 56

3. Medición directa de la dinámica del peatón 59

3.1. Uso del sensor acelerómetro 59

3.2. Algoritmos de clasificación 60

3.3. Información de aceleración 62

3.3.1. Adquisición 63

3.3.2. Experimentación 64

3.3.3. Método de validación 67

3.4. Análisis de aceleraciones antes de un cruce 67

3.5. Cálculo de la intención de cruce 73

3.6. Conclusión del capítulo 77

4. Medición indirecta de la dinámica del peatón 81

4.1. Obtención de la representación por esqueletos e información derivada 82

4.1.1. Extracción del esqueleto virtual 82

4.1.2. Corrección de fallas en detecciones del esqueleto 84

4.1.3. Cálculo de la información dinámica 86

4.2. Evaluación de la dinámica del cuerpo y sus partes 87

4.2.1. Experimentación 87

4.2.2. Validación 90

4.2.2.1. Con peatón detenido 90

4.2.2.2. Con peatón caminando 92

4.2.3. Resultados 97

4.2.3.1. Análisis relativo a la dirección del peatón 97

4.2.3.2. Extremidades 98

4.3. Cálculo de la intención de cruce 102

4.4. Conclusión del capítulo 105

5. Intención del peatón basado en la dinámica del cuerpo 107

5.1. Redes neuronales para la determinación de la intención 108

5.1.1. Redes LSTM[2] 109

5.1.2. Obtención de información de peatones reales 110

[2] *Long-Short Term Memory*

5.1.3. Pre-entrenamiento de la red para la evaluación de la calidad de la información utilizada 112
5.1.4. Entrenamiento de la red 113
5.2. Estimación de la intención 114
5.2.1. Análisis de las estimaciones 117
5.3. Conclusión del capítulo 121

6. Conclusiones **125**
6.1. Proyección y trabajo futuro 127
6.2. Epílogo 128

Bibliografía **129**

A. Redes LSTM **141**
A.1. La idea central detrás de las redes LSTM 141
A.2. El camino a través de la red 142

Lista de figuras

2.1. Diferentes puntos GPS comparados en Google Maps (columna izquierda) y OpenStreetMap (columna derecha). 46
2.2. Nube de puntos densa, obtenida mediante la técnica de SFM[3], la cual permite modelar el entorno en 3 dimensiones. 47
2.3. Modelo de puntos con la superficie creada. 48
2.4. Representación del simulador en diagramas de bloques. 51
2.5. Diferentes puntos de vista de la misma escena. 53
2.6. Comparación de imágenes y otras vistas. La vista principal pertenece a una cámara montada en el vehículo del simulador; la vista superior derecha es un mapa de la ciudad donde se muestra el vehículo y la trayectoria recorrida; la vista central derecha es la imagen de una cámara real, montada en el vehículo que capturó los datos; y la vista inferior derecha es una vista en tercera persona que permite observar la actividad del simulador desde cualquier punto fijo. 53
2.7. Símbolo sobre el vehículo para indicar algún tipo de peligro. Este tipo de advertencias son el resultado de algoritmos de prevención evaluados en el simulador. 55

3.1. Momento de la intención (rojo-continuo), marca de intención (verde) y momento de la acción (azul-discontinuo) expresados en una línea temporal. 62

[3] *Structure From Motion*

3.2. Momento de la intención (rojo-continuo), marca de intención (verde) y momento de la acción (azul-discontinuo) mapeados físicamente. 63
3.3. Dirección de los ejes preestablecidos. Ésta es la configuración de ejes utilizada para la adquisición de los datos. 65
3.4. Mapa de la experimentación. En el mismo se pueden observar los hitos que debía cumplir el peatón, iniciando en el Punto 1 y finalizando en el Punto 7. 66
3.5. Gráfico general de los datos de los ejes X, Y y Z. En el mismo se pueden ver alteraciones de la señal y el área perteneciente a la calle (sombreada). 68
3.6. Comparación de los datos crudos de los 3 ejes. El sombreado indica el área correspondiente a la calle. 69
3.7. Envolventes superior e inferior de los datos del eje Z. En línea gruesa se observa la envolvente superior (violeta) e inferior (naranja). También se muestran los datos crudos en color claro y el área correspondiente a la calle en sombreado. 70
3.8. Envolvente superior y envolvente máxima, producto del algoritmo. En línea gruesa se observa la envolvente superior (violeta) y la envolvente de máximos (verde). También se muestran los datos crudos en color claro y el área correspondiente a la calle en sombreado. Los puntos amarillos indican los máximos de la envolvente superior en cada ventana de tiempo. 71
3.9. Detección temprana de cercanía a la cinta asfáltica. En línea fina (verde) se muestra la envolvente de máximos y en línea gruesa (rojo) la detección temprana. Las zonas sombreadas indican el área correspondiente a la calle. 72
3.10. Aumento de tamaño de la Figura 3.9 en $t = 60s$. En línea fina (verde) se muestra la envolvente de máximos y en línea gruesa (rojo) la detección temprana. La zona sombreada indica el área correspondiente a la calle. 72

3.11. Gráfica de las mejores precisiones obtenidas evaluando el clasificador con diferentes lapsos de tiempo en los datos. 75
3.12. Matriz de confusión de la totalidad de los datos utilizados. 76
3.13. Tasa de verdaderos positivos y falsos negativos. 77
3.14. Valores predichos positivos y tasa de descubrimiento de falsos. . . 77

4.1. Esqueleto obtenido usando *OpenPose*. (a) muestra los puntos del esqueleto. (b) ilustra el esqueleto y los ángulos correspondientes. . 84
4.2. Secuencia del uso de información semántica para eliminar falsas detecciones. La imagen original capturada por la cámara frontal del vehículo se muestra en (a). (b) ilustra la detección del falso esqueleto en el sector derecho de la imagen. En (c), la información semántica es usada para filtrar la falsa detección. Los peatones son coloreados en amarillo, los vehículos en rojo, las áreas de manejo en marrón, la vegetación en verde y el cielo en azul, entre otros. . 86
4.3. Vehículo eléctrico autónomo modernizado con tecnología de visión perceptiva. 89
4.4. Dos peatones realizando diferentes acciones en la misma escena y en el mismo momento. Uno de ellos está parado en la borde la calle mientras el otro la está cruzando. 89
4.5. Comparación de la información de la velocidad angular de la muñeca. La línea azul representa la información del eje Z del giróscopo, mientras que la línea naranja es la velocidad angular obtenida del procesamiento del esqueleto en las imágenes. 91
4.6. Comparación de la información de aceleración lineal de la muñeca. En ambos gráficos, la línea azul es la información provista por la IMU[4] y la línea naranja es la información obtenida del procesamiento de las imágenes. El gráfico superior muestra la aceleración en el eje X mientras que el inferior muestra la del eje Y. 92
4.7. Imágenes del segundo experimento tomadas de los videos. La representación de los esqueletos es presentada sobre los peatones. . . 93

[4]Unidad de Medición Inercial -*Inertial Measurement Unit*-

4.8. Velocidad angular del primer peatón caminando. La línea azul representa la información de la IMU mientras que la naranja representa la información desde el procesamiento de la imagen. . . . 94
4.9. Velocidad angular del segundo peatón. 95
4.10. Aceleración lineal en el eje X (arriba) e Y (abajo) correspondientes al primer peatón. La comparación está dada por la información de la IMU, en línea azul, y el procesamiento de la imagen, en línea naranja. 96
4.11. Aceleración lineal en ambos ejes correspondiente al segundo peatón. 97
4.12. Ángulos entre diferentes articulaciones y segmentos de los brazos respecto a la línea horizontal, extraído desde el esqueleto de una persona que camina hasta $t = 2s$, se detiene hasta $t = 3s$ y luego camina de nuevo. 99
4.13. Velocidad angular del segmento hombro-muñeca del brazo derecho. 100
4.14. Ángulos entre diferentes articulaciones y secciones de las piernas, respecto a la línea horizontal. El último gráfico también muestra la diferencia del ángulo de las rodillas. La pierna derecha está coloreada en azul, la izquierda en naranja y la diferencia entre ambas en línea punteada amarilla. 101
4.15. Velocidad angular de los segmentos rodilla-tobillo de ambas piernas. Es fácil reconocer las diferentes acciones llevadas a cabo por el peatón, como caminar y detenerse. 101
4.16. Aceleración de los tobillos en los ejes X e Y. La aceleración inicial, desde el momento detenido al de caminar se muestra claramente en el momento $t = 3$ segundos. 102
4.17. Matriz de confusión de la totalidad de los datos utilizados. 104
4.18. Tasa de verdaderos positivos y falsos negativos. 104
4.19. Valores predichos positivos y tasa de descubrimiento de falsos. . . 105

5.1. Representación del funcionamiento la una celda de una red LSTM. 109

5.2. Representación de los ángulos usados como complemento a la información que alimenta a la red neuronal. En esta imagen se representan aquellos referidos al punto de la rodilla derecha del peatón. El ángulo α se calcula a partir del vértice inferior izquierdo y el ángulo β a partir del vértice inferior derecho. 112
5.3. Progreso del entrenamiento de la red neuronal. 114
5.4. Matriz de confusión del proceso de entrenamiento. 115
5.5. Matriz de confusión del proceso de evaluación. 116
5.6. Gráficos para mostrar las fallas en el uso de ventanas temporales de diferente duración para obtener resultados de la red. Los gráficos superiores, de izquierda a derecha, son el resultado del uso de las ventanas de 3, 2 y 1 segundos para peatones con la intención de cruzar frente al vehículo (probabilidad ~ 1). Los gráficos inferiores están en el mismo orden del uso de las ventanas pero para peatones sin intención de cruzar (probabilidad ~ 0). 117
5.7. Probabilidad de cruzar basada en una ventana de entrada de 3 segundos. 119
5.8. Probabilidad de cruzar basada en una ventana de entrada de 2 segundos. 119
5.9. Secuencia de uno de los ejemplos más relevantes. En la fila superior se muestra el gráfico de la probabilidad y en la inferior la imagen de video correspondiente a tal, para 6 instantes de tiempo diferentes. 122

A.1. Representación de la celda enfocada en el flujo de información principal C. 141
A.2. El círculo rojo claro representa un punto de operación, en este caso una multiplicación, y el rectángulo amarillo representa una compuerta, que en este caso es una función sigmoide. 142
A.3. Representación de la celda enfocada en la “compuerta de olvido”, la cual decide la información que no es relevante y debe olvidarse. 142
A.4. Representación de la celda enfocada en la “compuerta de entrada”, la cual decide qué información se utilizará para actualizar el estado actual. 143

A.5. Los valores candidatos son utilizados para actualizar el estado de la celda. 143
A.6. La última parte de la celda decide qué valores dispondrá en la salida para que sean utilizado en la siguiente celda. 144

Lista de tablas

2.1. Comparación de bibliotecas de comunicación. 42

3.1. Porcentajes de detecciones de los distintos tipos de experimentos. 73

3.2. Tabla de precisiones resultantes de los distintos clasificadores entrenados, ordenados de mayor a menor. 76

4.1. Correlación cruzada de un peatón detenido. 92

4.2. Correlación cruzada de los peatones que caminan. 94

Acrónimos

ADAS Sistemas Avanzados de Asistencia al Conductor -*Advanced Driver Assistance Systems*- 82

ECG Electro-Cardiograma 55

EEG Electro-Encefalograma 55

FFT Transformada Rápida de Fourier -*Fast Fourier Transform*- 68

FOV Campo de visión -*Field Of View*- 88

fps cuadros por segundo -*frames per second*- 88

GPS Sistema de Posicionamiento Global -*Global Positioning System*- 29

GPU Unidad de Procesamiento Gráfico -*Graphics Processing Unit*- 88

IMU Unidad de Medición Inercial -*Inertial Measurement Unit*- 17

ITS Sistemas de Transporte Inteligentes -*Intelligent Transportation Systems*-11

k-NN k-Vecinos Cercanos -*k-Nearest Neighbors*- 60

LCM *Lightweight Communications and Marshalling* 37

LIDAR *Light Detection and Ranging* 57

LSTM *Long-Short Term Memory* 12

MCL *Multiprocess Communications Library* 37

OSM *OpenStreetMap* 45

RMS Raíz Media Cuadrática -*Root Mean Square*- 73

RNN Redes Neuronales Recurrentes -*Recurrent Neural Networks*- 108

ROS *Robot Operating System* 37

RSU Unidad de Carretera -*Road-Side Unit*- 41

SFM *Structure From Motion* 15

SIFT Transformación de característica invariante a la escala -*Scale-Invariant Feature Transform*- 44

SURF Funciones Robustas y aceleradas -*Speeded-Up Robust Features*- 44

SVM *Support Vector Machines* 60

TCP *Transmission Control Protocol* 43

UDP *User Datagram Protocol* 43

URI Identificador de Recursos Uniforme -*Uniform Resource Identifier*- 40

USyd Universidad de Sídney -*The University of Sydney*- 89

ZMQ *ZeroMQ* 37

Capítulo 1

Introducción

1.1. Problemática y estado actual del arte

En el tránsito, las interacciones entre vehículos y usuarios vulnerables, como peatones y ciclistas, son procesos complejos que a menudo dependen del buen comportamiento o sujeción a las normas de circulación. Existe la expectativa de que los conductores humanos toleren un nivel de flexibilidad de reglas de tránsito que depende en gran medida del contexto de la situación. En un futuro cercano, se requerirá que tanto los vehículos autónomos que interactúen en entornos urbanos, como los sistemas de seguridad en vehículos conducidos por humanos, entiendan este contexto para tomar las decisiones apropiadas cuando operen alrededor de estos usuarios.

El contexto de tránsito se compone no sólo de la detección de los agentes (vehículos, personas, etc.) y las características o reglas del lugar, sino también de la capacidad de predecir el movimiento futuro de cada uno de los agentes.

Se ha demostrado que mientras las personas conducen, sólo pueden enfocar su atención en una sola tarea y cambiar el foco a otra rápidamente con pequeñas distracciones [1], ante la aparición de fatiga o durante la conducción monótona prolongada [2]. Si bien hay desarrollos relacionados a la detección y seguimiento de la atención del conductor [3] [4], la seguridad peatonal debe estar acompañada por otro tipo de enfoques. Por ello es necesario que el vehículo sea capaz de afrontar parte de esta seguridad, reconociendo con anticipación la actividad que realizará el peatón, o en otras palabras, cuál es su intención.

Durante los últimos años ha habido un progreso significativo en tecnología diseñada para prevenir accidentes entre peatones y vehículos. Muchos fabricantes de vehículos están introduciendo nueva tecnología para alertar a los conductores si los peatones son detectados en la trayectoria estimada del vehículo [5] [6] [7].

En el caso de los peatones, aunque su velocidad es en general mucho menor que la de los vehículos, son mucho más ágiles. Un peatón puede cambiar de dirección muy rápidamente haciendo un giro agudo (por ejemplo de 90º) sin reducir la velocidad. Esta agilidad es lo que limita a los sistemas actuales para lograr predicciones confiables de movimiento peatonal a sólo unos pocos cientos de milisegundos. Para predecir entonces el comportamiento del usuario vulnerable con antelación, es necesario encontrar indicadores de intención.

La intención se define como la voluntad de una persona de realizar una acción, pero en los momentos previos a llevarla a cabo. La intención está dada en general por el movimiento y la actitud de la persona. Otros factores pueden influir en la detección de la intención, y en el caso de los peatones puede ser la orientación del cuerpo, de la cabeza [8] o la postura [9].

Reconocer la intención de un peatón que circula por la vereda es primordial para la seguridad de un vehículo, sea autónomo o no, cuyo conductor no se encuentre con la atención enfocada. La intención no es una variable medible y debe ser inferida a través de medidas indirectas y modelos [10]. Un peatón es un agente de tránsito seguro mientras se encuentra alejado de los vehículos en movimiento, pero cuando se aproxima a ellos, sea acercándose a una esquina con intención de cruzar o haciéndolo a mitad de cuadra, se deben tomar precauciones al respecto. En ese instante, su intención puede ser la de cruzar o la de doblar. Detectar la intención debe hacerse con suficiente antelación y al mismo tiempo ser lo suficientemente robusta para reducir el número de falsas alarmas.

Actualmente hay varias contribuciones destacadas para abordar este problema utilizando diferentes modalidades y arreglos de sensores. Si bien puede predecirse que una persona cruzará la calle, en [11] [12] las mediciones para lograrlo provienen de cámaras monoculares. Mientras, las aplicaciones más exitosas en términos de implementación comercial que se basan en la tecnología de visión, usan tanto cámaras monoculares [13] como estéreos [14]. Más recientemente, el

alcance del láser y los sensores de ángulo han demostrado un alto nivel de confiabilidad para detectar y discriminar a los peatones en escenarios urbanos [15] [16]. También se han realizado demostraciones de tecnología de radar para detectar peatones [17] con cierto nivel de éxito.

Una estrategia para inferir la intención es utilizar la información generada por el propio peatón haciendo uso de sensores inerciales para estimar comportamientos. La información inercial, como relaciones de giro y aceleraciones de ciertas partes del cuerpo, proveen información dinámica de alta calidad que puede ser esencial para estimar las intenciones y comportamientos de una manera muy confiable.

Esta información se está volviendo más ubicua debido a la introducción generalizada de dispositivos portátiles como teléfonos móviles, relojes inteligentes y dispositivos para monitoreo de ejercicios, los cuales resultan cada vez más económicos. Éstos incorporan variedades de sensores que incluyen acelerómetros, giróscopos y GPS[1] que permiten la medición de información dinámica de peatones, útiles para reconocer las actividades de estas personas, como caminar, correr, nadar, entre otros [18]. Estos dispositivos pueden ser localizados en la muñeca, como las pulseras deportivas o relojes inteligentes, o en un bolsillo, como los teléfonos. Reconocer la ubicación del dispositivo en el cuerpo [19] resulta importante para reconocer las actividades del peatón. Estos dispositivos pueden proveer información dinámica de alta frecuencia y gran calidad que es requerida para estimar la intención y comportamiento de los peatones de modo seguro. No obstante, el uso de un solo dispositivo no contribuye con suficiente información como para reconocer las actividades peatonales y es necesario usar más de uno, como se demuestra en [20]. Además, esto permite plantear modelos de predicción y obtener información de un grupo de individuos y no sólo de una persona en particular. Sin embargo, el problema más importante es que la información provista por los dispositivos portátiles no puede ser usada por un vehículo inteligente porque la comunicación aún no es un componente estándar de estos dispositivos. Aunque algún tipo de tecnología ha sido desarrollada para resolver este problema [21] [22], ésta no está ampliamente implementada. Por otra parte, todos los

[1]Sistema de Posicionamiento Global -*Global Positioning System*-

peatones deberían estar instrumentados para que la detección de la intención sea eficiente y completa en todo el entorno.

Recientemente se ha demostrado que varios algoritmos basados en visión detectan de manera confiable a los peatones y estiman una postura 3D completa utilizando un esqueleto virtual como representación del cuerpo. Por esta razón, obtener información precisa del cuerpo del peatón usando los sistemas de visión vehiculares ha sido extensamente investigado. En esta tesis se demostrará que, mediante el uso de sistemas basados en visión, es posible obtener una representación dinámica de un peatón con información de calidad similar a la de los sensores usados en los dispositivos portátiles. Esto permite la implementación de algoritmos robustos de intención de peatones basados en información visual.

La contribución potencial que la visión por computadora puede ofrecerle a los vehículos inteligentes ha sido demostrada en los últimos años [23] [24]. La visión es la principal modalidad de sensado usada por los conductores humanos para percibir el entorno. Sin embargo, el uso de la visión para inferir las intención de los peatones tiene éxito limitado. Esto se debe al hecho de que una imagen simple no puede capturar información dinámica fácilmente.

Los modelos peatonales que usan partes deformables [25] han sido exitosos para detectar y seguir completa y parcialmente peatones ocultos en entornos concurridos. Diferentes partes del cuerpo están explícitas y se consideran partes móviles y ubicaciones relativas de cada parte. Más recientemente, la captura de posiciones en 2D de las articulaciones de los peatones y sus esqueletos son posibles mediante el uso de imágenes monoculares [26] [27]. El uso de esta información, cuadro a cuadro en el video, permite obtener información dinámica precisa de los peatones. En [28] los autores presentan un modelo de movimiento del peatón que tiene propiedades únicas tras comparar el fondo y los objetos rígidos en un perfil de movimiento espacio-temporal. Este enfoque permite identificar el movimiento de la pierna del peatón junto con el movimiento del cuerpo durante un corto período de tiempo. En [29] se presenta un enfoque utilizando cámaras, sensores de profundidad y sensores inerciales para recopilación de datos y reconocimiento de la marcha. Allí se proponen nuevos algoritmos para extraer la dinámica del movimiento, pero este procedimiento no puede ser extendido fácilmente a otras

partes del cuerpo. Esta información puede ser incorporada en modelos predictivos detallados para estimar la intención del peatón si la información y el modelo son de alta calidad.

Determinar la intención implica analizar información previa al evento que se procura conocer. Por ejemplo, se puede detectar que un peatón cruza una calle pero para averiguar la voluntad de hacerlo es necesario medir antes que esto ocurra. Esto se lleva a cabo extrayendo de la información disponible, aquellos indicios necesarios de intención. Una forma de resolver esto es hacer uso de redes neuronales y clasificadores que capturen, a partir de datos provistos por el peatón, los elementos necesarios para ayudar a determinar la intención. Esta información es vital para reducir falsas alarmas generadas por los sistemas anti-colisión.

1.2. Objetivo

El fin fundamental de esta tesis es demostrar que la intención de un peatón que circula por la vía pública, y que va a cruzar por delante del vehículo, puede ser inferida a partir de extraer información dinámica con sensores existentes en un vehículo. Por tal motivo, primero se validan los métodos para obtener esta información y luego se analiza su calidad con motivo de determinar si es relevante para este tipo de aproximaciones. Por último se la utiliza para la evaluación de algoritmos que logren obtener la intención de los peatones en momentos previos a realizar la acción. Esto permitirá que la intención de los peatones sea integrada a los algoritmos de asistencia al conductor y en vehículos autónomos mejorando la seguridad de los peatones y vehículos mismos.

1.3. Resúmen de los aportes

- Se presenta el desarrollo de una plataforma para la adquisición de datos de sensores montados en un vehículo. La plataforma es utilizada para crear los conjuntos de datos necesarios para el análisis del entorno del vehículo, lo que incluye no sólo información del propio vehículo sino también de otros agentes del tránsito como peatones y ciclistas que circulan a los alrededores

[30].

- Se diseña e implementa una plataforma de simulación virtual donde se recrean y establecen condiciones de manejo específicas. De este simulador se obtiene información basada en las pruebas de conducción de diferentes personas; se reproducen datos previamente adquiridos desde el mismo simulador o desde la plataforma de adquisición antes mencionada; y se entrenan y evalúan algoritmos desarrollados que ayuden a mejorar los sistemas de seguridad de los vehículos [31].

- Se presenta un análisis basado en la información de aceleración de un acelerómetro portado por un peatón. En este análisis se evalúa si este tipo de información es útil para inferir la intención de cruzar una calle. Para ello se buscaron patrones que determinen la posibilidad de detectar la intención utilizando un experimento propio, donde se obtuvo no sólo información de aceleración, sino también de GPS, para ayudar a localizar el momento en que la persona se aproxima al cruce [32].

- Se realiza la primera aproximación en la estimación de la intención peatonal. En este paso se utiliza la información egocéntrica, o provista por el propio peatón, de las aceleraciones del cuerpo. Una red de clasificación es entrenada para inferir la decisión del peatón momentos antes de cruzar (o no) la calle [33].

- Se realiza un segundo experimento consistente en adquirir el mismo tipo de información de aceleraciones (y sus consecuentes velocidades) del peatón a partir de cámaras montadas en el vehículo. Las imágenes son procesadas para generar esqueletos virtuales sobre los cuerpos de los peatones permitiendo obtener información de las extremidades y articulaciones de los mismos.

- Se valida la información adquirida y se la compara con la de los dispositivos acelerómetros utilizados anteriormente, obteniendo una alta correlación entre ambas fuentes de información. Este enfoque permite desvincular la

captura de información de los dispositivos portados por los peatones confinándola sólo a aquella obtenida desde el vehículo de interés [34].

- Haciendo uso de la información previamente validada, se evalúa la dinámica de los cuerpos y se analiza su utilización en la estimación de la intención. En este caso, sólo se utilizan los datos aportados por el propio vehículo [35].
- Se analiza la información dinámica obtenida con el fin de utilizarla como entrada en un clasificador. Se evalúa si este tipo de herramientas puede distinguir la intención de un peatón en momentos previos a cruzar o no frente al vehículo.
- Se presenta el uso de una red neuronal alimentada con información obtenida de los esqueletos para la detección de la intención de los peatones. Se utiliza una red neuronal con memoria temporal y se ensayan diferentes longitudes de secuencias de entradas para lograr el resultado óptimo en cuanto tiempo de secuencia necesaria para obtener la intención.

1.4. Organización

En el presente Capítulo se brindó una introducción al tema de estudio. Se realizó un repaso bibliográfico que describe el contexto y la problemática en base a la cual se desarrolla esta tesis. También se planteó el objetivo general que se abordará a lo largo de los siguiente capítulos y una lista de los aportes realizados durante el desarrollo de la misma.

El Capítulo 2 corresponde al desarrollo de un entorno virtual donde se evalúan los algoritmos desarrollados así como también se describe la plataforma adquisidora utilizada y el proceso llevado a cabo para generar conjuntos de datos adaptados a las necesidades de este trabajo.

El Capítulo 3 presenta una aproximación basada en la detección de patrones en la dinámica de los peatones para el reconocimiento de la intención. El desarrollo se realiza utilizando información provista por dispositivos montados en los peatones, de los cuales se capturan y analizan las aceleraciones y velocidades de diferentes extremidades.

El Capítulo 4 extiende los resultados del Capítulo 3 utilizando el mismo tipo de información, pero obtenida desde cámaras montadas en el vehículo. Para ello se recrean esqueletos virtuales sobre el cuerpo de los peatones utilizando procesamiento de imágenes. Esto permite obtener información de la dinámica de diferentes puntos del cuerpo a través de cuadros consecutivos del video. Un clasificador es utilizado para analizar esta información y proveer una estimación de la intención del peatón evaluado.

El Capítulo 5 presenta un enfoque en el cual se utilizan los puntos de las articulaciones del esqueleto para entrenar una red neuronal con memoria. La misma logra clasificar las probabilidades (o intención) de que el peatón cruce (o no) frente al vehículo. Esta probabilidad se obtiene con antelación a que el peatón ejecute la acción prevista.

Por último, el Capítulo 6 aborda las conclusiones generales de esta tesis y pone de manifiesto los trabajos futuros necesarios para continuar con esta línea de investigación.

Capítulo 2

Arquitecturas para simulación y adquisición de datos en ITS

La disponibilidad de datos realistas de alta calidad de situaciones de manejo o tránsito, permite desarrollar soluciones innovadoras para los Sistemas de Transporte Inteligentes -*Intelligent Transportation Systems*-. La toma de datos debe ser diligente, el diseño de los experimentos especialmente cuidados pero realistas y las pruebas de los algoritmos desarrollados implementados sobre estos datos o situaciones. Todo esto junto permite que la investigación progrese.

La posibilidad de usar simuladores es un elemento clave en situaciones de tránsito reales ya que permite evaluar algoritmos sin poner en riesgo personas y bienes. La información realista que alimenta estos simuladores, capturada previamente por diferentes sensores permite analizar conductores, o agentes de tránsito como los peatones, en experimentos controlados pero que puedan terminar en un riesgo virtual. En este contexto los métodos confiables y flexibles para la adquisición, el almacenamiento y el análisis de datos son claves. Mantener la infraestructura de datos es difícil y consume tiempo. A medida que los sistemas de datos crecen en escala y complejidad, la administración del sistema se vuelve cada vez más intensiva en recursos y propensa a fallas.

En este capítulo se presenta el desarrollo de un simulador a partir de la virtualización del entorno aplicando estrategias de procesamiento de imágenes y un motor de videojuegos, que brinda el soporte y las funcionalidades necesarias para la simulación. Además, se introduce una plataforma desarrollada para la captura

de los datos utilizados para las experimentaciones que se presentan a lo largo de esta tesis. Finalmente, se presentan resultados con datos experimentales en situaciones de tráfico normales demostrando entre otros aspectos el funcionamiento en tiempo real de este entorno.

2.1. La importancia de los simuladores

El empleo de datos reales almacenados durante pruebas de campo tiene múltiples aplicaciones en ITS. Permite recrear las situaciones de tránsito para ser estudiadas, pero además posibilita el entrenamiento y verificación de algoritmos. Un claro ejemplo es el que se describe en [36], donde se utilizan los datos de vuelo de un avión para recrear la escena en un entorno virtual y observar cómo se desenvolvió el piloto y cuáles fueron sus errores. Esto posibilita estudiar y también afectar la tarea del piloto mientras conduce el avión sin exponerlo a una situación peligrosa. Lo mismo sucede con la conducción de un vehículo y, por ello, estas recreaciones son verdaderamente útiles ya que es posible repetirlas indefinidamente y analizar en ellas diversos factores mediante procedimientos que no son factibles de llevar a cabo durante la prueba original por el riesgo que implica para personas y equipos. En forma complementaria, un simulador habilita el análisis del comportamiento de conductores ante situaciones de tránsito controladas en experimentos donde se pretenden demostrar hipótesis de trabajo.

Esta motivación estimuló el desarrollo de varios tipos de simuladores orientados a distintos fines. En [37] por ejemplo se presenta una estrategia de visualización de información en tiempo real para situaciones de emergencia; en [38] la simulación se emplea con el objetivo de evaluar el tráfico a micro y macro escala; y en [39] la experimentación a nivel científico para la evaluación de los conductores en diferentes situaciones de riesgo y su comportamiento al conducir un vehículo.

Este capítulo muestra la viabilidad de utilizar un motor de videojuegos para la construcción de un simulador de conducción orientado a la visualización de datos, los cuales son obtenidos desde las plataformas vehiculares que nuestro grupo posee; a la recreación de escenas virtuales a partir de datos reales; y a la

adquisición de información de personas que lo utilicen. No es objetivo mostrar la implementación del software, sino su aplicación en trabajos científicos.

Como se mencionó, los datos utilizados provienen de información preexistente (mapas por ejemplo) y de imágenes tomadas por vehículos equipados con diferentes sensores que recolectan en forma cooperativa ésta y toda la información propia y del entorno (otros vehículos, infraestructura, etc.). La captura se realiza mediante arquitecturas capaces de adquirir, procesar y almacenar grandes cantidades de datos. Estas capacidades son un punto importante en la adquisición, ya que la pérdida de información puede afectar directamente los resultados de los algoritmos que con ella se entrenan. En la bibliografía se destacan arquitecturas como LCM[1] [40], RabbitMQ [41], ROS[2] [42] y ZMQ[3] [43]. Sin embargo, no existe al momento una plataforma capaz de recolectar la información de los sensores que poseen nuestros vehículos. Para ello, una nueva arquitectura se propuso con motivo de proveer los mejores resultados, en cuando a adquisición, para las plataformas mencionadas. A ella se la ha llamado MCL[4] y se describe en la Sección 2.2.

2.2. Sistema de adquisición y almacenamiento de datos reales de conducción

Desarrollar software para administrar tareas simultáneas y complejas es difícil y consume mucho tiempo. Para garantizar el éxito y la longevidad de un sistema, el software debe ser extensible, fácil de mantener y confiable. En esta sección se propone una arquitectura de software para leer, procesar y almacenar datos en aplicaciones de ITS. La arquitectura de software propuesta está diseñada para poner énfasis en la flexibilidad, facilidad de mantenimiento y extensibilidad. La capacidad de agregar nuevos sensores y algoritmos al sistema sin requerir grandes cambios en el código base hace que el sistema sea una herramienta efectiva para investigación y desarrollo. Dos conceptos clave sustentan la filosofía de diseño del

[1] *Lightweight Communications and Marshalling*
[2] *Robot Operating System*
[3] *ZeroMQ*
[4] *Multiprocess Communications Library*

software: el paradigma publicador-suscriptor y la comunicación entre procesos, que son descriptos en la Sección 2.2.1 y 2.2.2 respectivamente.

2.2.1. Paradigma Publicadores-Suscriptores

Al dividir el software en unidades de funcionalidad, se pueden desarrollar y mantener pequeñas porciones de código de forma independiente. Estas unidades pueden combinarse de manera flexible al definir cómo se transfiere la información de una unidad a otra. Para ello se hace uso de un acoplamiento flexible utilizando el paradigma *publicador-suscriptor*. Este paradigma es un patrón de diseño para transferir datos de un objeto a otro. Los objetos, conocidos como publicadores, pueden hacer que los datos estén disponibles emitiendo eventos *publicar*. Otros objetos, conocidos como suscriptores, pueden recibir datos mediante la suscripción de una función de devolución de llamada (*callback*) para publicar eventos. Para que el paradigma de publicación y suscripción funcione de manera efectiva, los suscriptores deben saber cómo interpretar los datos que reciben a través de las devoluciones de llamada. Esto se hace definiendo un formato para cada tipo de datos utilizado por el sistema, llamados *mensajes*. Un objeto que genera eventos de publicación sólo puede pasar un tipo de mensaje a sus suscriptores. Siguiendo esta restricción, al registrar una devolución de llamada con un publicador, el suscriptor reconoce que recibirá un mensaje particular. Aunque los eventos de publicación están limitados a un tipo de mensaje, un objeto puede alojar un número arbitrario de diferentes eventos de publicación. Los mensajes que pasan alrededor del sistema heredan de una clase de mensaje base. La herencia garantiza que todos los mensajes tengan el formato correcto y que contengan atributos comunes. La definición de nuevos tipos de mensajes se convierte en una tarea trivial al heredar de la clase de mensaje base y especificar qué campos de datos contendrá el mensaje.

La ventaja de seguir los paradigmas de publicación y suscripción es que los publicadores no necesitan considerar cómo se manipularán los datos una vez que se hayan publicado. Del mismo modo, los suscriptores no necesitan considerar cómo se generan los datos antes de que sea publicado. Además de promover objetos débilmente acoplados y reutilizables, el paradigma publicador-suscriptor

también permite un software impulsado por eventos. El software escrito para leer la información del sensor puede hacer que los datos estén disponibles e impulsar la actividad dentro del sistema usando eventos de publicación. Las devoluciones de llamada diseñadas para procesar los datos del sensor sólo se ejecutarán una vez que los datos estén disponibles a través de los eventos de publicación.

2.2.2. Comunicación entre procesos

El paradigma publicador-suscriptor descripto permite una colección de objetos débilmente acoplados para compartir datos dentro de un solo proceso. En un sistema complejo, no es deseable manejar todas las funcionalidades de esta forma, ya que crea un único punto de falla que es difícil de diagnosticar. Al dividir el sistema en módulos asíncronos bien definidos, el sistema puede distribuirse.

Dado que los procesos operan en espacios de memoria independientes, el mecanismo de publicación y suscripción no se puede aplicar directamente. La comunicación entre procesos se debe usar para extender el paradigma de publicación-suscripción del nivel de objeto al nivel de proceso. Para transferir datos, los procesos deben compartir un mecanismo de comunicación común. Se puede utilizar cualquier biblioteca de comunicación que admita la distribución de datos de muchos a muchos, como multidifusión (*multicast*, en inglés) [40] o topologías en estrella [41], [42]. En el marco propuesto se usa multidifusión de datagramas con direcciones IPv6. Al usar la comunicación entre procesos para vincular procesos, se descentraliza la funcionalidad del sistema. Los procesos pueden ejecutarse en *hosts* heterogéneos, arquitecturas de *hardware* y sistemas operativos. La distribución del sistema a través de múltiples procesos también permite que el software aproveche un número cada vez mayor de núcleos de procesamiento disponibles para la ejecución simultánea de código. Además de ser eficiente, la estrategia multiproceso también mejora la integridad del sistema. Como a cada proceso se lo ha aislado y asignado sus propios recursos, si un proceso falla, no hará que todo el sistema falle. De manera similar, un proceso que consuma muchos recursos no bloqueará ni impedirá que otros procesos accedan a una parte equitativa de los recursos del sistema.

La topología de la red de comunicación está estructurada de modo que cada

tipo de mensaje está asociado con un URI[5] específico. Nuevamente, esta restricción de uno a uno hace explícita la estrategia de paso de mensajes. Al escuchar transmisiones en un URI particular, el oyente ha reconocido que recibirá un tipo de mensaje particular. Aunque existe un mapeo uno a uno entre el URI y el tipo de mensaje, puede existir una relación de varios a varios entre los organismos de difusión y los oyentes. Múltiples procesos pueden transmitir a un sólo URI, mientras que muchos procesos escuchan las transmisiones. La capacidad de suscribirse y publicar datos en un pequeño conjunto de URIs conocidos hace que la creación de una relación muchos a muchos entre los organismos de difusión y los oyentes sea simple y fácil de mantener. La simplificación de la capacidad de crear redes complejas de tráfico de datos promueve el desarrollo de sistemas capaces de recopilar grandes volúmenes de datos. En un sistema que contiene múltiples tipos de mensajes, los procesos necesitarán identificar qué mensajes están disponibles y dónde ubicarlos. La topología de red está disponible para el sistema a través de una especificación de mensaje que se correlaciona desde el tipo de mensaje a un URI. Los procesos que deseen participar y cumplir con el tráfico de red deben cumplir con la especificación. Para eliminar la ambigüedad de los mensajes que se generan en una topología de muchos a muchos, los organismos de difusión y los oyentes pueden usar temáticas durante la transmisión para filtrar los mensajes. Por ejemplo, en un vehículo equipado con cámaras orientadas hacia adelante y hacia atrás, los mensajes de imagen podrían ser desambiguados asociando emisiones con los temas “adelante” y “atrás”, respectivamente.

2.2.3. Registro de datos

El análisis de datos es esencial para comprender las características de los escenarios naturales de conducción y desarrollar nuevos algoritmos. Como resultado, la recopilación de datos de calidad es fundamental para facilitar la investigación y los avances en las aplicaciones de ITS. Uno de los principales objetivos del *hardware* utilizado es proporcionar un mecanismo para registrar datos. Esto se hace usando la arquitectura de *software* flexible descripta en la Sección 2.2.1.

Como se describe en la Sección 2.2.2, es posible que los procesos identifiquen

[5]Identificador de Recursos Uniforme - *Uniform Resource Identifier*-

qué mensajes están disponibles en el sistema y dónde ubicarlos. Esto hace posible que cualquier proceso acceda al tráfico de red de todo el sistema. Dado que hay un mapeo uno a uno entre el tipo de mensaje y el URI, todo el tráfico de la red puede registrarse en tiempo real suscribiendo un oyente a cada URI en el sistema. Cuando se recibe un mensaje, el oyente registra el tiempo que ha transcurrido desde el inicio de sesión, la temática de difusión y la carga útil del mensaje en un archivo de texto sin formato. Esta estrategia da como resultado un archivo de registro para cada tipo de mensaje transmitido dentro del sistema.

Los datos de alta frecuencia o de abundante cantidad, como las mediciones inerciales y las imágenes, pueden hacer que los archivos de registro crezcan rápidamente de tamaño. Los grandes archivos de datos no son propicios para la recolección oportunista de datos. Los vehículos sólo pueden estar dentro del alcance de una RSU[6] durante unos segundos, lo que imposibilita la transferencia de archivos de registro grandes. Para que los archivos de registro sean aptos para la recolección oportunista, pueden dividirse por el número de entradas de datos, por tiempo, o por ambos. Para evitar que se graben datos cuando el vehículo no está circulando, el sistema de registro sólo se inicia cuando el motor está en funcionamiento.

De manera similar, el sistema de registro finaliza cuando el motor se apaga. Al tiempo que promueve el uso eficiente de los recursos, este sistema de registro automático también es conveniente ya que no depende de un operador humano. Los conjuntos de datos se definen por los eventos de encendido y apagado del motor. Cada conjunto de datos se almacena en un directorio creado recientemente, con un sello de tiempo.

2.2.4. Comparación con otros sistemas similares

Los puntos motivadores en el diseño del sistema fueron la extensibilidad, la facilidad de mantenimiento y la confiabilidad. A la red troncal de comunicaciones del sistema se la llamó MCL y está públicamente disponible en [44]. Antes de desarrollar este software se consideraron las bibliotecas LCM [40], ZMQ [43], ROS [42] y RabbitMQ [41] para analizar sus fortalezas y debilidades.

[6]Unidad de Carretera -*Road-Side Unit*-

Una comparación de las diferencias claves entre las bibliotecas de comunicación se muestra en la Tabla 2.1. Estas bibliotecas se pueden dividir en dos categorías: centralizadas y descentralizadas. ROS y RabbitMQ son similares en el sentido que necesitan un servicio central para facilitar el enrutamiento de mensajes. El problema con estas arquitecturas es que el servicio central forma un único punto de falla que puede deshabilitar la comunicación dentro del sistema. Si bien agregar nuevos procesos o mensajes es relativamente trivial, agregar nuevos nodos al sistema requiere una configuración adicional. Cada máquina que se une a la red necesita conocer tanto la especificación de un mensaje como la ubicación del servicio central. En nuestra arquitectura de software, los nodos que se unen a la red sólo necesitan confirmar la especificación del mensaje y no necesitan realizar ninguna configuración. Al reconocer que cada tipo de mensaje está asociado de forma exclusiva con una dirección de multidifusión, los nuevos nodos pueden participar simplemente uniéndose a la red.

Tabla 2.1: Comparación de bibliotecas de comunicación.

	MCL	LCM	RabbitMQ	ROS	ZMQ
Estructura	DD	DD	AdM	DC	SCM
Transporte	UDP	UDP	TCP	TCP	TCP
Topología	P/S	P/S	Varios	P/S	Varios

Referencias:

- DD : Directorio descentralizado
- DC : Directorio centralizado
- AdM : Agente de Mensajes
- SCM : Socket como cola de mensajes
- P/S : Publicador/Suscriptor

ZMQ es una biblioteca de comunicación de alto rendimiento que proporciona una interfaz tipo *socket* para transmitir y recibir datos. Uno de los objetivos fue implementar un sistema de comunicación donde las conexiones de muchos a muchos son posibles. Es decir, muchos emisores pueden publicar datos en la red a muchos consumidores. Mientras que ZMQ permite que muchas conexiones

escuchen datos en un *socket* en particular, sólo un proceso puede vincularse a un *socket* y publicar. Para permitir una topología de muchos a muchos, se deben implementar redes complejas, un servicio de directorio o intermediario. La implementación robusta de estas soluciones es una tarea desafiante. El sistema presentado es más similar a LCM. En LCM, los mensajes fuertemente escritos, que no admiten cambios en su formato, se transmiten mediante multidifusión UDP[7] IPv4. En nuestra arquitectura, los mensajes débilmente, que sí admiten algunos cambios, escritos se transmiten utilizando multidifusión UDP IPv6. Al utilizar multidifusión, ambas bibliotecas son robustas en el sentido de que no existe un sólo punto de falla. En lugar de confiar en intermediarios o servicios de descubrimiento, la multidifusión permiten a cualquier proceso escuchar o transmitir datos en una dirección particular. Si un proceso en particular se bloquea, no impedirá que otros procesos se comuniquen. Un punto de diferencia entre las bibliotecas es que nuestra biblioteca es capaz de transmitir mensajes débilmente escritos donde se puede transmitir cualquier objeto serializable. Si bien nuestro software permite definir mensajes con campos obligatorios, también es posible crear y transmitir nuevos campos dinámicamente. En LCM, los mensajes están fuertemente escritos por lo que es imposible agregar dinámicamente nuevos campos a un mensaje.

Las bibliotecas de comunicación también se pueden categorizar por el protocolo utilizado para transmitir datos. ZMQ, ROS y RabbitMQ usan TCP[8] para transmitir datos. MCL y LCM usan UDP para datos de multidifusión. TCP es un protocolo más robusto, ya que garantiza la transmisión y el orden de entrega, mientras que UDP no lo hace. Por otro lado, UDP ofrece una latencia más baja y un modelo de transmisión sin conexión. El modelo de transmisión sin conexión es lo que permite a las bibliotecas de comunicación basadas en UDP operar de forma descentralizada.

[7] *User Datagram Protocol*
[8] *Transmission Control Protocol*

2.3. Plataforma de visualización y simulación de datos

La plataforma de simulación propuesta es una herramienta que puede ser utilizada en investigación a los fines de analizar la información registrada y evaluar diferentes algoritmos. La misma se compone principalmente de un ambiente virtualizado semejante al de la realidad donde fue recolectada la información de interés o donde se desee realizar experimentos. El ambiente virtual, recreado mediante diversas técnicas, es potenciado mediando el uso de un motor de videojuegos, cuya plataforma de desarrollo es apta para los requerimientos de este tipo de simuladores debido a las prestaciones que presentan.

2.3.1. Virtualización del ambiente

Para el desarrollo de entornos virtuales semejantes a los de la realidad es posible utilizar diversas técnicas. Una de ellas es a partir de software dedicado que crea edificios aleatorios basados en reglas pre-establecidas por el usuario, utilizando modelos y texturas estándar, como por ejemplo *CityEngine*. Otra es la recreación de los modelos edilicios de modo independiente, construyendo cada modelo individual y manualmente para que se asemejen lo más posible a los de la realidad, como *SketchUp* de *Google*. Por último, otro procedimiento es procesar información a partir de una secuencia de imágenes en 2D para reconstruir la estructura en 3D con algoritmos como *Structure From Motion* (SFM) [45].

Con la técnica de SFM es posible recrear un objeto tridimensional tomándole fotografías desde diferentes ángulos. Para encontrar la correspondencia entre las imágenes, se determinan de imagen a imagen algunas características que son comunes y que poseen gradientes en múltiples direcciones (como esquinas). Uno de los algoritmos más utilizados para la detección de estas características es SIFT[9] [46], el cual identifica aquellas que son invariantes a la escala, la rotación y los cambios de iluminación de las imágenes. Otro algoritmo popular es SURF[10] [47], el cual realiza la suma de los componentes de los gradientes y de sus módulos.

[9] Transformación de característica invariante a la escala -*Scale-Invariant Feature Transform*-
[10] Funciones Robustas y aceleradas -*Speeded-Up Robust Features*-

Para correlacionar y unir los puntos característicos de cada imagen se utiliza el algoritmo de Lukas-Kanade [48]. La técnica también es aplicable a la reconstrucción de infraestructura y se utiliza a menudo para el modelado de ciudades, tanto de sus edificios como de sus fachadas [49] [50] [51], así como también de ambientes donde se encuentren objetos en movimiento, como vehículos [52].

En este trabajo, el modelado 3D de los caminos, fachadas, edificios, señales, etc. fue creado en base a un mapa digital obtenido desde OSM[11]. OSM [53] es un proyecto colaborativo para crear mapas libres con la posibilidad de exportar los datos y trazas capturada con dispositivos GPS móviles, ortofotografías y otras fuentes libres. Debido a que este proyecto no posee imágenes de radares, las cuales fueron usadas como base para estampar el suelo del modelo virtual, también se utilizó el servicio *Google Maps* [54] para hacer uso de ellas.

Para comparar la similitud entre ambas fuentes cartográficas, se realizó un breve análisis comparativo utilizando coordenadas GPS de los lugares donde frecuentemente los vehículos circulan para recolectar datos. En la Figura 2.1 se puede observar la ubicación de coordenadas en ambos mapas. Cada fila de figuras corresponde a una coordenada en particular y cada columna a un servicio de mapas. La primera de ellas está basada en los mapas de *Google Maps* mientras que la segunda lo está en los de OSM.

De la comparación resulta un pequeño desplazamiento que no es relevante para el fin de la utilización de los mapas como imagen base. Además, el corrimiento está dado en los mapas de *Google Maps* y no en los de OSM que son en los que se presentan las trazas GPS de los vehículos utilizados.

Por otra parte, mediante el uso del software *CityEngine*, se recrearon los edificios de la ciudad. Sus fachadas se hicieron semejantes a las de la realidad utilizando fotografías extraídas desde *Google StreetView* y desde las imágenes de video adquiridas por las cámaras del vehículo. Los edificios característicos o relevantes fueron modelados con mayor detalle utilizando el software *SketchUp*.

También se virtualizó parte de la ciudad utilizando la técnica de SFM. Para esto se utilizaron imágenes de las cámaras montadas en el vehículo, de las cuales se extrajeron fotogramas a una velocidad de cuadros menor a la que normalmente

[11] *OpenStreetMap*

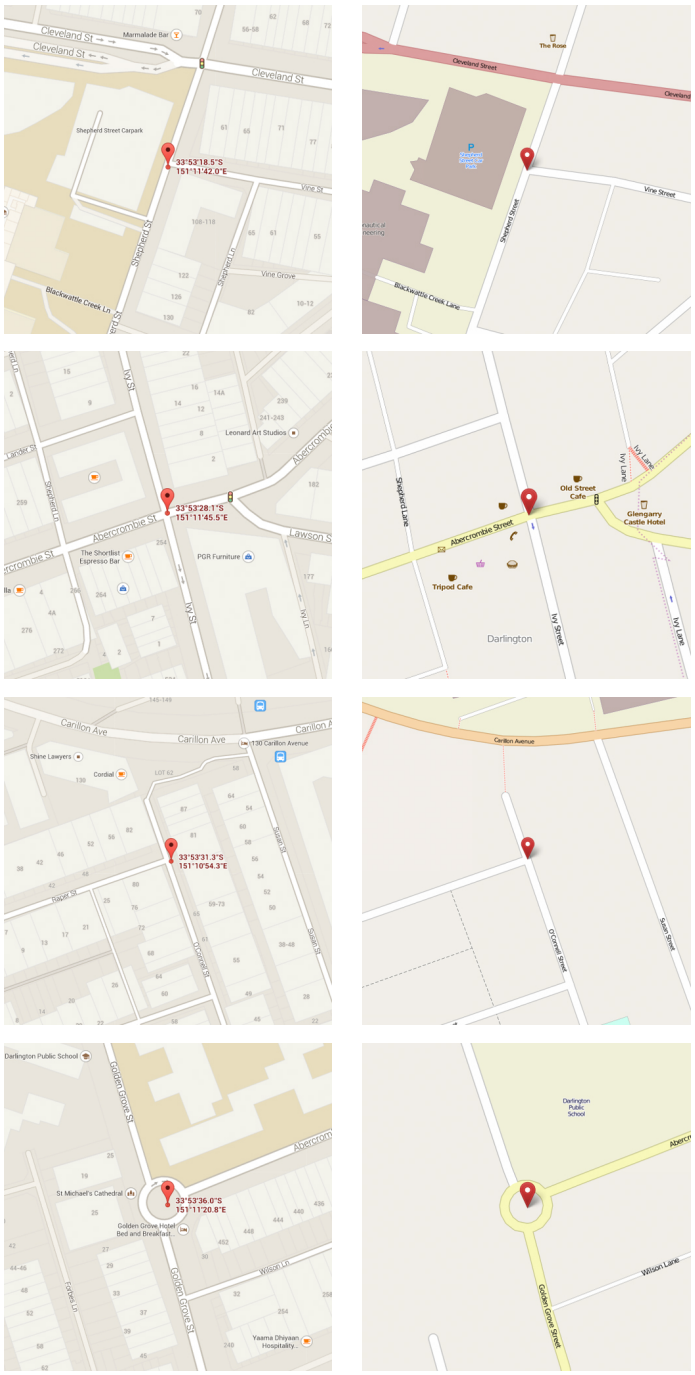


Figura 2.1: Diferentes puntos GPS comparados en Google Maps (columna izquierda) y OpenStreetMap (columna derecha).

captura la cámara y se generó una nube de puntos con la forma del entorno. La Figura 2.2 muestra una de las vistas del entorno modelado.

Este procedimiento brinda una ventaja extra ya que el modelo del entorno es completado con nuevos puntos cada vez que el vehículo toma imágenes de ese lugar. Además, el modelo es ampliado cuando se recorren nuevos lugares que no fueron transitados con anterioridad o se circula en el sentido contrario a la toma

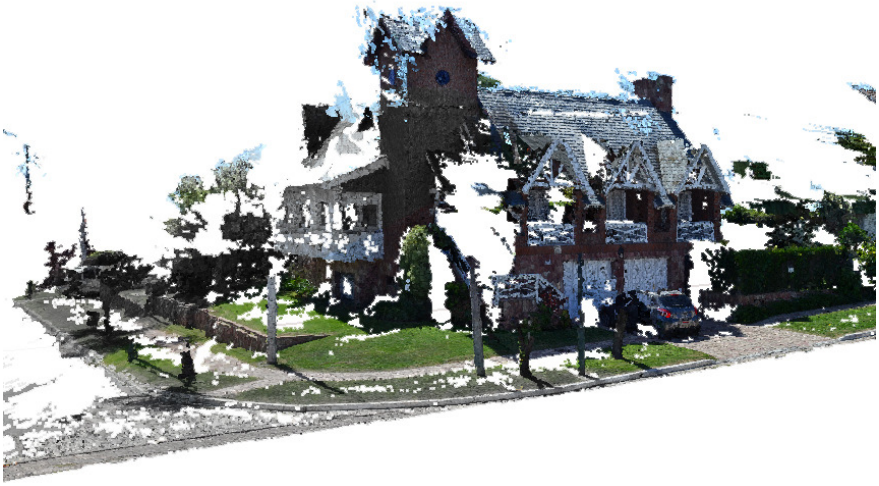


Figura 2.2: Nube de puntos densa, obtenida mediante la técnica de SFM, la cual permite modelar el entorno en 3 dimensiones.

previa. Para completarlo, los puntos de la nube se unen con líneas y se forman superficies que hacen del modelo un objeto físico en el mundo virtual.

La técnica presenta problemas cuando se trata de virtualizar ciertos tipos de texturas como la de arbustos, paredes lisas y lugares donde la iluminación es escasa. En la Figura 2.3 se puede observar, en la casa modelada que se encuentra frente al vehículo, la presencia de una zona donde no se ha podido construir el modelo. Este problema surge debido a la ausencia de puntos relevantes o poco visibles (lugares con sombra por ejemplo) entre las imágenes de esa zona, los cuales no pueden ser obtenidos por el algoritmo y por lo tanto éste arroja como resultado una parte del modelo incompleta.

Por ello, en la construcción del entorno se utilizaron las primeras dos estrategias descriptas, mientras que la basada en SFM se utilizó sólo a modo de evaluación debido a los problemas mencionados y que no son controlables en una captura de datos realista.

2.3.2. Motor de videojuegos para simulación

El entorno virtual obtenido, sumado a las funcionalidades brindadas por los motores de videojuegos actuales, brinda el soporte necesario para la creación del tipo de simulador que se propuso para fines de investigación.

Un motor de videojuegos es un software diseñado para la creación y desa-

Figura 2.3: Modelo de puntos con la superficie creada.

rrollo de videojuegos. Su núcleo incluye motores de representación (*rendering* en inglés) para gráficos 2D y 3D, motores de física, de sonidos, manejo de *scripts*, animaciones, inteligencia artificial, empleo de redes, *streaming*, *threading*, etc.

Debido a su capacidad para el procesamiento y representación de información, estos motores han sido utilizados como herramientas de visualización [55]. También se los utiliza para simulación e investigación ya que proveen un ámbito apropiado para el desarrollo de algoritmos en tiempo real. Las potencialidades brindadas por el motor, para las aplicación que se presenta, se describen en la Sección 2.3.2.1.

El modelo virtual obtenido con las técnicas de virtualización se introdujo dentro del motor de videojuegos, al cual también se le incorporaron modelos de vehículos, peatones, señales de tránsito, semáforos y vegetación. El motor utilizado es *Unity3D*, el cual admite el uso de cascos de realidad virtual para lograr que el usuario reviva la escena de forma más realista y se encuentre completamente inmerso en ella.

2.3.2.1. Potencialidades

Los motores de videojuegos brindan funciones que cubren todas las áreas de la computación, desde la implementación de algoritmos hasta el procesamiento de imágenes, pasando por el manejo de datos, la integración de software y hardware,

etc. Las potencialidades brindadas por estos motores como núcleo de un simulador son un punto clave para la puesta en marcha de todas las características necesarias para cumplir los objetivos planteados en esta sección.

Tanto la lectura de datos (o ingreso de ellos al simulador) como la salida (o datos producidos) pueden llevarse a cabo por diferentes medios. Los dispositivos externos como teclado, ratón, volantes o pedales pueden conectarse mediante puerto serie o USB. También es posible realizar lecturas o escrituras en archivos de texto locales en formato .txt y .csv, o desde la red utilizando direcciones IPv4 e IPv6.

La visualización de la simulación es un punto importante para la interpretación de la información provista. La ejecución de las aplicaciones en más de un monitor facilita la creación de simuladores que cubren el campo de visión de una persona. Esto se lleva a cabo utilizando una vista extendida que abarca las resoluciones de los monitores que se utilizan. También, mediante la incorporación de *plugins*, se logra la visualización en otros tipos de pantallas como cascos de realidad virtual, por ejemplo *Oculus Rift* y *Google Cardboard*.

Además, es posible visualizar simultáneamente lo que está ocurriendo dentro del mundo virtual desde diferentes puntos de vista. Estos puntos de vista se denominan *cámaras de simulación* y a diferencia de las cámaras que obtienen imágenes o graban video, son objetos virtuales que se utilizan dentro del simulador para mostrar el mundo virtual al usuario. Las cámaras se colocan en diferentes posiciones, sea de forma estática (en un objeto de la infraestructura del ambiente virtual) para observar un área particular del ambiente virtual, o adosada a un objeto dinámico que se desplaza por la escena, para visualizar acciones dinámicas. A la imagen de cada una de estas cámaras debe corresponderle una vista para su representación.

La pantalla principal puede dividirse en subpantallas facilitando la función de multi-vista. Además, en cada una de estas vistas se pueden mostrar imágenes de video o fotos de forma independiente a lo que esté haciendo el motor. Las diferentes pantallas también se utilizan para presentar gráficas de funciones o de datos mediante bibliotecas gráficas disponibles en el entorno de programación. Diferentes tipos de figuras (círculos, cuadrados, botones, etc.) también están

disponibles para utilizarlas con el fin de resaltar objetos o crear menúes.

Para construir el mundo virtual es posible crear y/o importar objetos 3D desde otros entornos como *Blender*, *Autocad*, *SketchUp*, *MakeHuman*, etc. y así completar la escena con vehículos, personas, señales y otros objetos que no son modelados con las técnicas mencionadas en la sección anterior.

Las escenas se componen de diferentes objetos. Los edificios, el asfalto, los vehículos, peatones, entre otros, son objetos. Cada objeto de la escena admite diferentes tipos de componentes como texturas, *scripts*, cámaras, etc. Estos acompañarán siempre al objeto sin afectar al resto de los mismos. Los objetos son entes independientes y pueden funcionar, si es necesario, sin interactuar con el entorno que los rodea.

El motor implementa el uso de *barreras invisibles* que, al ser atravesadas por objetos de la escena, disparan eventos y ejecutan algoritmos. Las barreras son utilizadas para lanzar acciones en momentos determinados y personalizar la escena de acuerdo a la experimentación que se este desarrollando. Un ejemplo claro es que un peatón comience a cruzar la calle cuando el vehículo evaluado se aproxima a la esquina, o el caso inverso, que un automóvil se aproxime cuando el peatón quiere cruzar la calle. Otros tipos de eventos pueden ser personalizados, entre los más importantes se encuentran el cambio de luces de los semáforos, el movimiento de personas, animales u objetos en la escena y el de vehículos que completan el tránsito de una ciudad. El cambio dinámico de colores, texturas, tamaño, posición y demás propiedades de cada objeto de la escena son modificados mediante *scripts*. *Scripts*, eventos y rutinas se programan en lenguaje *Javascript*, *C#* o *Boo*. Cada fragmento de código agregado a la escena es ejecutado independientemente del resto y tiene la capacidad de conectarse con otros para compartir datos.

Finalmente, el programa compilado se exporta a plataformas de escritorio (como *Mac*, *Windows* y *Linux*), plataformas móviles (*Windows Phone*, *iOS* y *Android*), plataformas de juego (*Playstation*, *Wii* y *Xbox*), televisores (*Android TV*, *Samsung Smart TV* y *tvOS*), ambientes de realidad virtual (*Steam*, *Playstation VR*, *Gear VR* y *Windows Mixed Reality*), de realidad aumentada e implementación web.

2.3.3. Simulador para aplicaciones de ITS

El sistema desarrollado en este capítulo utiliza el modelo 3D de un entorno basado en mapas y fotos para representar el ambiente donde se desarrolló la recolección de datos experimentales. Esta representación asimila el mundo virtual al real para el posterior análisis de los resultados.

Gran cantidad de datos provenientes de distintas fuentes alimentan al motor de juego. Este flujo es recibido a través de las interfaces mencionadas (IPV6, IPV4, etc.) que facilitan su acceso sin restricciones o limitantes de volumen y/o temporizado para el tipo de aplicaciones buscadas (muestreo de sensores del vehículo o del entorno). Los grandes volúmenes de información son controlados en parte por el usuario, administrando el esquema de visualización disponible en el entorno. Una representación de estas interacciones se muestra en la Figura 2.4, donde se ve como el motor de videojuegos recibe distintos tipos de información para procesarla con algoritmos internos y luego visualizar los resultados en pantalla.

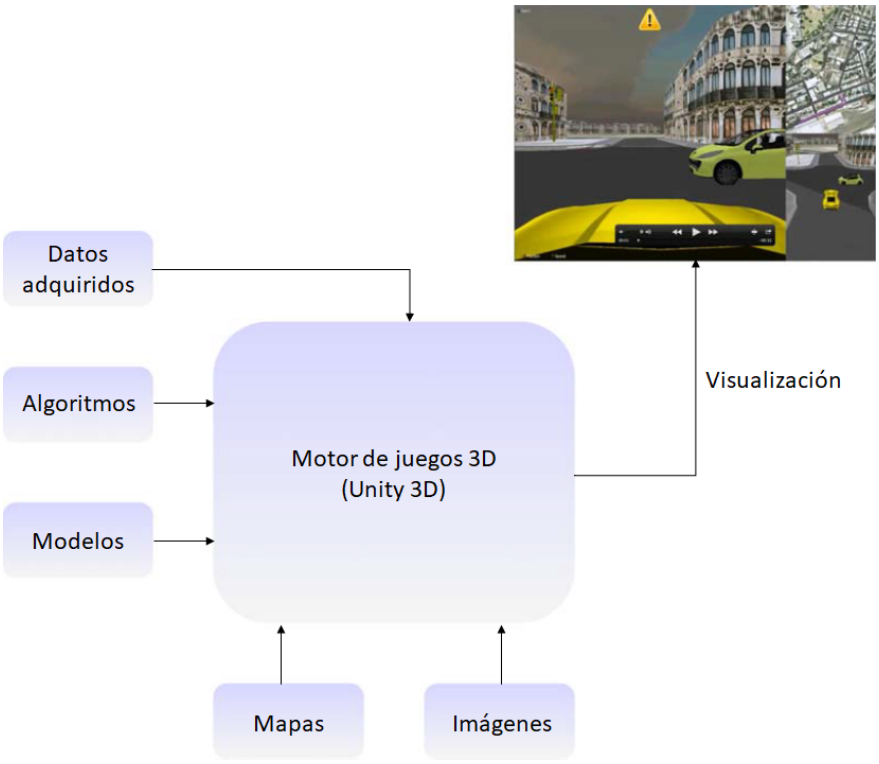


Figura 2.4: Representación del simulador en diagramas de bloques.

Para facilitar la comprensión de los datos recibidos desde los vehículos, estos son visualizados en la escena de diferentes modos dependiendo del tipo de datos

y de lo que se quiere destacar de ellos. Por un lado, algunos tipos de datos son representados por medio de gráficos de barras, relojes, texto, etc., tales como la velocidad del vehículo, la presión ejercida sobre el pedal del freno y el tiempo transcurrido. Otros, sin embargo, requieren de una mejor representación para su comprensión. Por ejemplo, la posición de uno o más vehículos es visualmente más atractiva y mejor comprendida si se representa sobre un mapa y no simplemente mostrando las coordenadas como texto, el cual cambia a gran velocidad y no es amigable a la lectura. De este modo, y utilizando algoritmos que se ejecutan en el momento de la reproducción de la escena, se destacan diferentes objetos del entorno o sus características, en un determinado momento. Gracias a la ejecución de multitareas que permite el motor, la velocidad de un vehículo que se encuentra frente a otro se visualiza utilizando un gráfico de barra sobre el mismo o coloreándolo de acuerdo a su velocidad actual. También se muestra otro tipo de variables como, por ejemplo, cuando ese vehículo ha comenzado a frenar.

Es de interés también destacar otro tipo de figuras del tráfico. Las sendas peatonales y las señales de tránsito muchas veces no son rápidamente visualizadas. Destacar este tipo de información adquirida desde cámaras montadas en el vehículo es muy importante para la asistencia al conductor y el análisis de la escena y del entorno. También se remarcan señales pintadas sobre el asfalto que se encuentran desgastadas o señales que se encuentren ocultas o ausentes, pero que son obtenidas por medio de mapas digitales.

Mientras la escena se está reproduciendo, ésta se puede visualizar desde distintos puntos de vista, lo que significa que se la observa, en el mismo instante, desde distintos ángulos. Por ejemplo, puede ser observada desde los ojos de un peatón, desde la posición de otro vehículo o en tercera persona (Figura 2.5). Con la llamada “vista ocular”se visualiza la escena como si la estuviera viendo el conductor del vehículo que está en movimiento, o una persona que camina por la vereda o un conductor situado en otro vehículo. La “vista dinámica”, en tercera persona, se refiere, por ejemplo, a una vista donde el vehículo sea el protagonista, ubicando la cámara detrás del mismo, la cual se desplaza dinámicamente acorde lo haga el vehículo. También puede optarse por una cámara libre, que permite al usuario desplazarse por la escena con total libertad, sin estar sujeto a lo que

sucede en la misma. Por último, con la "vista estática"se observa el desarrollo de la escena desde puntos fijos como esquinas, semáforos y otras posiciones convenientes para el análisis de la escena. Esto contribuye a ampliar la visión del intérprete y es clave en la comparación del hecho desde las diferentes ópticas.

Figura 2.5: Diferentes puntos de vista de la misma escena.

Otra perspectiva interesante para visualizar es una cámara de simulación que se corresponda con una cámara real del vehículo. Esto permite comparar la reproducción del entorno virtual con el video obtenido, como se observa comparando la vista principal, tomada con una cámara montada en el centro del vehículo, detrás del parabrisas, con la vista central derecha, que es una imagen real de una cámara montada en el mismo lugar que la virtual, de la Figura 2.6.

Figura 2.6: Comparación de imágenes y otras vistas. La vista principal pertenece a una cámara montada en el vehículo del simulador; la vista superior derecha es un mapa de la ciudad donde se muestra el vehículo y la trayectoria recorrida; la vista central derecha es la imagen de una cámara real, montada en el vehículo que capturó los datos; y la vista inferior derecha es una vista en tercera persona que permite observar la actividad del simulador desde cualquier punto fijo.

En la misma figura se observa como la utilización de multi-vistas ayuda a entender el entorno desde diferentes ópticas. Para este caso, en la imagen se muestra, además de la comparación antes mencionada, otra vista adicional en

tercera persona (abajo a la derecha) que permite al usuario ampliar la vista a los alrededores del vehículo de interés y un mapa (arriba a la derecha) de la ciudad real. En este mapa se muestra el recorrido que está llevando adelante el vehículo, marcado en color fucsia.

Otro ejemplo puede visualizarse en el video disponible en [56], donde la comparación entre la cámara real y la montada en el vehículo se observa en las vistas superiores, sobre el márgen derecho. La primera de ellas corresponde al mundo virtual, mientras que la que se encuentra por debajo de ella reproduce las imágenes de una cámara real.

La interacción con el usuario se hace en tiempo real. Existen dos tipos de interacción de acuerdo a si el simulador está reproduciendo datos o generándolos. Para el caso de la reproducción, algunas de las funciones disponibles son mostrar u ocultar la información visualizada en pantalla, agregar eventos a la simulación, cambiar ángulos de visión de la escena, etc. Cuando el simulador está generando datos, el usuario puede conducir un vehículo, generar eventos que permitan a otros objetos interactuar en la escena (mover otros vehículos, personas cruzando la calle, cambio de luces de un semáforo, etc), etc.

Debido a la gran cantidad de datos involucrados, el simulador incorpora *scripts* de código que utilizan estos datos para procesar y/o evaluar diferentes algoritmos. De este modo algunos algoritmos ayudan a la detección de objetos como señales de tránsito, vehículos o personas, otros generan alertas al conductor dependiendo del entorno en el que se encuentre, así como algoritmos que son capaces de realizar una conducción semi-automática.

Por ejemplo, ante la detección de situaciones de riesgo, si un vehículo se aproxima y puede interferir en el camino del vehículo de interés generando una situación de riesgo, el conductor es advertido. En esta ocasión, el simulador dibuja sobre el vehículo de riesgo un símbolo y emite una señal sonora que advierte al conductor de dicha situación como se muestra en la Figura 2.7.

El usuario tiene la posibilidad de desplazarse por la escena con un vehículo conducido por él mismo. Para este caso resulta de interés la elaboración de experimentos en los que el entorno pueda ser controlado. Controlar el entorno significa que el simulador cuenta con la capacidad de crear ambientes virtuales

Figura 2.7: Símbolo sobre el vehículo para indicar algún tipo de peligro. Este tipo de advertencias son el resultado de algoritmos de prevención evaluados en el simulador.

que se ajusten y adecúen a las pruebas que se desean realizar, que por algún motivo no se pueden llevar a cabo en la realidad. Por ejemplo, si se desea prestar atención a la concentración de un conductor mientras conduce, se puede evaluar en primer medida un camino limpio, sin elementos que colaboren a la distracción del conductor y a medida que se desarrolla el experimento se pueden ir agregando distracciones para generar los efectos esperados.

La realización de experimentos para estudiar cómo las personas procesan y reaccionan frente a estímulos visuales relacionados con la conducción y el estudio de patrones de atención visual de los conductores se llevan a cabo con la integración de equipamiento como *eye-trakers*, EEGs[12], ECGs[13], etc., admitidos por el simulador. Sumado a esto, es posible agregar nuevos eventos que no sucedieron en la realidad. Los eventos pueden ser algoritmos que agreguen dinamismo a algunos objetos del entorno, como vehículos o peatones; el cambio de las condiciones climáticas, etc. Así es posible modificar la escena para analizar reacciones del conductor y el funcionamiento de algoritmos de seguridad y percepción del entorno.

Por otro lado los datos generados por algoritmos propios obtenidos fuera del simulador pueden contrastarse en tiempo real con los de la simulación. Estos datos pueden ser evaluados una y otra vez gracias a la reproducción indefinida

[12]Electro-Encefalogramas
[13]Electro-Cardiogramas

de la escena. A partir de ellos, por ejemplo, se infieren las intenciones de los conductores cuando son evaluados en situaciones controladas dentro del entorno. Los resultados obtenidos son nuevamente incorporados a la base de datos.

2.4. Conclusión del capítulo

En esta sección se presentó una arquitectura modular para la adquisición y el almacenamiento de datos en aplicaciones ITS. El diseño se basa en un paradigma de publicación y suscripción que utiliza la multidifusión IPv6 para la comunicación entre procesos. Esta estrategia permite dividir el código en diseños modulares que se pueden ejecutar en muchos procesos independientes. El mensaje de publicación y suscripción que pasa por la red permite que la comunicación se produzca de forma transparente en una única computadora, en múltiples computadoras y en plataformas heterogéneas que ejecutan lenguajes de programación diferentes.

La arquitectura de software descrita se usa tanto para la adquisición de datos del vehículo como para la adquisición de datos fuera del vehículo. El sistema de adquisición de datos en el vehículo registra automáticamente las secuencias de datos y los almacena en archivos, que luego son cargados automáticamente a un servidor central e insertados en una base de datos. El diseño combinado de software y hardware permite registrar datos complejos de vehículos y almacenarlos sin intervención humana. La naturaleza automática de este diseño hace que el sistema sea ideal para recopilar grandes cantidades de información.

Cuando se compara con ZMQ, LCM, ROS y RabbitMQ, queda en claro que la arquitectura propuesta ofrece una buena combinación de alta velocidad de ancho de banda y transmisión de baja latencia, como se describe en la Sección 2.2.4. Debido al diseño del sistema descentralizado, es resistente a las fallas y es fácil de escalar en múltiples máquinas con una configuración mínima. Por ello que los paradigmas de diseño recomendados en la arquitectura fomentan el desarrollo de código transparente, extensible y fácil de mantener.

Los datos experimentales utilizados en esta tesis se recolectaron utilizando la arquitectura descripta, en pruebas realizadas en la ciudad de Sídney, Australia.

Para la misma se utilizaron vehículos equipados con diferentes tipos de sensores como cámaras de video, acelerómetros, magnetómetros, velocímetros, GPS, LIDAR[14], entre otros. Además se incorporaron datos de otras fuentes como el nombre de las calles obtenidos desde mapas virtuales y la determinación del tiempo a partir de relojes externos. Los vehículos recorrieron parte de la ciudad siguiendo un camino establecido previamente. Los datos fueron almacenados en una base de datos y transmitidos mediante direcciones IPv6 al simulador, en el cual se realizaron diferentes ensayos para evaluar la capacidad de simulación, obteniendo resultados prometedores.

Los resultados obtenidos en cuanto a funcionalidades demuestran la ductilidad del motor de videojuegos para la construcción de un simulador que permita no sólo la recreación y reproducción de escenas para el análisis de la información sino también la evaluación de algoritmos que no pongan en riesgo personas y equipos, como sería el caso de aquellos para evitar colisiones. A su vez, se obtienen nuevos datos de interés relacionados a la comprensión del entorno e intención del conductor a partir de la posibilidad de generar experimentos controlados en contraposición con ambientes abiertos donde pueden introducirse distractores que permitan demostrar las hipótesis planteadas.

La virtualización del ambiente se lleva a cabo satisfactoriamente, imitando el realismo necesario con el entorno real, con excepciones de zonas incompletas a causa de fallas en el algoritmo de SFM.

También es importante destacar que el simulador permite el aprovechamiento de los datos almacenados, los cuáles pueden ser utilizados reiteradas veces con la ventaja de realizar la captura de los mismos por una única vez.

Por último, si bien la plataforma de simulación fue desarrollada y evaluada utilizando información real recolectada por vehículos, el uso de sus funcionalidades es mas provechoso para la evaluación de situaciones riesgosas. Al momento de finalizada esta tesis, el simulador no fue utilizado mas allá que para la reproducción de información, ya que aún no se ha comenzado con la evaluación de este tipo de situaciones.

[14] *Light Detection and Ranging*

Capítulo 3

Medición directa de la dinámica del peatón

3.1. Uso del sensor acelerómetro

Los equipos móviles actuales, como teléfonos, relojes y pulseras inteligentes, poseen diferentes tipos de sensores, como acelerómetros, giróscopos, GPS, etc. Originalmente, estos sensores fueron utilizados para lograr una mejor experiencia del usuario para con el dispositivo, pero gracias a la facilidad y accesibilidad con que las empresas fabricantes de los sistemas operativos permiten su acceso, los mismos se pueden usar para otros fines. Gracias a ellos es posible obtener distintos datos de movimiento, aceleración y geo-localización en tiempo real.

El GPS es una buena alternativa de uso para la estimación de la intención del peatón, pero su exactitud y temporizado no son suficientes como para hacer un seguimiento preciso, como se demuestra en [57] y si bien en [58] logran mejorar la exactitud, lo hacen a cambio de una frecuencia de adquisición de 5Hz, muy baja para este propósito. Además, en ciudades con edificios altos, la cantidad de satélites disponibles para hacer la localización disminuyen y esto hace aumentar aún más la imprecisión del dispositivo.

En tanto, algunos trabajos utilizan la información del acelerómetro de un teléfono para reconocer actividades como caminar, trotar, subir y bajar escaleras, sentarse o mantenerse parado [59]. La cantidad de información utilizada para mejorar la precisión de los resultados [60] también es un factor importante en la

correcta detección, así como el tipo de métodos utilizados para el reconocimiento [61]. Otros trabajos desarrollan mejoras mediante la utilización de varios acelerómetros, realizando fusión ponderada de la información obtenida [62]. En [63] se analiza la ubicación de acelerómetros en diferentes partes del cuerpo, como en la cadera, el brazo, la mano, la pierna, etc. Los resultados arrojaron que llevar el teléfono en el bolsillo del pantalón es la mejor opción porque es el lugar donde la información representa de mejor manera el movimiento del peatón. Esto da soporte a que en este ensayo se utilice dicho lugar para la ubicación del dispositivo adquisidor.

Para lograr el reconocimiento de las actividades es importante detectar los patrones que las identifiquen. En [64] se discute el reconocimiento de los patrones de movimiento, restringido a las situaciones de parada y cruce de calle con semáforo reglamentario, utilizando únicamente los datos provenientes del acelerómetro del teléfono móvil del peatón.

Para obtener estos resultados, la información debe ser procesada por algoritmos capaces de reconocer el tipo de actividad. Los algoritmos de clasificación son el medio más apropiado para esta tarea, por ello en la Sección 3.2 se realiza una breve descripción de los mismos.

3.2. Algoritmos de clasificación

En aprendizaje automático (conocido comúnmente como *Machine Learning*), la clasificación es el problema de identificar a que clase pertenece una nueva observación dadas observaciones previamente etiquetadas. Esto es posible gracias a que el entrenamiento previo del modelo se realiza con salidas conocidas para el conjunto de entradas que alimentan a la máquina. Este tipo de aprendizaje se denomina *Aprendizaje Supervisado*. Existen diferentes tipos de clasificadores, los más ampliamente difundidos son árboles, SVM[1], k-NN[2], entre otros. Su uso nos da como ventaja una salida discretizada, para este caso binaria (cruza/no cruza), que clasifica concretamente los datos. También es posible obtener salidas probabilísticas, que pueden ser útiles para otro tipo de reconocimiento.

[1] *Support Vector Machines*
[2] k-Vecinos Cercanos -*k-Nearest Neighbors*-

En este capítulo se utiliza como clasificador principal el de k-NN. Su funcionamiento se basa en la partición del espacio en igual cantidad de regiones como cantidad de clases existen en el entrenamiento. A cada una de estas regiones se le asigna la clase que con más frecuencia se encuentra entre las muestras de esa región. Luego del entrenamiento es posible obtener un modelo que sea capaz de predecir la clase de salida para una entrada determinada. Para ello se mide la distancia de la muestra ingresada al resto de las muestras ya etiquetadas y se escoge la clase de los vecinos más cercanos. En general, para la medición de esta longitud se utiliza la distancia euclidiana mostrada en la Función 3.1, donde x_i es la muestra ingresada al predictor y x_j es la muestra del predictor que ya fue etiquetada durante el entrenamiento. El total de las muestras entrenadas se denota con p.

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{p} (x_{ri} - x_{rj})^2} \tag{3.1}$$

Un clasificador k-NN es simple de implementar y comprender, y más aún en un problema como el que se plantea aquí que es de baja dimensionalidad como veremos en la Sección 3.3.2. Si bien no es una técnica adecuada para el reconocimiento de patrones, ya que requiere de mucha información para representar al clasificador, es útil en el análisis que presenta este trabajo. Sin embargo, en la Sección 3.4 se presentan resultados de otros clasificadores lineales como SVM, que sirven para complementar las conclusiones.

En el presente desarrollo se muestra una estrategia de detección basada en la envolvente de las variaciones de la aceleración de los movimientos de una persona al caminar. Adicionalmente, más que proponer un clasificador particular que resuelva el problema planteado, el objetivo es analizar si la información de movimiento medida con un acelerómetro sobre el propio peatón permite reconocer la intención de cruzar o no una calle antes que se produzca la acción. Por esto se plantea el uso de clasificadores simples tipo k-NN como fueron descriptos. En la Sección 3.3 se explica cómo se llevó a cabo el trabajo, detallando los equipos utilizados en la Sección 3.3.1. Para el análisis se diseñó un experimento especialmente preparado para permitir que las personas se muevan libremente en situaciones de tránsito y controlando algunas variables ajenas al experimento como se pre-

senta en la Sección 3.3.2. Por último, en la Sección 3.3.3 se expone el método para validar los resultados y en la Sección 3.6 se comentan las conclusiones del capítulo.

3.3. Información de aceleración

Como se mencionó anteriormente, en este capítulo se trata la detección de patrones de datos de aceleración y posterior clasificación de la intención mientras una persona camina y se aproxima a cruzar la calle. Pero antes de avanzar en la metodología utilizada, se definirá el concepto de intención utilizado, ya que es nombrado en este y en los próximo capítulos.

Definición del concepto de intención utilizado

Determinar la intención implica que primero se defina, en una secuencia temporal, qué significa intención. En la Figura 3.1 se muestra una interpretación temporal de los conceptos que se utilizan para determinar cual es el momento al que nos referimos cuando se habla de intención. Allí se representa con una línea negra el movimiento de una persona en el tiempo, la punta de la flecha indica el final de la acción, que puede ser la de cruzar o no la calle. En rojo se indica el momento en que el peatón desarrolla la intención que concluirá en la acción deseada. El momento entre la intención y la ejecución de la acción es referido en este capítulo como *marca de intención*, marcado en verde.

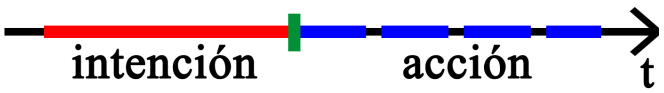


Figura 3.1: Momento de la intención (rojo-continuo), marca de intención (verde) y momento de la acción (azul-discontinuo) expresados en una línea temporal.

El momento posterior a la marca de intención es el momento donde se ejecuta la intención, referida en este texto como acción, el cual es marcado en la figura con una línea azul discontinua. La acción toma dos valores posibles, *cruza* o *dobla*. En el caso de *cruza*, la acción puede producirse en línea recta y/o curva, mediante la desviación de la trayectoria que tenía el peatón. Un ejemplo más representativo se muestra en la Figura 3.2. En ella se pueden ver tres tipos posibles de acción,

siendo dos de ellas coincidentes en la misma clase. La figura representa la vista superior de una esquina típica de una cuadra, donde se puede observar un área prohibida (área sombreada rayada correspondiente a la zona de construcción edilicia), la vereda y la calle (área sombreada lisa).

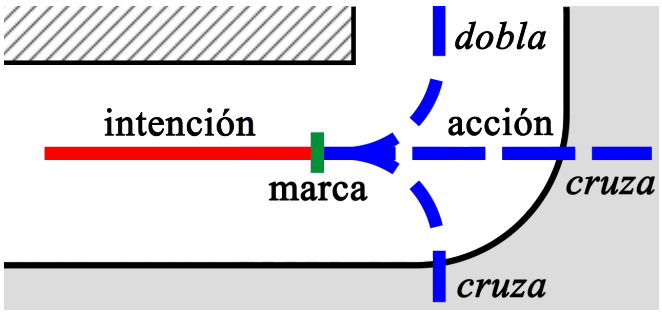


Figura 3.2: Momento de la intención (rojo-continuo), marca de intención (verde) y momento de la acción (azul-discontinuo) mapeados físicamente.

Cabe notar que el nombre de las acciones hacen referencia al tipo de intenciones que se estima del peatón, previo a realizar dicha acción.

3.3.1. Adquisición

La información utilizada en este trabajo es obtenida utilizando un teléfono móvil *Samsung Galaxy Core*, con sistema operativo *Android*, que fue portado por diferentes peatones. Al teléfono se le instaló una aplicación desarrollada por terceros que captura y almacena los datos en la memoria interna del dispositivo. Además se bloquearon las actualizaciones y sincronización del dispositivo, así como todo proceso que pudiera interferir en la toma de los valores. Estos últimos pasos se llevaron a cabo desde la interfaz de usuario del teléfono a fin de mantener el sistema operativo en funcionamiento y que aún pueda realizar las tareas de telefonía. El sensor disponible en el dispositivo es un BMC150 de Bosch Sensortec con una resolución de $0{,}009575196 m/s^2$ y un rango máximo de $156{,}88 m/s^2$. El teléfono se alojó en el bolsillo delantero del pantalón, ubicación elegida como mejor lugar para la recolección de este tipo de información, como se demuestra en [63]. Los datos de los ejes X, Y y Z, junto al tiempo de captura de los mismos, son almacenados a una frecuencia de $50 Hz$, para su posterior utilización en el entrenamiento del clasificador.

Acompañando al teléfono, se utilizó también un dispositivo de soporte con

una estructura de adquisición similar a la del teléfono a modo de resguardo de los datos capturados ante posibles interrupciones en la toma de los valores. Para la construcción de este segundo adquisidor se utilizó una placa con un microcontrolador Atmel, un sensor acelerómetro MMA7341LC y una memoria SD para el almacenamiento de los datos. Las variaciones de aceleración son mostradas en una salida analógica como variaciones de tensión correspondiente al rango de fuerzas entre $-3g$ y $+3g$. El dispositivo captura estos datos de modo analógico y luego los codifica en valores discretos que varían entre 0 y 1023. Basándose en la sensibilidad del sensor y aplicando una transformación lineal, una fuerza de 1g se discretiza en aproximadamente 170 valores y corresponde a una aceleración aproximada de $9{,}8m/s^2$.

Para mantener una consistencia entre los datos de ambos dispositivos, a la placa adquisidora se le dispuso una frecuencia de lectura de los datos del acelerómetro similar a la de los acelerómetros provistos en los teléfonos, la cual es de $50Hz$. La información de ambas fuentes se procesa fuera de línea para evitar sobrecarga al sistema, que pueda conllevar a demoras en la adquisición.

Para el caso particular y dada la posición del dispositivo adquisidor, se estableció que la componente Z es la que apunta en la dirección por la que se desplaza la persona, la componente X se orienta en dirección al centro de la Tierra y la Y en la dirección ortogonal a las dos anteriores, como se observa en la Figura 3.3. Esto permite detectar fácilmente distintos tipos de movimientos como el que ocurre cuando una persona lleva su pie adelante para realizar un paso o eleva o extiende la pierna cuando sube o baja de la vereda a la calle y viceversa. Esta configuración se tomó como referencia, pero la posición del equipo puede cambiar la orientación de estos ejes sin afectar los resultados presentados.

3.3.2. Experimentación

El equipo fue portado por varias personas a las que se les asignó la tarea de recorrer un camino. En primera instancia las personas debieron llegar a la esquina disminuyendo su velocidad, detenerse antes de cruzar, mirar a ambos lados de la calle y luego cruzarla. Se nombrará a esta prueba como “prueba con detención” en las próximas referencias a ella. En una segunda oportunidad, la

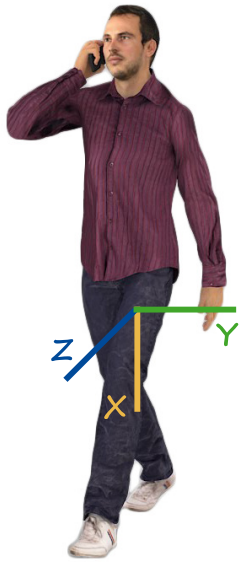


Figura 3.3: Dirección de los ejes preestablecidos. Ésta es la configuración de ejes utilizada para la adquisición de los datos.

metodología consistió en dirigirse hacia la esquina disminuyendo la velocidad sin llegar a detenerse (cuando sea posible) a la vez que se observa a ambos lados de la calle, y cruzarla. A esta prueba se la llamó "prueba con disminución de la aceleración". De esta forma se logró un entrenamiento del algoritmo que fue capaz de detectar cuando las personas se acercan a la cinta asfáltica.

En un tercer experimento, el conjunto de datos se construyó con información aportada por 3 personas, dos del sexo masculino y uno del sexo femenino, en la misma zona geográfica y en diferentes días, todos por la mañana, lo que permite evaluar el mismo tipo de tránsito vehicular y peatonal. Referiremos a ella como "prueba normal". Para esto se le pidió a cada persona que uniera diferentes puntos, marcados de forma ordenada con números sobre un mapa y que caminara como habitualmente lo hace. Las personas hicieron el recorrido de modo habitual sin prestar atención a la forma de llegar a la esquina (disminuyendo la velocidad o no, frenando o no). De esta forma, fue posible aproximar el estado de conciencia de la persona en la situación planteada.

La Figura 3.4 muestra el mapa donde se llevó a cabo la experiencia. En la misma se puede observar que el experimento comenzó en el Punto 1 y finalizó en el Punto 7, pasando por todos los puntos intermedios (2 al 6, en orden creciente). La ubicación de cada uno de estos puntos se determinó de tal forma de garantizar la

Figura 3.4: Mapa de la experimentación. En el mismo se pueden observar los hitos que debía cumplir el peatón, iniciando en el Punto 1 y finalizando en el Punto 7.

equidad de esquinas en donde el peatón debe cruzar y en las que debe doblar. Esta planificación permite que cada individuo realice el recorrido con total libertad de decisión en cada cruce, pero limitándolo a un área determinada, como se explicó en el párrafo anterior.

Además, las personas que realizaron el recorrido utilizaron auriculares con música en sus oídos. Esto permite poseer cierto grado de control sobre algunos factores, en este caso el de la distracción, y así bloquear parcialmente el sentido auditivo y forzar al peatón a prestar más atención al acercarse a la esquina.

La marca de intención antes mencionada y la acción final desarrollada fue anotada manualmente por un ayudante, que no es la misma que quien es el sujeto de experimentación, para no interferir con el objetivo dado. Para apuntar la marca de intención, el ayudante camina alrededor de 30 metros por detrás de la persona evaluada y registra el momento cuando esta última se encuentra al menos a 3 metros de distancia de la cinta asfáltica. Luego, se toma nota de la

acción que se desarrolló.

3.3.3. Método de validación

Se utilizó validación cruzada con el método de *k-fold*, el cual particiona el conjunto de datos en k divisiones independientes. En este caso se particionaron $k = 10$ conjuntos de datos separados. El primer paso del algoritmo fue realizar el entrenamiento usando $k - 1$ particiones y evaluar el rendimiento en la partición restante. El proceso se repite k veces hasta que todas las particiones han sido usadas como conjunto de evaluación. Por último se calculó el promedio del error de evaluación sobre todas las particiones. Este método de validación provee una buena estimación de la precisión utilizando todo el conjunto de datos. Aunque requiere múltiples ajustes, hace un uso eficiente de toda la información, lo cual es beneficioso para pequeñas cantidades de datos, como en este caso. Este sistema de validación fue el usado para evaluar el rendimiento de todos los clasificadores que se utilizaron en la obtención de los resultados.

3.4. Análisis de aceleraciones antes de un cruce

Se reconoció a simple vista la existencia de patrones en los datos del acelerómetro que comprenden la primer etapa de la experimentación. Los patrones son visualmente legibles debido a la disminución de su aceleración al llegar a la esquina y al cambio en la frecuencia de sus pasos.

En la Figura 3.5 se muestra una secuencia completa de las aceleraciones obtenidas de una de las personas evaluadas. La señal azul son los datos de aceleración crudos de la persona caminando, obtenidos desde el acelerómetro, mientras que las franjas de color celeste indican el lapso de tiempo durante el cual la persona caminó por la cinta asfáltica. Cabe destacar que el instante $t = 0s$ es el comienzo de la experiencia y que la aceleración se muestra en unidades gravitacionales g, donde $1g$ equivale a $9,8m/s^2$ aproximadamente. A los ejes afectados por la gravedad se les eliminó la componente de continua para una mejor visualización de las alteraciones en la señal. A partir de estos datos se puede observar que existen alteraciones en cercanías a las franjas celestes.

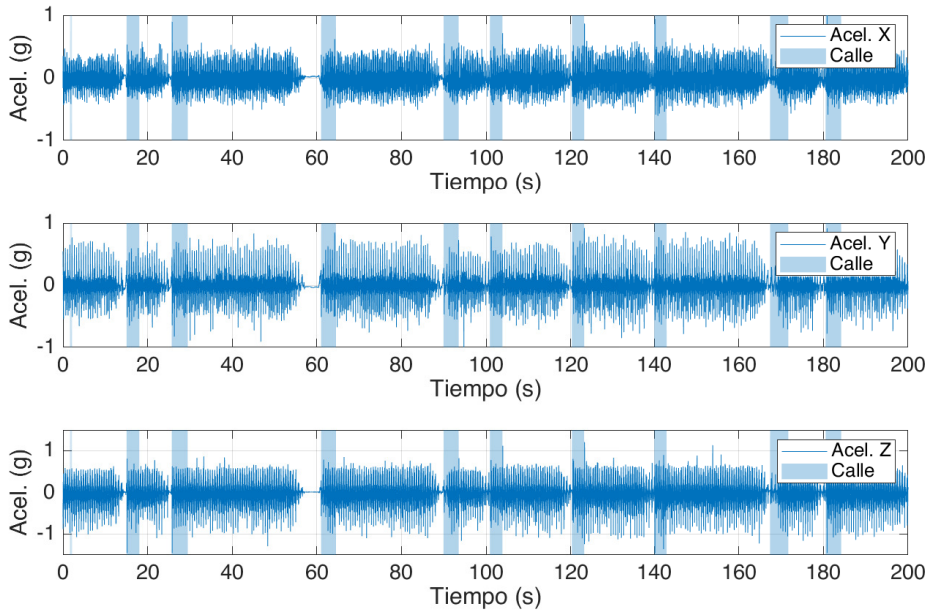


Figura 3.5: Gráfico general de los datos de los ejes X, Y y Z. En el mismo se pueden ver alteraciones de la señal y el área perteneciente a la calle (sombreada).

En primera instancia se analizó la detección de la frecuencia de los pasos con el fin de obtener una variación de la misma cuando la persona se acerca a la calle. La técnica de detección se llevó a cabo por medio del uso de la FFT[3]. Para la implementación se aplicó la FFT a una ventana temporal en la secuencia de datos. Se analizaron diferentes tamaños de ventanas temporales. La que mejor resultados entregó, para este caso, fue una de 2 segundos de duración. Este es el tiempo óptimo en que la FFT puede calcularse con bajo error. Luego, la ventana es desplazada 0,5 segundos y vuelve a realizarse el procedimiento de cálculo. De este modo se pueden obtener las frecuencias fundamentales cada 0,5 segundos. Posteriormente, mediante la comparación de la frecuencia principal en ventanas consecutivas, se esperó observar el corrimiento de la frecuencia cuando la persona se aproximaba a la calle, debido a la disminución en la frecuencia de los pasos. Si bien los datos obtenidos son de buena calidad, la FFT no produjo un espectro limpio donde se destaque una frecuencia principal que se pueda comparar con las demás, por lo que fue difícil observar su corrimiento respecto de los espectros obtenidos en las otras ventanas.

[3]Transformada Rápida de Fourier -*Fast Fourier Transform*-

Este procedimiento se descartó y se optó por la detección de las envolventes y consecuente detección de la disminución de la magnitudes de las aceleraciones. El algoritmo desarrollado para este caso consiste en la detección de las envolventes superior e inferior de las aceleraciones. Entre los datos de los tres ejes, se decidió el uso de los del eje Z debido a que, para el objetivo propuesto, son los que mejor se pueden interpretar y tienen un patrón repetitivo más marcado que el resto. Una comparación que soporta esta elección se puede observar en la Figura 3.6 donde se observa, alrededor de los 88 segundos, una atenuación de las magnitudes de las aceleraciones medidas y donde se muestra sombreado el momento en el que se comienza a cruzar la calle.

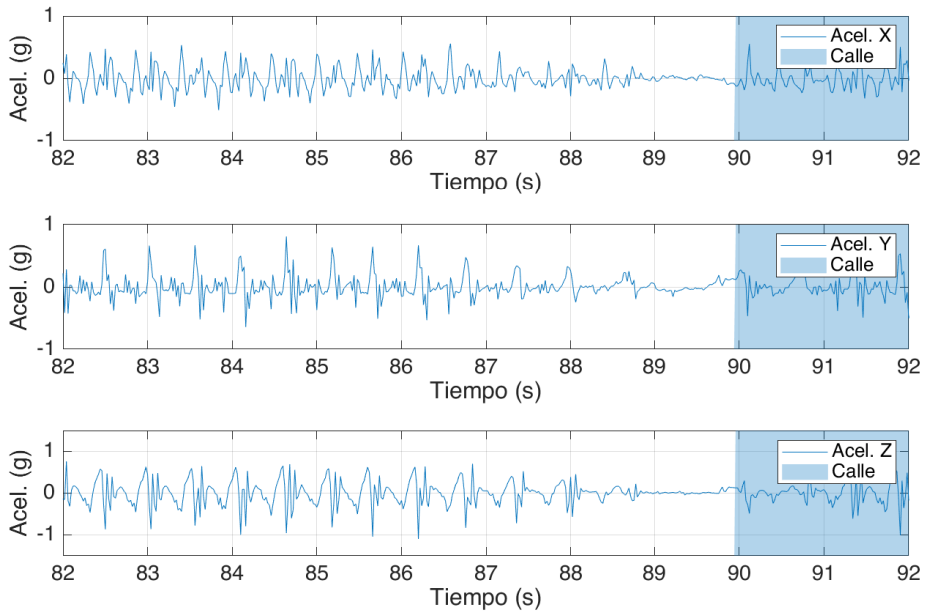


Figura 3.6: Comparación de los datos crudos de los 3 ejes. El sombreado indica el área correspondiente a la calle.

Las alteraciones son detectadas obteniendo la envolvente superior y inferior de la señal. De esta forma se puede estimar la señal interna y eliminar gran cantidad de información que no resulta relevante. La Figura 3.7 muestra una ampliación en tiempo de la señal del eje Z de la Figura 3.5, que es la misma que la Figura 3.6 pero con la envolvente superior e inferior aplicadas. Este procedimiento, además de ser necesario para el enfoque propuesto en este capítulo, arroja como resultado una señal de la que se puede obtener una detección en tiempo de los pasos del

peatón. Este tipo de procesamiento es utilizado generalmente por aplicaciones deportivas (especialmente en teléfonos) para el cálculo de otros parámetros como distancia recorrida, ritmo, etc.

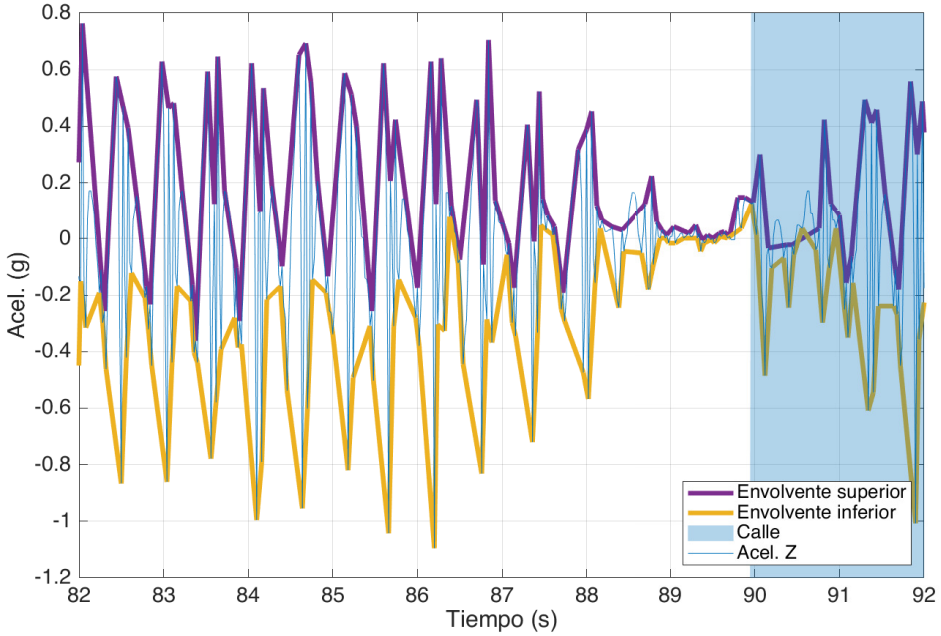


Figura 3.7: Envolventes superior e inferior de los datos del eje Z. En línea gruesa se observa la envolvente superior (violeta) e inferior (naranja). También se muestran los datos crudos en color claro y el área correspondiente a la calle en sombreado.

Posteriormente, se calcula una tercer envolvente teniendo en cuenta sólo los puntos máximos de la envolvente superior, a la que se llamó "envolvente de máximos". Esta envolvente es el resultado del desplazamiento de una ventana temporal, diferente de la mencionada anteriormente, de longitud fija, que calcula el valor máximo sobre la envolvente superior de la señal. La Figura 3.8 muestra la envolvente de máximos junto con la envolvente superior y la señal original. Los puntos claros sobre la envolvente máxima son los puntos máximos de la envolvente superior, dentro de la ventana temporal, resultantes del procedimiento explicado.

Con el objetivo de detectar la intención cuando un peatón se acerca a la calle, otra parte del algoritmo determina la disminución de la aceleración sobre la envolvente de máximos. Este procedimiento permite confirmar que el peatón disminuye la aceleración en el momento previo a cruzar la calle. La Figura 3.9 muestra esta detección en línea más gruesa, en momentos previos a la zona som-

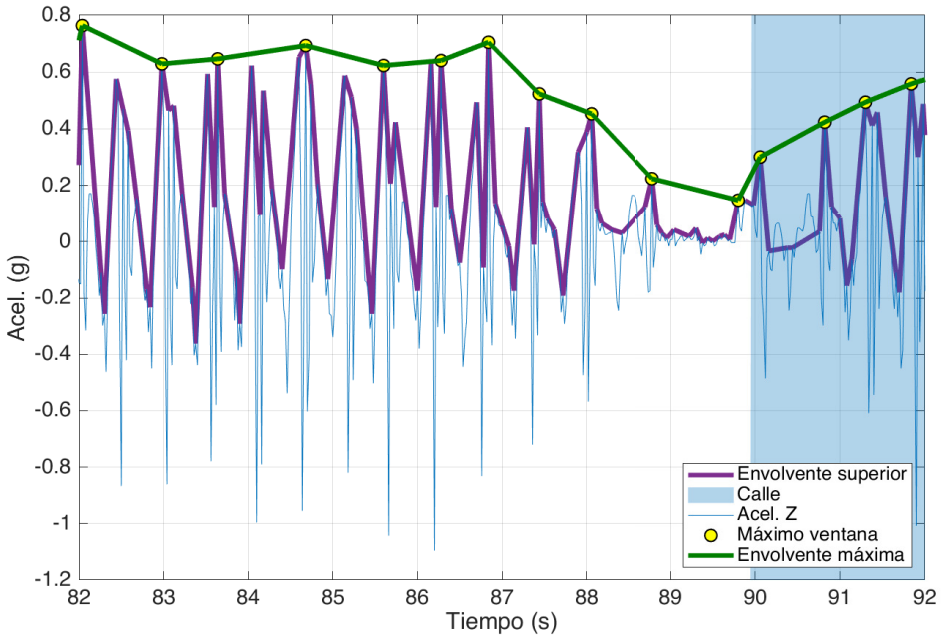


Figura 3.8: Envolvente superior y envolvente máxima, producto del algoritmo. En línea gruesa se observa la envolvente superior (violeta) y la envolvente de máximos (verde). También se muestran los datos crudos en color claro y el área correspondiente a la calle en sombreado. Los puntos amarillos indican los máximos de la envolvente superior en cada ventana de tiempo.

breada, que es la que indica el área correspondiente a la calle.

La disminución de la aceleración es el resultado de la atención que la persona está prestando a su entorno. Cuando el peatón se aproxima a una esquina, toma recaudos tales como observar a ambos lados de la calle, a su vez que reduce su aceleración. Se observa en la misma figura, que en algunas ocasiones, la persona ha detenido su marcha, como en el instante $t = 60seg$ de la Figura 3.10. En otras, en cambio, la disminución de la misma sufre pocos cambios, pero es suficiente para detectar un comportamiento diferente al que ocurre mientras la persona transita alejada de un cruce. Cuando no es posible detectar la disminución o alteración en el patrón de aceleraciones estándar que lleva a cabo la persona durante el camino, se comprende que la misma no se encuentra totalmente con la atención enfocada en su entorno. Esta actitud no es un riesgo si la persona se encuentra caminando lejos de una esquina o del borde de la calle, pero si lo presenta en el caso de estar cerca de una zona de cruce.

Para reforzar los resultados, la Tabla 3.1 muestra los porcentajes de falsos

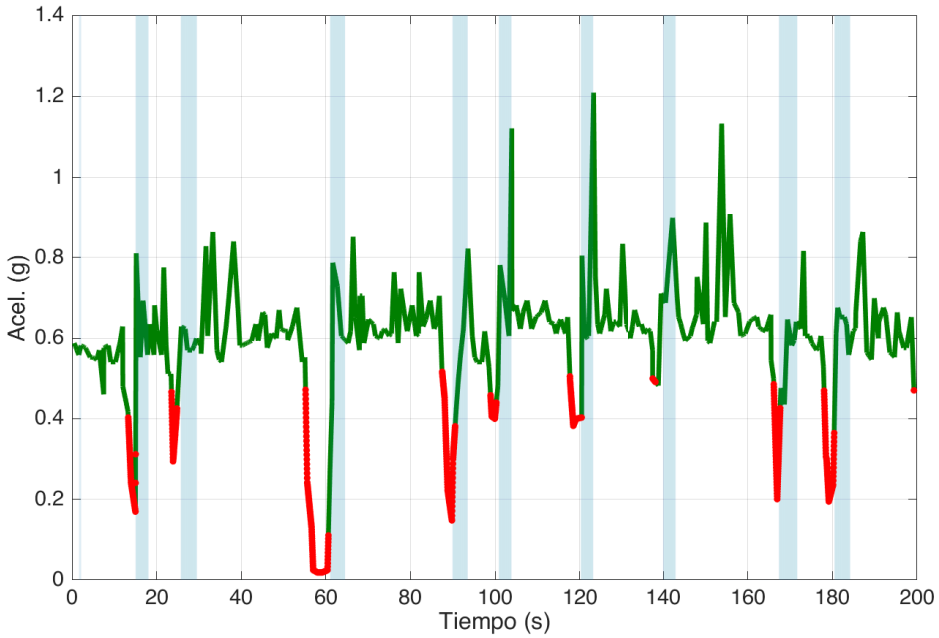


Figura 3.9: Detección temprana de cercanía a la cinta asfáltica. En línea fina (verde) se muestra la envolvente de máximos y en línea gruesa (rojo) la detección temprana. Las zonas sombreadas indican el área correspondiente a la calle.

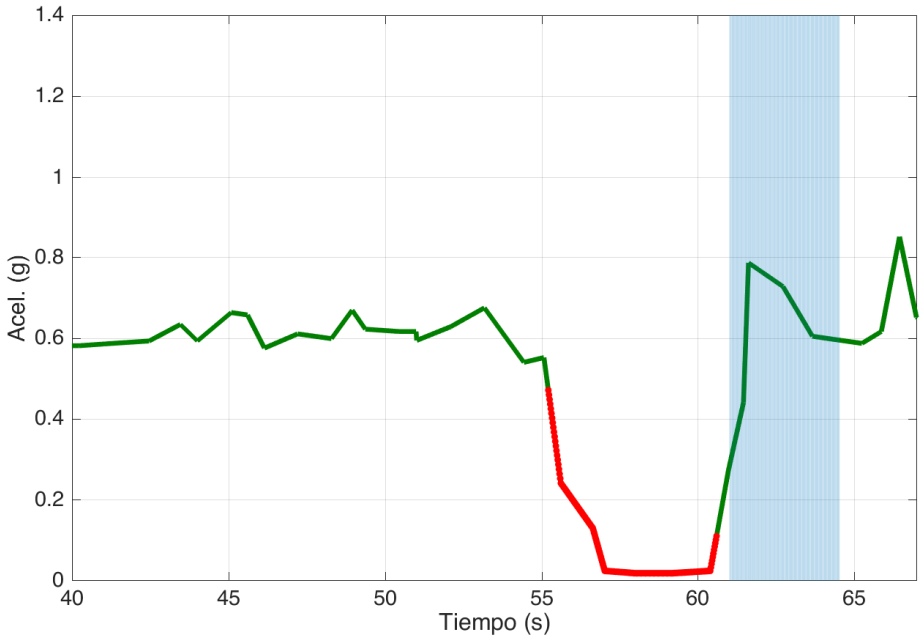


Figura 3.10: Aumento de tamaño de la Figura 3.9 en $t = 60s$. En línea fina (verde) se muestra la envolvente de máximos y en línea gruesa (rojo) la detección temprana. La zona sombreada indica el área correspondiente a la calle.

positivos y negativos obtenidos en los tres distintos tipos de pruebas descriptas en la Sección 3.3.2.

Tabla 3.1: Porcentajes de detecciones de los distintos tipos de experimentos.

	Falso Positivo	Falso Negativo
Con detención	40 %	0 %
Con disminución	47 %	0 %
Normal	48 %	19 %

Se observa que en función del aumento de la complejidad de las pruebas (comenzando por *Con detención* y terminando por *Normal*) el porcentaje de ambas columnas también se incrementa. Cabe destacar que los falsos negativos (cuando el peatón se acerca a la calle pero su comportamiento no es detectado) siempre es inferior a los falsos positivos (cuando se detecta un comportamiento similar al de cruzar) pero no en inmediaciones de la calle. Esto da la pauta de que el porcentaje de fallas del algoritmo es en beneficio de la seguridad del peatón, detectando mayor cantidad de veces un comportamiento preventivo que la no detección del mismo cuando realmente ocurre.

3.5. Cálculo de la intención de cruce

Para hacer una evaluación de la intención utilizando aprendizaje automático, la información provista por el acelerómetro no se utiliza directamente como entrada en los clasificadores. Para ello se calcula la RMS[4], los coeficientes de correlación de la señal, los máximos, los mínimos, la media y el desvío estándar, como métricas de los tres ejes de la aceleración, las que suman un total de 21 características de la señal. La elección de estas métricas se debe a que son las más adecuadas y utilizadas en la bibliografía [60] [65] para este tipo de usos de la señal.

El cálculo de cada una se realiza sobre una ventana temporal de una longitud de 1 segundo, que se desplaza sobre los datos originales con una superposición de 50 % con la ventana anterior, brindando métricas cada 0,5 segundos.

El tiempo de la ventana asegura que se consideren al menos dos pasos del peatón, uno con la pierna izquierda y uno con la derecha, o viceversa, que permite evaluar la dinámica del peatón. Esta información se almacena en forma de matriz, donde las primeras columnas corresponden a las métricas y la última al tipo de

[4]Raíz Media Cuadrática -*Root Mean Square*-

acción que realizará. En tanto, cada fila es una muestra de cada ventana temporal.

Por tanto, el conjunto de datos utilizado para el entrenamiento de las redes se compone de las métricas, que se utilizan como predictores en las entradas de los clasificadores, junto a la etiqueta de la clase correspondiente, como señal de respuesta en la salida, obtenida a partir de la acción posterior. El total de datos posee 68 % de subconjuntos etiquetados como *cruza* y 32 % de datos etiquetados como *dobla*. Nótese que los datos de ambas clases no se encuentran balanceados. Si bien el diagrama del recorrido del experimento se realizó para acercar lo más posible la cantidad de datos entre ambas clases, lograr una equidad entre ambas clases supone diagramar un recorrido complejo que no le permite un comportamiento natural a los individuos, concepto que se tomó como objetivo inicial de este experimento.

Para la obtención de resultados se utilizó el módulo *Classification Learner* de *MATLAB* ®, en el que se evaluaron los clasificadores de uso habitual en la bibliografía ya mencionada.

Como se menciona en [66], una buena solución se alcanza con un conjunto de clasificadores basados en redes neuronales y es en base a la cuál se desarrollaron los resultados presentados. Principalmente se utilizó el algoritmo de subespacios aleatorios (*Random Subspace*) con clasificadores k-NN. Este algoritmo combina modelos producidos por diferentes clasificadores en un conjunto que mejora el rendimiento de los clasificadores originales. Para ello toma aleatoriamente un subconjunto de características de los datos y luego aplica el algoritmo de entrenamiento, lo que significa que cuando la muestra de prueba es comparada con el modelo, sólo las características seleccionadas contribuyen a la distancia. Geométricamente, esto es equivalente a la proyección de todos los puntos en el subespacio seleccionado y los vecinos cercanos son encontrados usando las distancias proyectadas. En cada iteración, un subespacio es seleccionado aleatoriamente y un nuevo conjunto de k-NN es computado reduciendo la correlación entre los estimadores. La salida se produce como resultado de la votación por mayoría de la combinación de las salidas individuales de cada modelo.

A fin de conseguir el rendimiento más alto de este clasificador, fue necesario redefinir la ventana de tiempo sobre la que se calculan las métricas antes men-

cionadas. Para ello se determinó que el lapso de tiempo previo para la evaluación de la intención (marcado en la Figura 3.1 y 3.2 con una línea roja continua) que produce el mejor rendimiento del clasificador es de 3,5 segundos. Este tiempo es el resultado de la evaluación de diferentes tiempos, como se muestra en la Figura 3.11, tomando aquel que mejor precisión arroja en la estimación de la intención. Cada punto graficado es el promedio de cuatro entrenamientos con la misma ventana de tiempo, ya que el resultado del clasificador puede variar en cada iteración.

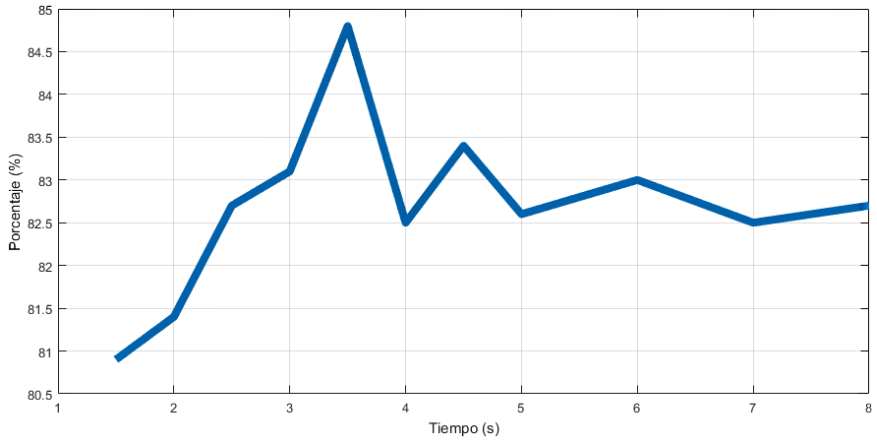


Figura 3.11: Gráfica de las mejores precisiones obtenidas evaluando el clasificador con diferentes lapsos de tiempo en los datos.

De esta forma, con este clasificador se consigue como resultado una precisión del 84.8 %.

Otros tipos de clasificadores fueron entrenados a fin de comparar la precisión de los resultados utilizando los mismos datos de entrada. En la Tabla 3.2 se muestran las precisiones resultantes de k-NN pesado, SVM gausiano y cúbico, y un árbol de decisión simple. La comparación se basa en la tasa de error, que es una medida rápida y directa. Aunque ésta no sea lo suficientemente informativa, ayuda a reconocer que es posible utilizar este clasificador para brindar información importante que, complementariamente, ayude a reconocer la intención de un peatón.

En la matriz de confusión de la Figura 3.12 se observa el total de los datos utilizados para el entrenamiento y las coincidencias y errores entre las predicciones de ambas clases.

De forma más detallada, en la Figura 3.13 se observan los porcentajes de

Tabla 3.2: Tabla de precisiones resultantes de los distintos clasificadores entrenados, ordenados de mayor a menor.

Tipo de clasificador	Precisión
Subespacio k-NN	84.8 %
k-NN Pesado	73.7 %
SVM Gausiano	72.2 %
SVM Cúbico	71.7 %
Arbol Simple	70.7 %

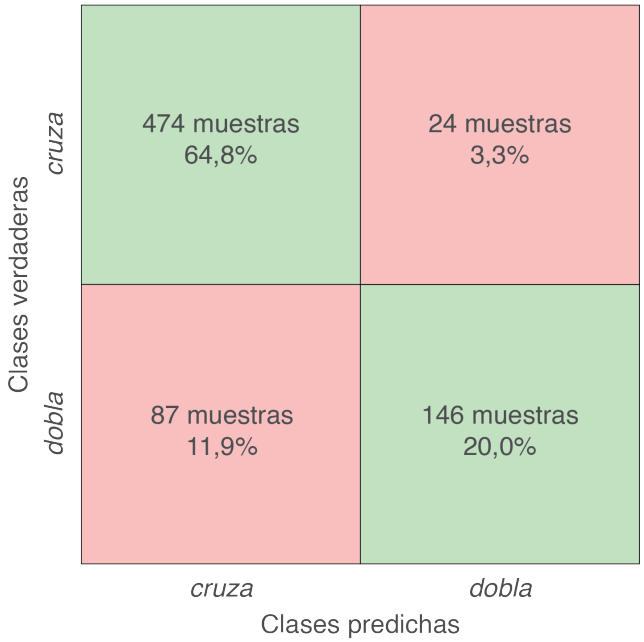


Figura 3.12: Matriz de confusión de la totalidad de los datos utilizados.

predicción correcta en las clases verdaderas. Así, se desprende que los verdaderos positivos (aquellas intenciones que fueron bien predichas por el clasificador) superan ampliamente a los falsos negativos (aquellos casos que la predicción resultó errónea con respecto a la clase original). En el caso de la clase *cruza*, el 95 % de las predicciones fueron correctas, dejando de lado sólo un 5 % de incorrectas. Para la clase *dobla*, el porcentaje es inferior, acertando en un 63 % las ocasiones en las que el peatón dobló y la red acertó en la predicción.

Del lado de las predicciones de las clases, en la Figura 3.14 se muestra que se ha predicho satisfactoriamente el 84 % de la clase *cruza* y 86 % de la clase *dobla*. Estos valores demuestran que el clasificador tiene un alto porcentaje de acierto al momento de realizar las predicciones.

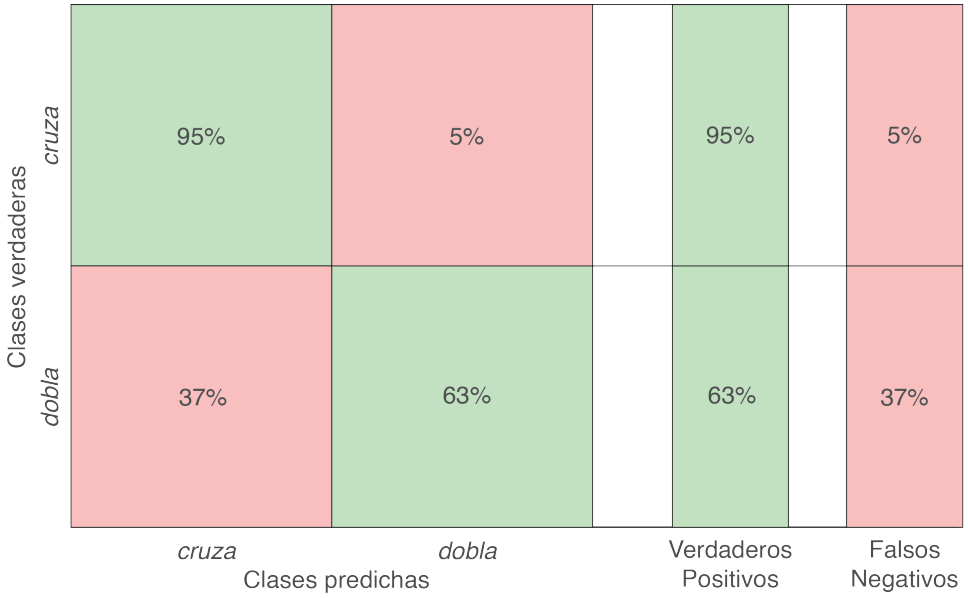


Figura 3.13: Tasa de verdaderos positivos y falsos negativos.

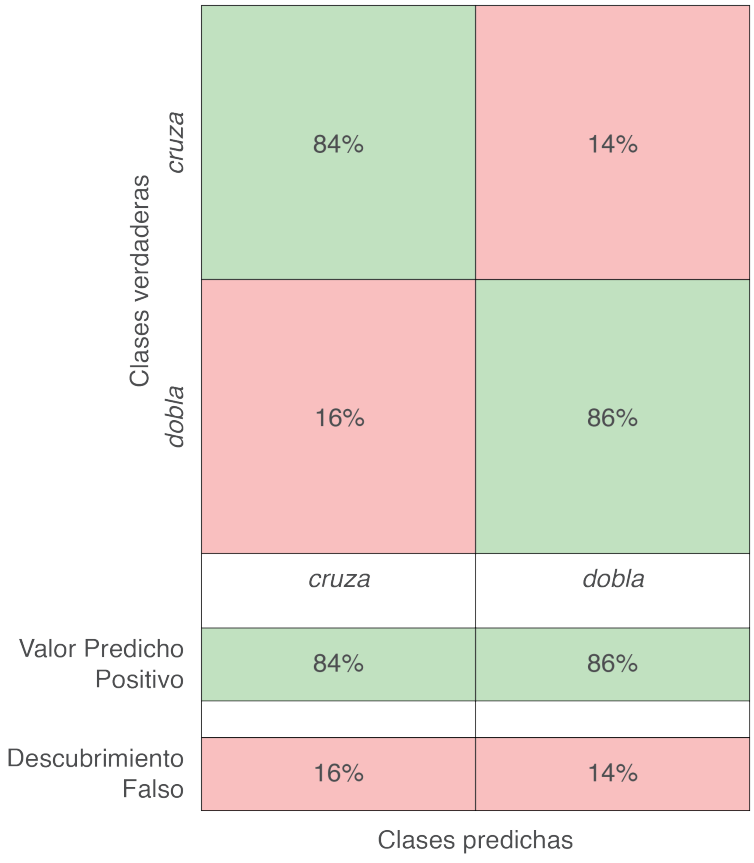


Figura 3.14: Valores predichos positivos y tasa de descubrimiento de falsos.

3.6. Conclusión del capítulo

Se puede afirmar que el peatón está prestando atención al momento de cruzar una calle si su aceleración disminuye. Esto significa que la persona está consciente

de la situación a llevar a cabo y, debido a las precauciones que toma al acercarse al borde de la vereda y a la carga cognitiva que demanda realizar una acción de este tipo, se establece que el nivel de atención en ese momento es alto, por lo que no presentaría un riesgo para la misma. En caso de que no sea detectada una disminución de la aceleración mientras la persona se acerca a la calle, puede decirse que la persona no tiene el nivel adecuado de atención puesto en el entorno en el que se desarrolla la situación. Esto podría derivar en un cruce peligroso, donde el peatón no es consciente de los vehículos en circulación. De este último párrafo se desprende otra cuestión, en la que el peatón no tenga intención de cruzar la calle. Un ejemplo de esto es el caso en el cual la persona doble en la esquina y continúe por la vereda sin suponer riesgo alguno, por lo que es posible que en ningún momento su aceleración disminuya.

Por ello es que resulta de interés evaluar la intención del peatón antes que realice la acción prevista. Para esto se evaluaron una serie de clasificadores de distinto tipo utilizando características de las señales de aceleración para entrenarlos, consiguiendo una precisión en la detección de la intención del 84,4 %.

En lo que respecta al análisis de los resultados, en la Figura 3.13, se puede observar que el porcentaje de aciertos de la clase *cruza* es superior al de la clase *dobla*. Considerando que es más importante detectar la intención de cruzar que la de doblar (ya que deriva en condiciones de seguridad), los resultados demuestran que parte de los errores ocurren con más frecuencia en el área donde la seguridad del peatón no se encuentra afectada por la proximidad de un vehículo. En el caso opuesto, cuando el peatón tiene intención de cruzar la calle, la probabilidad de predecir erróneamente esta situación es de alrededor del 5 %. Cabe agregar que estos resultados se encuentran ligados al desbalanceo de datos entre ambas clases, lo que puede ser una fuente de error de la clase menos presente, y que se debe considerar al momento de extender la utilización de este método.

Si bien, en algunos trabajos como [67], la precisión para estimar la intención es superior al resultado obtenido en esta experimentación, se demuestra que es posible usar un clasificador como el mencionado para obtener resultados que ayuden a la detección de intenciones. Además, es importante considerar que sólo se utilizan los datos provistos por el acelerómetro portado por el peatón. Esto es

provechoso ya que no depende de la infraestructura del lugar ni de condiciones externas de visibilidad.

Además, los resultados de la predicción están disponibles inmediatamente luego del tiempo propuesto para medir la intención (3,5 segundos para este caso). Esto significa que, en referencia a la Figura 3.2, la marca de intención se encuentra en un tiempo y posición anterior a que el peatón pise la calle (para el caso de que desee cruzar). De este modo se asegura que el vehículo, sin considerar las demoras en transmisión de la información, podría obtener la intención del peatón con antelación, y así llevar a cabo las acciones necesarias para proveer al peatón y al conductor la seguridad necesaria para evitar el accidente.

Capítulo 4

Medición indirecta de la dinámica del peatón

El seguimiento de las extremidades de un peatón es, probablemente, la información más relevante usada para inferir la intención. Cada peatón puede realizar una entre una variedad de acciones en un entorno urbano, las más comunes son caminar y estar de pie. Estos dos estados, combinados con otra información contextual y secuencial, podrían ser usados para obtener un modelo predictivo de la intención del peatón.

Algunos autores han usado cámaras y lásers para detectar los estados del peatón [68] [69]. Otros, por ejemplo [70], [71] y [64], usan sensores como acelerómetros, giróscopos, magnetómetros y GPS que son localizados en los peatones. Estos dispositivos portátiles demostraron que pueden proveer información dinámica exhaustiva de los movimientos del peatón. Con el ajuste y ubicación apropiados de los sensores, se puede medir la aceleración y velocidad de diferentes extremidades. Distribuyendo sensores por sobre el cuerpo del peatón se obtiene una descripción dinámica completa de los movimientos del mismo. Los dispositivos portátiles actuales son usados para reconocer actividades como caminar, correr y estar de pie, entre otras. La información que ellos proveen es precisa, pero no tienen la capacidad de transmitirla a los vehículos de la proximidades, por lo que esta información sólo está disponible para los peatones que portan los dispositivos.

Estas limitaciones han impedido el uso de este tipo de información en ADAS[1] y aplicaciones de vehículos autónomos. La pregunta fundamental es si es posible obtener información dinámica y cinemática, precisa y de confianza de las extremidades de los peatones usando sensores de visión que están presentes en los vehículos inteligentes.

Estos últimos tienen distintos tipos de sensores para percibir sus entornos próximos, los más comunes son las cámaras. Este capítulo demuestra que desde la información de las cámaras es posible obtener la dinámica de los peatones con una precisión similar a la de los dispositivos portátiles antes mencionados. Esto se realiza comparando la rotación y aceleración obtenida de los dispositivos portátiles, instalados en las muñecas de los peatones, con información similar obtenida por visión. La información dinámica, a partir de la la visión, es obtenida usando métodos robustos que combinan la representación por esqueletos virtuales con información semántica. Los resultados experimentales presentados demuestran la fuerte correlación entre las medidas de los dispositivos portátiles y las observaciones visuales de la información de velocidad y aceleración. Así, este capítulo presenta una solución de sensado para resolver el problema fundamental que es la seguridad de los peatones a partir de la inferencia de la intención.

4.1. Obtención de la representación por esqueletos e información derivada

4.1.1. Extracción del esqueleto virtual

Obtener la postura precisa de un peatón desde una imagen es un verdadero desafío. En [72] se propone un método para estimar el esqueleto subyacente del peatón capturado por una cámara de profundidad y se lo hace corresponder con los datos de una base de datos. Esta base de datos consiste en nubes de puntos de modelos humanos de alta resolución en diferentes posturas.

En [73] se propone otro método que explora la antropometría general de un sujeto humano detenido, fotografiado por una cámara estéreo. Una búsqueda

[1]Sistemas Avanzados de Asistencia al Conductor -*Advanced Driver Assistance Systems*-

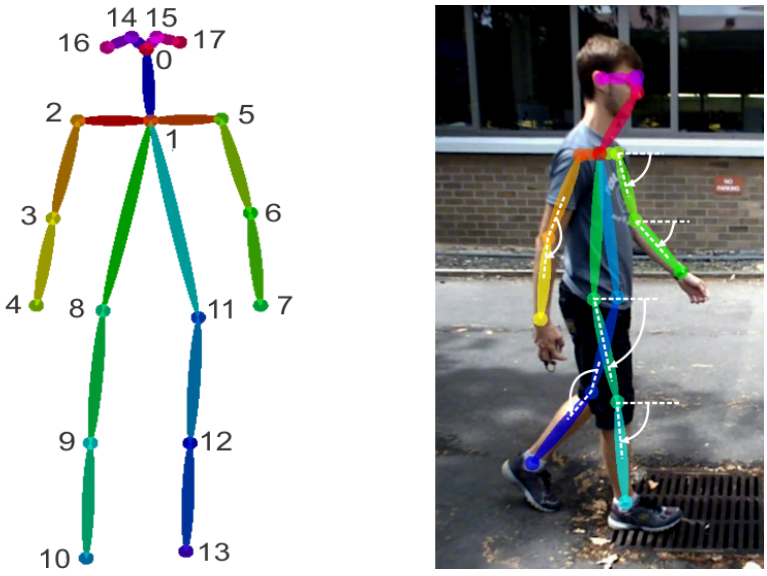
jerárquica de arriba hacia abajo (*top-down* en inglés) de las articulaciones del peatón, basadas en medidas antropométricas, es realizado de acuerdo a la estimación del esqueleto del peatón. Luego, un algoritmo de puntuación y muestreo permite la determinación de los segmentos y articulaciones del esqueleto.

Una aproximación novedosa para la estimación y seguimientos monoculares de posturas en 3D, en condiciones reales de calle, es presentado en [74]. La solución explora métodos de seguimiento en un número de cuadros consecutivos de una cámara monocular, y los correlaciona con posibles soluciones de posturas previamente almacenadas.

Por último, un reciente y exitoso algoritmo de código abierto, denominado *OpenPose*, desarrollado por [75] [76] [77], permite obtener la representación del esqueleto de un peatón en tiempo real. También se ha demostrado que este algoritmo funciona de manera robusta incluso con una multitud de peatones en una sola imagen, lo cual es un requisito esencial para el seguimiento de personas en un entorno urbano. Por esta razón, dicha biblioteca se usará como herramienta para la obtención de los esqueletos de los peatones en esta tesis.

El esqueleto de cada persona en la imagen es representado usando 18 puntos claves que incluyen 5 puntos en la cara, 1 en el pecho, 3 por cada brazo (hombro, codo y muñeca) y 3 por cada pierna (cadera, rodilla y tobillo) que son extraídos usando modelos particulares previamente entrenados por los autores del algoritmo. La Figura 4.1a muestra la representación de un cuerpo dados los puntos mencionados y los segmentos conformados entre ellos, mientras que la Figura 4.1b lo hace sobre una imagen real.

Cada uno de los puntos, coincidente en su mayoría con articulaciones, son representados en la imagen como un par de coordenada X e Y. Estas coordenadas se utilizarán para calcular los ángulos de las extremidades, la velocidad angular y la aceleración lineal en cuadros consecutivos de video, como se describirá en la Sección 4.1.3. Además, el algoritmo provee un tercer valor referido a la confiabilidad del punto mostrado, utilizado algunas veces para descartar puntos falsos.



(a) (b)

Figura 4.1: Esqueleto obtenido usando *OpenPose*. (a) muestra los puntos del esqueleto. (b) ilustra el esqueleto y los ángulos correspondientes.

4.1.2. Corrección de fallas en detecciones del esqueleto

La representación del esqueleto de los peatones es obtenida usando la biblioteca *OpenPose* como se mencionó anteriormente. A pesar de su precisión, existen ciertas condiciones en las cuales los resultados son incompletos o el algoritmo arroja falsas detecciones. Una falla común aparece cuando un brazo está oculto detrás del cuerpo o una pierna detrás de la otra. En algunos casos la extremidad no es mostrada, pero en otros éstas aparecen en un lugar incorrecto. Para evitar este problema, se optó por seleccionar el brazo o pierna más visible del sujeto de interés como extremidad principal para seguir su trayectoria.

También es frecuente la alteración de la lateralidad de los miembros, tanto inferiores como superiores. Esto significa, por ejemplo, que el algoritmo confunde o altera la pierna derecha con la izquierda, y lo hace de un cuadro de video a otro, y no a lo largo de toda la secuencia de video. Lo mismo ocurre para los brazos. Este cambio en la representación de las extremidades afecta directamente los resultados de los cálculos de la dinámica de esos segmentos. Para solucionar este inconveniente, se desarrolló un algoritmo que pre-procesa los esqueletos resultantes de las imágenes y hace un seguimiento de cada extremidad, corrigiendo aquellos casos en que las extremidades derechas estén representadas en las izquierdas, y viceversa.

Otro problema que puede manifestarse es que el algoritmo produce falsos positivos detectando esqueletos erróneamente en lugares donde no se encuentran personas. Esto trae aparejado el inconveniente que, ante la presencia de dos o más personas en la escena, el orden de la información de los esqueletos puede verse afectado, haciendo que la información de un esqueleto que se está evaluando, pase a ser la del otro esqueleto, ocasionando graves fallas en los algoritmos desarrollados. En el conjunto de datos recolectados para este experimento, a menudo había sombras o formas similares a humanos, que no correspondían a tales. Este problema fue resuelto usando información semántica en las imágenes. Algunos algoritmos actuales pueden operar en tiempo real y clasificar cada píxel de la imagen como perteneciente a una clase particular, incluyendo peatones. En [78] se presentan resultados experimentales adaptando un modelo existente a un nuevo entorno, re-entrenando una red neuronal con su propio conjunto de datos, describiendo así nuevas características del entorno local. Este algoritmo fue usado para enmascarar sólo las áreas de la imagen etiquetadas como "peatones", lo cual ayudó en el problema de las falsas detecciones. Una secuencia comparativa se muestra en la Figura 4.2 y en el video disponible en [79]. La Figura 4.2a muestra una imagen estándar tomada con la cámara frontal del vehículo. Todas las imágenes fueron rectificadas usando los parámetros intrínsecos de la cámara. La imagen en la Figura 4.2b fue procesada para extraer el esqueleto de la persona que está cruzando la calle. En esta imagen se muestra un resultado erróneo, donde un esqueleto extra fue detectado en el sector derecho de la imagen. Este esqueleto, marcado con un ovalo punteado amarillo, no está relacionado al peatón en la escena ni a ningún otro. La Figura 4.2c muestra el resultado de la clasificación semántica de la misma imagen, la cual es usada para filtrar el esqueleto fantasma detectado.

Por último, cabe destacar que la conformación del esqueleto de una persona tiene limitaciones relacionadas con el tamaño de esa persona en la imagen. Esto se ve claramente en los casos en los que los peatones se encuentran demasiado lejos del vehículo y no es posible conformar un esqueleto sobre ellos. De las pruebas realizadas, se puede concluir que el rango de detección correcta para las cámaras del vehículo es de entre 1 y 8 metros, medidos a partir de la cámara

(a) (b) (c)

Figura 4.2: Secuencia del uso de información semántica para eliminar falsas detecciones. La imagen original capturada por la cámara frontal del vehículo se muestra en (a). (b) ilustra la detección del falso esqueleto en el sector derecho de la imagen. En (c), la información semántica es usada para filtrar la falsa detección. Los peatones son coloreados en amarillo, los vehículos en rojo, las áreas de manejo en marrón, la vegetación en verde y el cielo en azul, entre otros.

que toma la imagen. La medición es factible utilizando otros sensores montados en los vehículos, como lidares. Con esta medición es posible saber, con certeza, que existen más peatones de los que puede reconocer el algoritmo. Sin embargo este tipo de detecciones no forma parte de este trabajo y se considera que la distancia donde el esqueleto puede ser conformado es suficiente como para evitar situaciones de riesgo.

4.1.3. Cálculo de la información dinámica

Como los puntos del esqueleto son provistos cuadro a cuadro en el video, obteniendo las imágenes a una frecuencia apropiada es posible evaluar la información dinámica de los principales segmentos del esqueleto, principalmente las extremidades, que son el punto de interés en este capítulo.

El ángulo de cada extremidad o articulación se obtiene considerando los puntos extremos de cada segmento y computando su ángulo respecto a la línea horizontal por medio de funciones trigonométricas, como se muestra en la Figura 4.1b. También se calculan otros ángulos entre diferentes segmentos del esqueleto, como el caso del codo, que resulta del ángulo formado entre el segmento hombro-codo y codo-muñeca.

Las velocidades angulares son la primer derivada de los ángulos mencionados, obtenidas considerando el cambio de ángulos en dos imágenes consecutivas y el intervalo de tiempo entre ellas. Para el caso de la velocidad angular de una extremidad articulada, como el brazo o la pierna, con un punto pivot en el centro, la

velocidad se calcula en base al segmento que conforman los puntos más distantes de la extremidad, sin considerar el central. Esta metodología se utilizó considerando que la complejidad del cálculo incluyendo el punto intermedio no justifica el error producido, ya que en general el ruido provocado por la variabilidad de los puntos es mayor al error de calculo comentado. La variabilidad del punto en cada extremidad es un pequeño corrimiento de su localización en la imagen, producido por la biblioteca *OpenPose* ante la dificultad de contar con imágenes de mejor calidad que la ayuden a mantener con mayor precisión la ubicación del punto en el lugar indicado. Debido a ello, la señal resultante suele ser un poco ruidosa por lo que es suavizada usando un filtro promediador móvil.

Por último, las aceleraciones lineales son la segunda derivada del desplazamiento lineal de los puntos del esqueleto en tres imágenes consecutivas. A esta señal también se le aplica el filtro ya mencionado.

Con esta metodología es posible obtener la información dinámica necesaria para el modelado del movimiento peatonal. Así mismo se demostrará que la calidad de la información obtenida es comparable a la de los giróscopos y acelerómetros típicos instalados en los dispositivos portátiles.

4.2. Evaluación de la dinámica del cuerpo y sus partes

En esta sección se detalla la metodología propuesta, validando los resultados y mostrando el alcance logrado. Los resultados experimentales se presentan para demostrar la posibilidad de obtener información dinámica muy completa de las extremidades que representan el esqueleto de un peatón.

4.2.1. Experimentación

Con el objeto de demostrar que la calidad de los resultados inerciales obtenidos utilizando sensores visuales es comparable con la información brindada por los sensores inerciales, se diseñaron dos experimentos que son el marco soporte para la validación de la información dinámica.

Ambos experimentos fueron realizados para obtener datos de visión e información inercial, involucrando peatones y vehículos con cámaras. El primero considera una persona detenida y moviendo sus brazos, como simulando el movimiento de caminar. En la mano y pie más visibles por la cámara, la persona porta giróscopos y acelerómetros que recolectan los datos del movimiento y son los que permitirán realizar la comparación propuesta. El segundo experimento considera a peatones caminando naturalmente, sin restricciones de movimiento, y también portando los dispositivos en sus manos. Todos los peatones realizaron las acciones frente a la cámara del vehículo y el experimento fue repetido independientemente para cada uno.

Para la recolección de los datos se usó un vehículo eléctrico con un conjunto de sensores montados para facilitar la conducción autónoma, incluyendo seis cámaras de gran angular y una computadora automotriz NVIDIA DRIVE PX2, la cual incluye dos GPUs[2] para permitir realizar computación paralela más potente. Las seis cámaras montadas en el vehículo cubren 360º de visión. Cada una de ellas tiene 100º FOV[3] y una frecuencia de 30 fps[4]. La información usada proviene de las imágenes de la cámara frontal. La Figura 4.3 muestra la configuración del vehículo eléctrico descripto.

La información inercial fue provista por una IMU colocada en las muñecas y tobillos de los peatones. Este dispositivo, estaba basado en una computadora Raspeberry Pi 2 junto con un módulo Sense HAT que incluye un acelerómetro, un giróscopo y un magnetómetro. Los datos de la IMU fueron adquiridos a 50 Hz que fue considerado lo suficientemente alto para capturar características cinemáticas de las extremidades de los individuos. La Raspberry Pi fue conectada al vehículo utilizando la red inalámbrica provista por el mismo. La marca de tiempo y la sincronización de la información de la visión y de los datos inerciales son esenciales para este trabajo, por lo que se prestó especial atención en su captura y almacenamiento.

Luego de la validación de la información obtenida de las cámaras, que se desarrolla en detalle en la Sección 4.2.2, se llevó a cabo una segunda etapa ex-

[2]Unidad de Procesamiento Gráfico -*Graphics Processing Unit*-s
[3]Campo de visión -*Field Of View*-
[4]cuadros por segundo -*frames per second*-

Figura 4.3: Vehículo eléctrico autónomo modernizado con tecnología de visión perceptiva.

perimental donde se conformó un conjunto de datos reales. Los mismos fueron obtenidos de un grupo de personas caminando en áreas públicas, llevando a cabo el tipo de acciones descriptas anteriormente, como ilustra la figura 4.4. Este conjunto de datos es usado para demostrar la viabilidad de la aproximación con todas las extremidades y en situaciones reales.

Figura 4.4: Dos peatones realizando diferentes acciones en la misma escena y en el mismo momento. Uno de ellos está parado en la borde la calle mientras el otro la está cruzando.

Los datos fueron recolectados en áreas públicas en calles alrededores del campus de la USyd[5], en Australia. Las acciones evaluadas en los peatones fueron: *parado*; *caminando* (despacio y rápido); *comenzando a caminar*; *caminando y*

[5]Universidad de Sídney -*The University of Sydney*-

luego parando; y *corriendo*. Estas son las actividades de mayor interés y que realizan los peatones en los entornos urbanos.

Aunque las imágenes en los experimentos fueron capturadas con el vehículo parado, el enfoque propuesto puede ser extendido a vehículos en movimiento. Para ello es necesario realizar correcciones en el esqueleto de los peatones para compensar el cambio de longitudes de los segmentos del mismo a medida que el vehículo se aproxima o aleja de estos. Por otro lado, también es importante considerar las vibraciones producidas durante el desplazamiento del vehículo, estabilizando los videos y evitando así obtener medidas con ruido no deseado.

4.2.2. Validación

La validación de la información se realizó mediante la comparación entre los datos inerciales medidos con la IMU y las dinámicas de las extremidades, estimadas a partir de los datos visuales provistos por la cámara. De este modo se comparó la velocidad angular y la aceleración lineal de las muñecas y tobillos obtenidos desde el acelerómetro y el giróscopo, con las correspondientes calculadas en base a los esqueletos en las imágenes. Especialmente, se analizó en profundidad la dinámica de la muñeca, como articulación de interés, debido al amplio movimiento que desarrolla y a la claridad de los datos obtenidos.

Para computar la validez de la información se utilizó el coeficiente de correlación cruzada. Este coeficiente es una medida de similaridad entre señales, con un valor igual a 1 para el grado de máxima similaridad entre las señales, y 0 para el caso contrario.

4.2.2.1. Con peatón detenido

Los primeros resultados fueron obtenidos de un peatón detenido que movía sus brazos hacia adelante y hacia atrás simulando el movimiento producido al caminar.

La Figura 4.5 muestra la información de la velocidad angular del eje Z del giróscopo. En línea azul se muestra la señal de la IMU, que fue colocada en la muñeca derecha del peatón mientras movía sus brazos. La línea naranja presenta la velocidad angular obtenida del procesamiento de la misma muñeca en la

representación del esqueleto derivada de las imágenes. Se puede observar que los valores son muy cercanos unos a otros. La similaridad entre las señales resulta en un 99,33 %. Se considera que este es un excelente resultado porque la información inercial del sensor visual puede compararse con la alta precisión del sensor portátil.

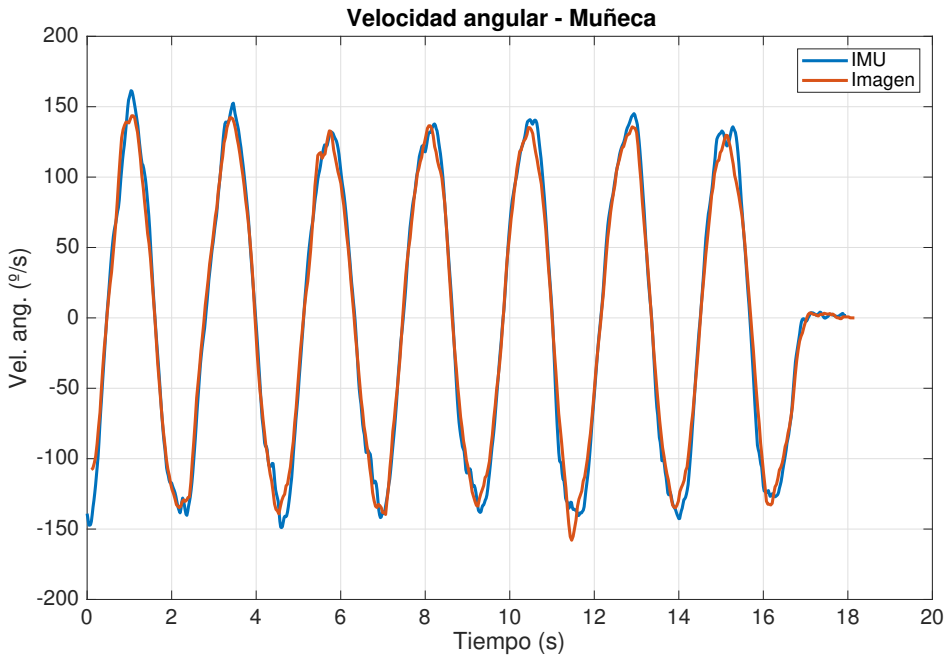


Figura 4.5: Comparación de la información de la velocidad angular de la muñeca. La línea azul representa la información del eje Z del giróscopo, mientras que la línea naranja es la velocidad angular obtenida del procesamiento del esqueleto en las imágenes.

Los resultados de la aceleración lineal no fueron tan precisos como los de la velocidad angular. Sin embargo, se correlacionan muy bien con la aceleración medida por los sensores portátiles. La frecuencia, la forma y el tiempo de las señales son muy similares, la discrepancia aparece sólo en la amplitud de las señales. Esto es más notable en el eje X como lo muestra la parte superior de la Figura 4.6, debiéndose al hecho de que la masa de la extremidad del peatón no es conocida cuando se calcula la aceleración utilizando el modelo de péndulo invertido. La similaridad en este eje se calculó en 79,76 %. En cuanto al eje Y, la aceleración lineal se encuentra más cercana a las medidas de la IMU en términos de ganancia y forma. En este caso la similaridad de este eje mejora y el resultado es de 83,31 %. Esta comparación se muestra en la parte inferior de la Figura 4.6.

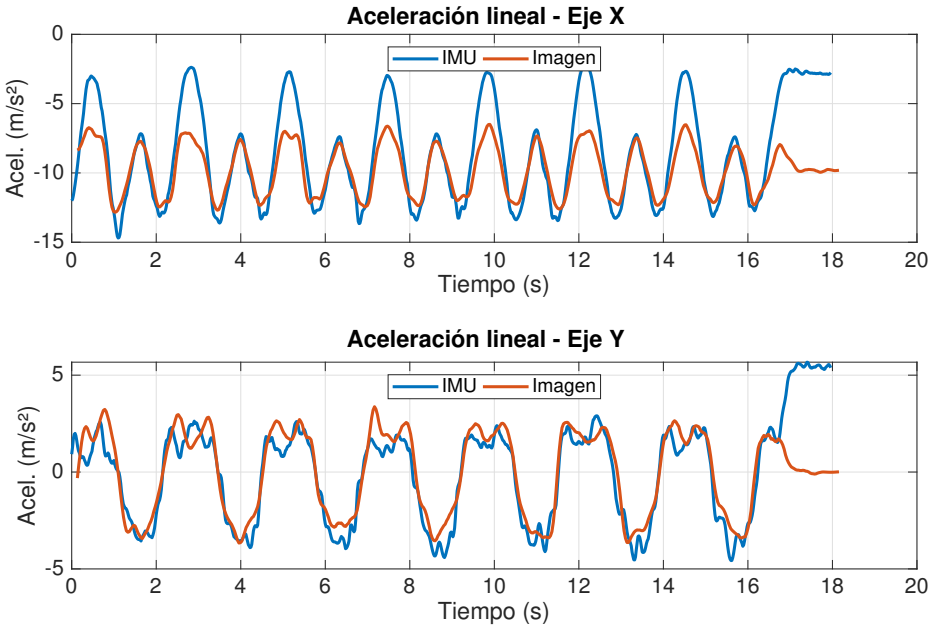


Figura 4.6: Comparación de la información de aceleración lineal de la muñeca. En ambos gráficos, la línea azul es la información provista por la IMU y la línea naranja es la información obtenida del procesamiento de las imágenes. El gráfico superior muestra la aceleración en el eje X mientras que el inferior muestra la del eje Y.

La Tabla 4.1 resume los resultados de la correlación cruzada del peatón detenido, para la velocidad angular y la aceleración lineal en los ejes X e Y.

Tabla 4.1: Correlación cruzada de un peatón detenido.

Período (s)	18,00
Velocidad angular (°/s)	99,33 %
Aceleración lineal, eje X (m/s^2)	79,76 %
Aceleración lineal, eje Y (m/s^2)	83,31 %

4.2.2.2. Con peatón caminando

Los siguientes resultados fueron obtenidos de peatones caminando. El experimento fue realizado por personas en diferentes lugares y desde diferentes distancias del vehículo, como lo muestra la Figura 4.7. En este caso los movimientos de los brazos son realizados naturalmente y no son forzados como en el primer caso. Esto permite obtener información realista y resultados en condiciones reales de movimiento, como sucede con los peatones caminando en la vía pública.

(a) Primer peatón

(b) Segundo peatón

Figura 4.7: Imágenes del segundo experimento tomadas de los videos. La representación de los esqueletos es presentada sobre los peatones.

El período de tiempo en que las acciones fueron realizadas es más corto que las del experimento previo debido al tiempo de acción del peatón al frente de la cámara del vehículo. Debido a que el FOV de la cámara es fijo, si la persona camina más alejada del vehículo, este podría tomarla con las cámaras por más tiempo. El caso opuesto sucede cuando la persona camina cerca. La dependencia de la distancia entre la persona y el vehículo no es un problema mayor ya que puede ser solucionado usando múltiples cámaras o cámaras con un campo de visión más amplio.

En líneas generales, la correlación cruzada de esta experiencia es menor que la de la anterior debido a la precisión de la representación del esqueleto realizada por *OpenPose* mientras los peatones caminan. La Figura 4.8 muestra la velocidad angular de ambas fuentes relacionadas al primer peatón. La línea azul representa el eje Z del giróscopo y la línea naranja representa la información obtenida del procesamiento de la imagen. La forma y el tiempo entre ambas son cercanos entre sí. Existen algunas diferencias en la amplitud, que se muestra particularmente en el extremo de los ángulos negativos. Estas diferencias se deben a los momentos de aceleración máxima, donde la resolución de los datos entregados por la representación del esqueleto no es suficiente para realizar el cálculo correcto de estos puntos. De todos modos, el resultado es una correlación de 97,48 %.

La Figura 4.9 muestra la misma información pero del segundo peatón donde se muestra una mayor discrepancia entre las trazas. La traza naranja obtenida del procesamiento de imágenes es más ruidosa que la azul, que es obtenida directamente del giróscopo. Esta diferencia se debe a que el segundo peatón está

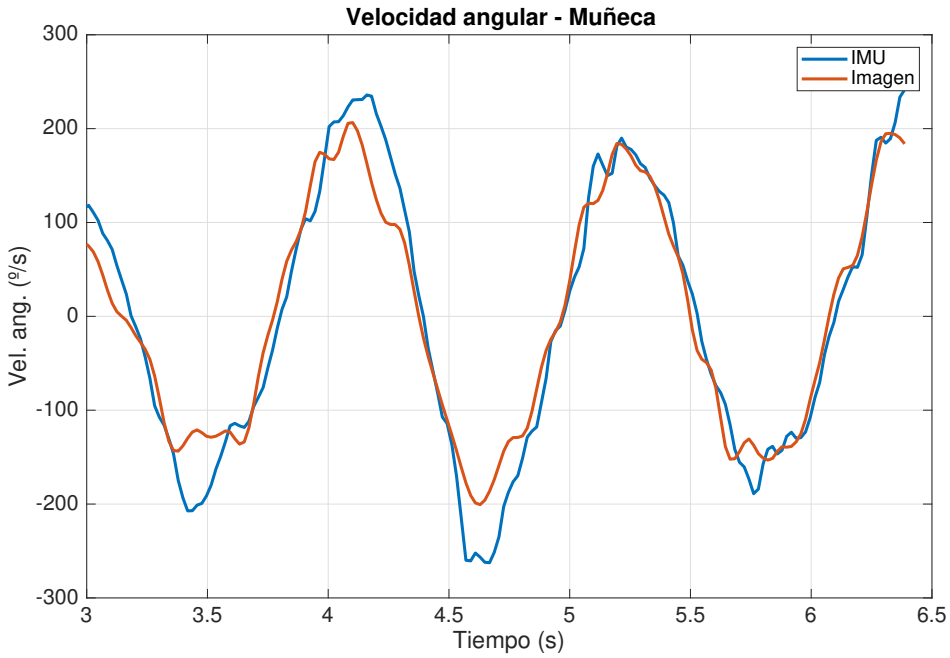


Figura 4.8: Velocidad angular del primer peatón caminando. La línea azul representa la información de la IMU mientras que la naranja representa la información desde el procesamiento de la imagen.

caminando más lejos que el primero en relación al vehículo, causando una variación en la representación del esqueleto. La variación se debe a la imprecisión de la biblioteca *OpenPose* al generar un esqueleto preciso cuando el sujeto en la imagen es chico respecto al tamaño de la imagen misma. En este caso, se interpreta como la lejanía del peatón al vehículo. Como consecuencia, la variación agrega ruido a los resultados, que no pueden ser tratados ni atenuados por ser causados por la herramienta utilizada. Sin embargo, la correlación cruzada es de 97,65 %, del mismo orden que los resultados del primer peatón. Esto se muestra resumido en la Tabla 4.2.

Tabla 4.2: Correlación cruzada de los peatones que caminan.

	Peatón 1	Peatón 2
Período (s)	3,4	5,0
Velocidad angular (°/s)	97,48 %	97,65 %
Aceleración lineal, eje X (m/s^2)	86,14 %	72,53 %
Aceleración lineal, eje Y (m/s^2)	73,23 %	71,62 %

Volviendo al primer peatón, el análisis del eje X de la aceleración lineal muestra una discrepancia menor en cuanto a la amplitud entre ambas señales, pero

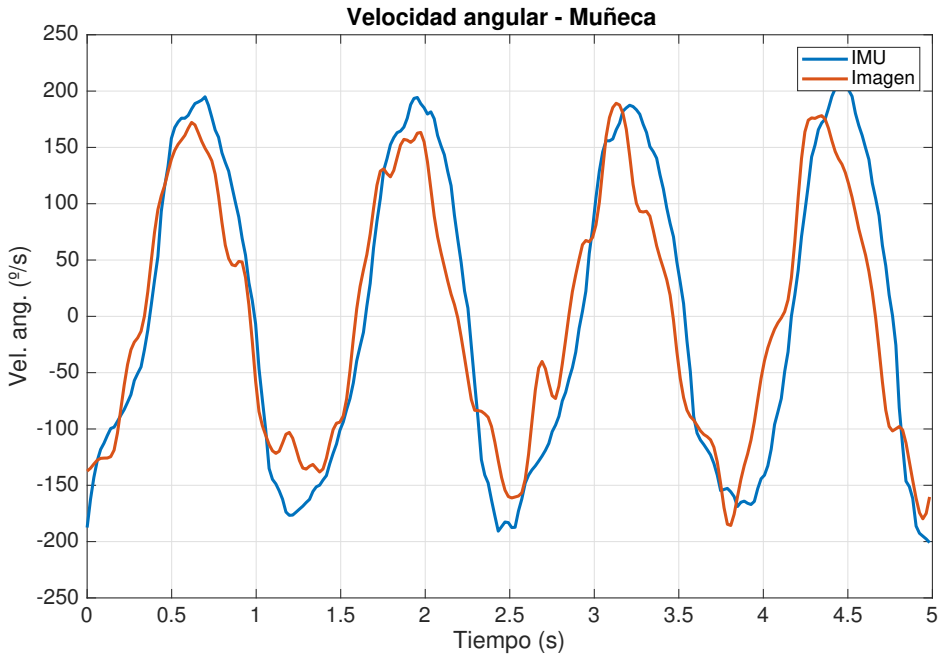


Figura 4.9: Velocidad angular del segundo peatón.

esto no afecta los resultados porque las formas de las señales son cercanas entre sí. El tiempo presenta un retraso de $t = 0,04s$ debido a la diferencia de muestras utilizadas para calcular la doble diferenciación y así obtener la aceleración. Los resultados de este análisis se pueden observar en la parte superior de la Figura 4.10, la cual muestra la información de la IMU en línea azul y la información de las imágenes en línea naranja. La correlación cruzada en este caso es de 86,14 %.

Además, como en el eje X, en el eje Y existe una diferencia en tiempo entre las trazas como resultado del método utilizado para calcular la aceleración. La misma se puede observar en la parte inferior de la Figura 4.10, donde la traza naranja, que es el resultado del procesamiento de la imagen, se encuentra desplazado hacia adelante en relación a la señal de la IMU representada en azul. Además, la señal naranja no posee tanto ruido como la azul. Para este eje, la correlación cruzada es de 73,23 %.

Los resultados de la aceleración lineal del segundo peatón caminando se muestra en la Figura 4.11, donde la señal del procesamiento de la imagen está trazada en naranja y la información de la IMU en azul. Es importante notar que el tiempo de este experimento es mayor al del experimento relacionado a la primera persona. Un tiempo mayor implica que la acción del peatón fue tomada durante

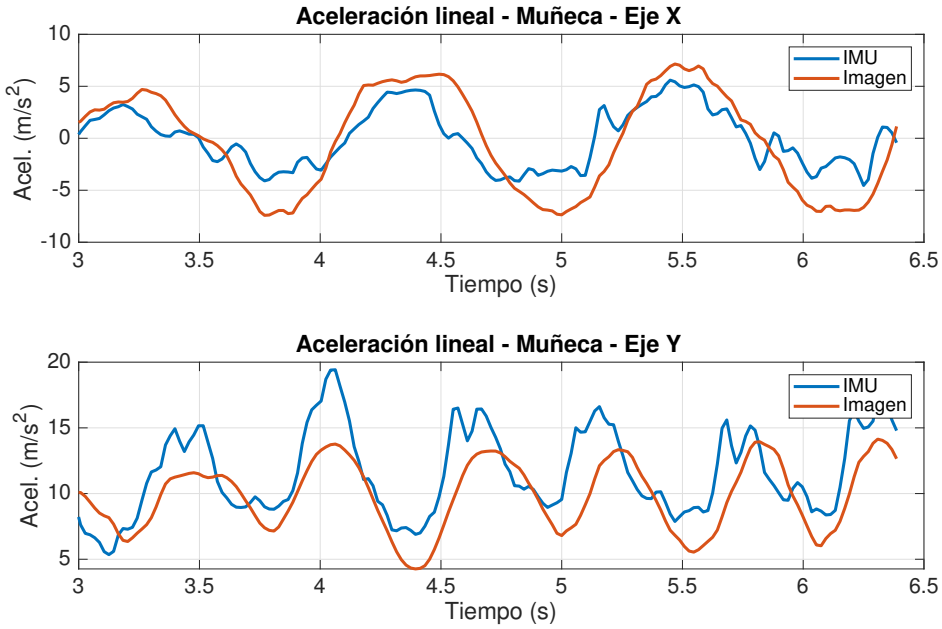


Figura 4.10: Aceleración lineal en el eje X (arriba) e Y (abajo) correspondientes al primer peatón. La comparación está dada por la información de la IMU, en línea azul, y el procesamiento de la imagen, en línea naranja.

más tiempo por la cámara. Si se lo compara con el peatón anterior, ambos se encontraban caminando a una velocidad aproximadamente similar y en direcciones perpendiculares al vehículo, por lo que se deduce que este último peatón lo hizo más alejado del vehículo y la cámara lo tuvo en su campo de visión por más tiempo. Y como se mencionó en algunos párrafos anteriores, la representación del esqueleto es menos precisa a medida que el peatón se encuentra más alejado de la cámara.

En el eje X, el valor de la correlación cruzada es de 72,53 %, menor que el del primer peatón debido a la precisión de la representación mencionada del esqueleto del peatón. La forma de la línea naranja no parece ser similar a la línea azul pero esto es por la diferencia en la amplitud de la señal. Además existe un pequeño retraso temporal, que es más notable en el eje Y. En este eje, cuya señales se muestran en la parte inferior de la Figura 4.11, hay un retraso de aproximadamente $t = 0,04s$, similar al retraso presentado en los resultados del primer peatón, producto de la doble diferenciación. Además, la línea naranja es menos ruidosa que la línea azul, como se mencionó antes. Sin embargo, la correlación cruzada es de 71,62 %. Esto es un buen resultado si se considera el

tiempo total del experimento y el ruido de la información de la IMU tomados para la comparación.

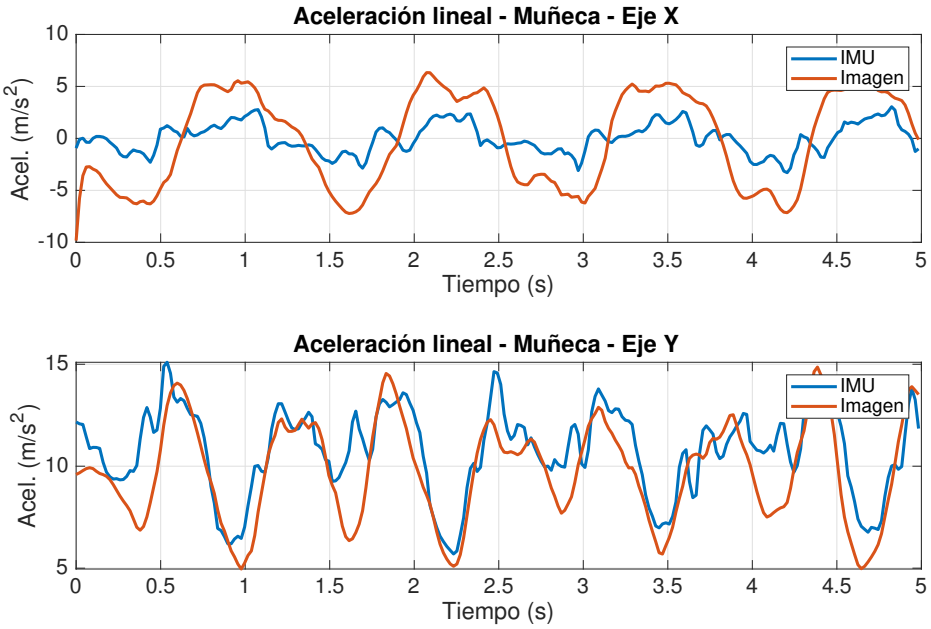


Figura 4.11: Aceleración lineal en ambos ejes correspondiente al segundo peatón.

En cuanto a la aceleración lineal, es de destacar que los valores de correlación cruzada de este último experimento son menores que los obtenidos en el primer experimento con un peatón detenido. Esto es un resultado lógico porque el movimiento de la persona contribuye a la inexactitud de la representación del esqueleto, derivando en resultados menos precisos.

4.2.3. Resultados

Los resultados validados son extendidos a otras partes del cuerpo y las dinámicas del esqueleto son evaluadas para diferentes tipos de acciones sobre un número diferente de peatones, dos de ellas, caminar y estar detenido, son presentados en esta sección. Los resultados obtenidos usando el presente método son divididos de acuerdo con diferentes partes del cuerpo como se muestra a continuación.

4.2.3.1. Análisis relativo a la dirección del peatón

Como parte de los resultados analizados es importante hacer algunos comentarios relacionados a la posición del peatón en la escena y a su dirección de

movimiento. En todos los casos propuestos, el peatón se mueve en el plano paralelo de captura de la cámara del vehículo. Esta condición es sugerida ya que es de interés el momento en que el peatón cruza la calle y que los experimentos se han realizado con el vehículo detenido. Por tal motivo es que las longitudes de los segmentos del esqueleto se mantienen constantes en la mayoría de los casos. Sin embargo, se han utilizado promedios de cada uno de los segmentos en muestras consecutivas, permitiendo obtener mejores resultados de la dinámica de estas partes y reducir posibles variaciones como las descriptas en la Sección 4.2.2.

En los casos en que la persona se aproxima o aleja del vehículo, como cuando se encuentra caminando por la vereda o cruza la senda peatonal en un sentido diferente y no paralelo a la misma, se presentan una serie de inconvenientes relacionados al ocultamiento de algunas extremidades y a las variaciones de las longitudes de los segmentos, haciendo que la confiabilidad de los resultados se vea reducida. Si bien la solución a este problema puede ser elaborada mediante el uso de otros sensores disponibles en el vehículo, como un láser o un sensor *bearing*, no es un tema que se aborda en esta tesis.

4.2.3.2. Extremidades

Los ángulos referidos a distintos segmentos del esqueleto de un peatón, realizando diferentes actividades, se muestran en la Figura 4.12. En ella se grafican los ángulos de diferentes articulaciones del brazo extraídas del esqueleto, mientras el peatón camina frente a la cámara, desde $t = 0s$ a $t = 2s$. Luego se detiene hasta $t = 3s$ y comienza a caminar de nuevo.

El tiempo durante el cual la persona está parada (desde $t = 2s$ a $t = 3s$) puede ser reconocido por medio de la baja variación de cambios en el ángulo en cada uno de los segmentos del brazo. La línea azul muestra el ángulo del segmento hombro-codo respecto a la línea horizontal. El promedio de este ángulo es alrededor de 90^o y sus variaciones son de sólo pocos grados. La línea roja muestra el ángulo del segmento codo-muñeca, desde la cual se puede diferenciar claramente entre caminar y estar detenido, y también reconocer la dirección del movimiento de la persona. En este caso, de acuerdo al ángulo establecido y mostrado en la

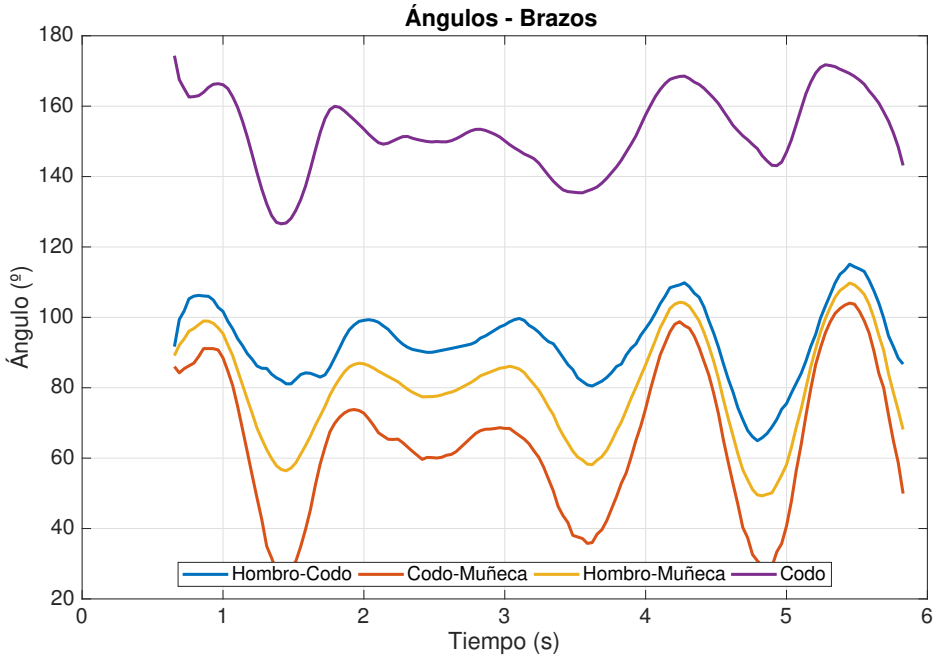


Figura 4.12: Ángulos entre diferentes articulaciones y segmentos de los brazos respecto a la línea horizontal, extraído desde el esqueleto de una persona que camina hasta $t = 2s$, se detiene hasta $t = 3s$ y luego camina de nuevo.

Figura 4.1b se puede determinar que la persona está caminando de izquierda a derecha porque el promedio del ángulo es menor a 90º. Cuando la persona se está moviendo de derecha a izquierda, el promedio de la señal se encuentra por encima de los 90º. La línea amarilla es el ángulo de la línea imaginaria que une el hombro con la muñeca. La línea púrpura es el ángulo del codo, comprendido entre los segmentos hombro-codo y codo-muñeca. La Figura 4.13 presenta la velocidad angular del segmento hombro-muñeca. Cuando el peatón se detiene luego de caminar, sus brazos conservan el movimiento natural que se atenúa luego de algunos segundos. Ésto hace que la velocidad no sea cero instantáneamente sino que converje a este valor de forma progresiva.

Las piernas, que están divididas en tres articulaciones: cadera, rodilla y tobillo (puntos 8-9-10 y 11-12-13 en la Figura 4.1a), también fueron analizadas para extraer información dinámica. La variación del ángulo de la rodilla indica actividades como caminar y subir o bajar el cordón cuneta. Si el ángulo entre el segmento cadera-rodilla de la pierna derecha e izquierda no cambia es porque el peatón está detenido. Este caso se muestra en la Figura 4.14 donde son presentados los ángulos entre los diferentes segmentos de las piernas. Los dos

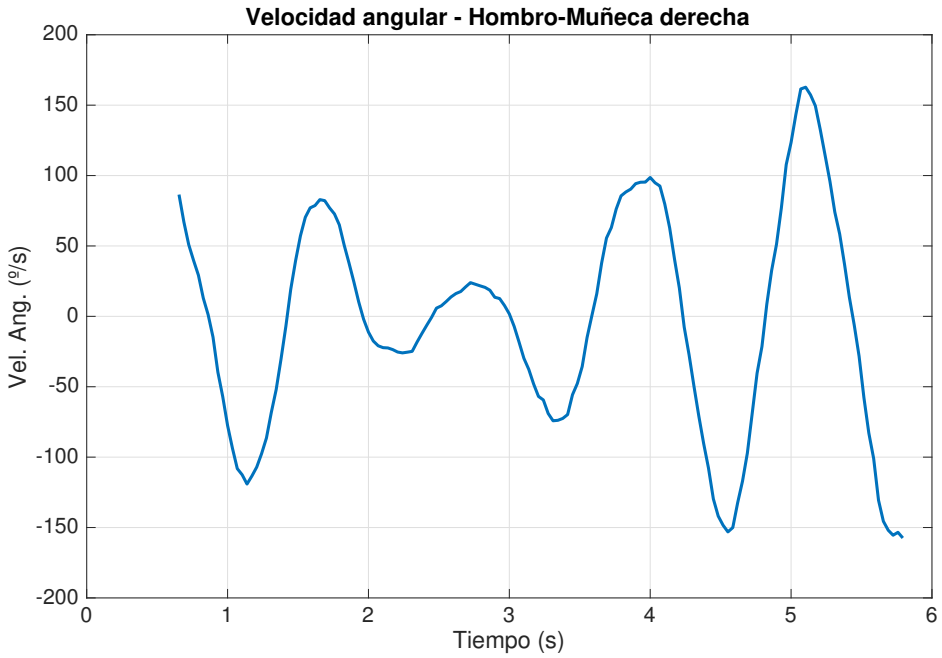


Figura 4.13: Velocidad angular del segmento hombro-muñeca del brazo derecho.

gráficos superiores muestran los ángulos del segmento cadera-rodilla y rodilla-tobillo, respecto a la línea horizontal, para ambas piernas. El momento en que el peatón está detenido es claramente visible entre $t = 2s$ y $t = 3s$, durante el cual los ángulos de ambas piernas es de 90^o, lo que significa que las piernas no se están moviendo. El gráfico inferior izquierdo presenta los ángulos del segmento imaginario cadera-tobillo, el cual muestra claras diferencias entre el tiempo en que caminó y estuvo parado. El gráfico inferior derecho muestra el ángulo de la rodilla, compuesto por la intersección de los segmentos cadera-rodilla y rodilla-tobillo, y la diferencia entre ambas, en línea punteada.

La Figura 4.15 muestra la velocidad angular del segmento rodilla-tobillo. Éste proporciona información clara que permite discriminar un peatón caminando o detenido. El patrón de la señal durante la caminata es claramente distinguible cuando la persona mueve su pie hacia adelante, mientras que en el de detenido la velocidad angular se encuentra aproximadamente en cero.

La velocidad lineal de un peatón puede obtenerse procesando la aceleración de las extremidades. La aceleración lineal también provee más información relacionada al estado del cuerpo, particularmente si está comenzando a moverse desde un estado detenido, como se muestra en el momento $t = 3s$ en la Figura

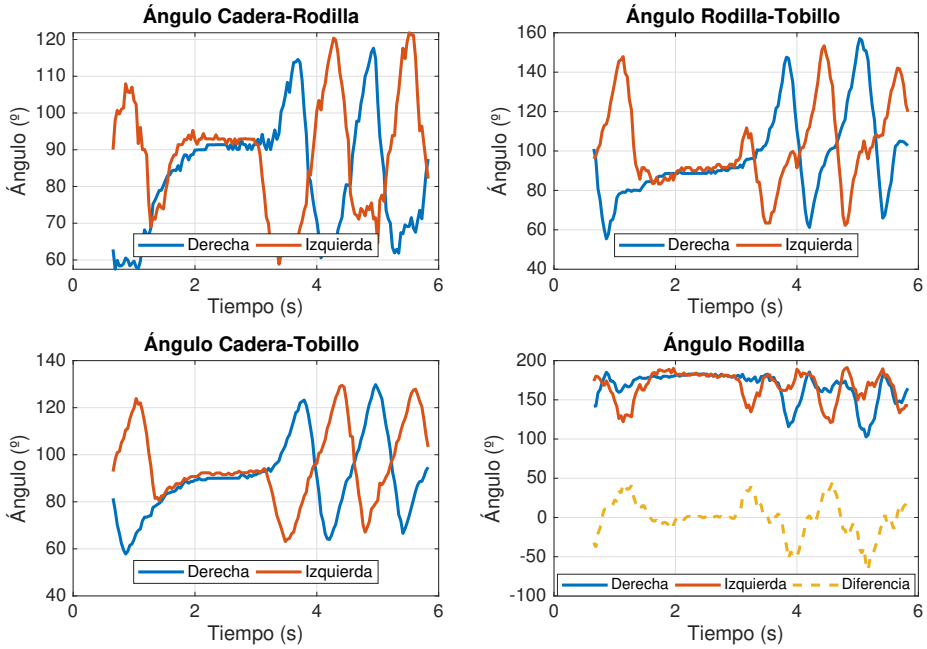


Figura 4.14: Ángulos entre diferentes articulaciones y secciones de las piernas, respecto a la línea horizontal. El último gráfico también muestra la diferencia del ángulo de las rodillas. La pierna derecha está coloreada en azul, la izquierda en naranja y la diferencia entre ambas en línea punteada amarilla.

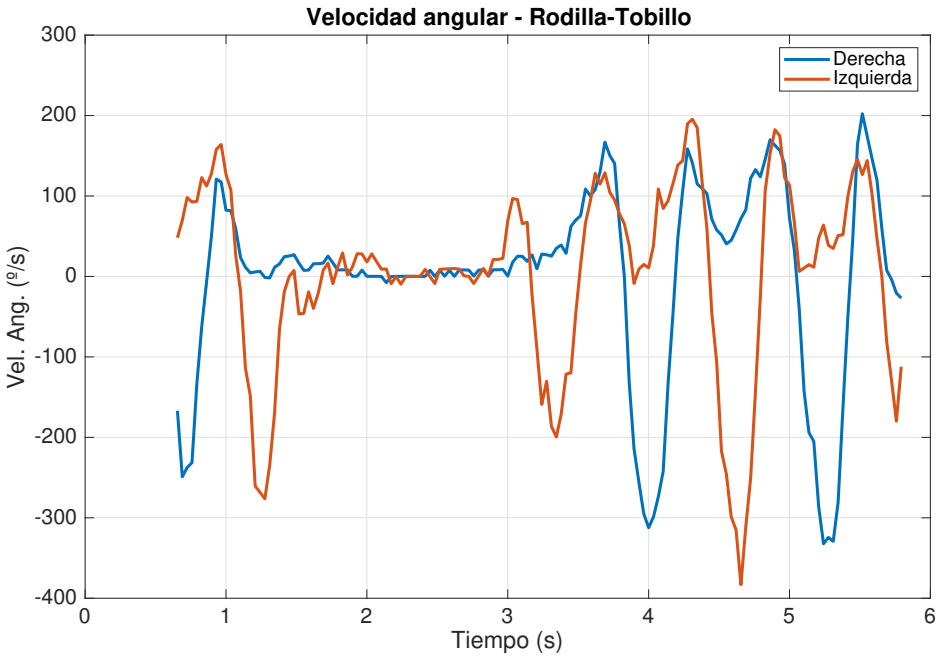


Figura 4.15: Velocidad angular de los segmentos rodilla-tobillo de ambas piernas. Es fácil reconocer las diferentes acciones llevadas a cabo por el peatón, como caminar y detenerse.

4.16. En este momento se destaca claramente como una de las piernas ejecuta el primer paso y luego lo hace la segunda, que se encontraba en reposo. En [80] se usa este tipo de información para calcular la longitud de la pierna usando el modelo de péndulo invertido.

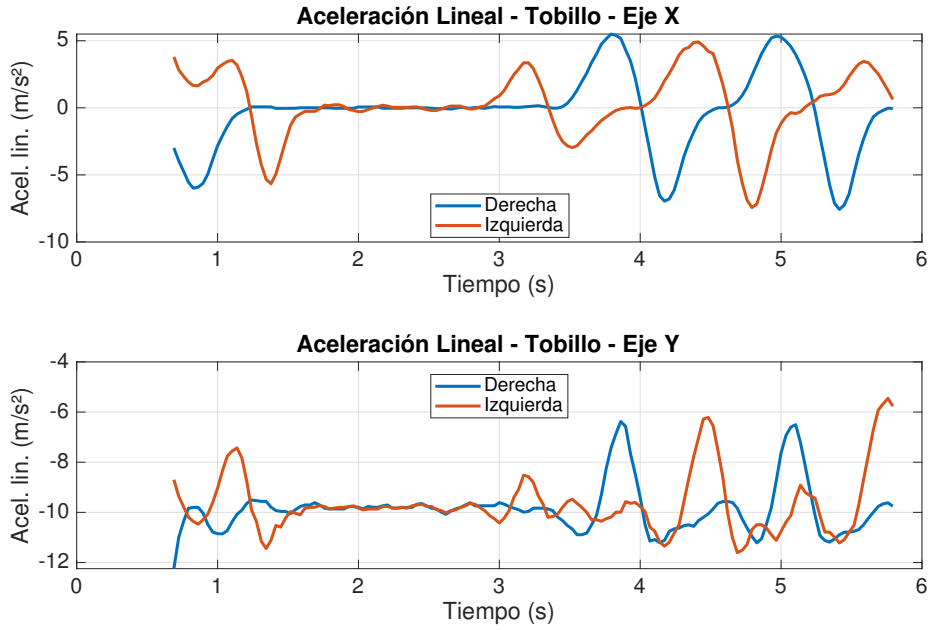


Figura 4.16: Aceleración de los tobillos en los ejes X e Y. La aceleración inicial, desde el momento detenido al de caminar se muestra claramente en el momento $t = 3$ segundos.

Un análisis de la utilización de este enfoque propuesto ha sido presentada en [81], donde los autores usan una IMU sujeta al tobillo para reconocer el ciclo de un paso de peatón. El avance del pie crea un patrón distintivo para reconocer, de acuerdo con los autores, seis actividades diferentes incluyendo caminata lenta, normal y rápida, correr y subir y bajar escaleras. Los resultados obtenidos en el enfoque propuesto en este capítulo, usando sólo visión, son comparables con los de [81] usando sensores inerciales.

4.3. Cálculo de la intención de cruce

A fin de evaluar la calidad de la información obtenida desde las cámaras, se la utilizó para obtener la intención de cruce con el uso de un clasificador. El algoritmo utilizado es el de subespacios aleatorios con clasificadores k-NN, el

cuál ya se ha utilizado para evaluar la información de aceleración provista por un acelerómetro en la Sección 3.5.

El conjunto de datos, utilizado para entrenar y validar la red, consiste en 87 secuencias de datos de peatones, desenvolviéndose en situaciones de tránsito reales y capturados por las cámaras del vehículo. Hay 42 casos en donde el peatón tiene la intención de cruzar la calle por el frente del vehículo y 45 casos en los que el peatón no tiene intención de cruzar. Los casos *cruza* y *no-cruza* son etiquetados manualmente en los videos de acuerdo a la acción final desarrollada. Todas las secuencias finalizan en momentos previos a que el peatón realice la acción, especialmente aquellos en los cuales el peatón cruzará frente al vehículo. Esto significa que los videos utilizados no muestran al peatón cruzando por delante del vehículo, sino que finalizan un momento antes de que lo haga. De este modo se logra entrenar la red durante el momento de intención, etapa previa a la acción, como se describió dentro de la Sección 3.3, evitando que la acción misma interfiera en la estimación de la intención.

Para el entrenamiento de la red no se consideraron otras características de la imagen como formas, bordes o segmentación del entorno del peatón, ni tampoco señales, cordones y carriles. Esto brinda la oportunidad de reconocer la intención de cruzar por frente al vehículo sin importar el lugar donde el peatón esté circulando, abriendo la posibilidad de aplicar este estudio a áreas donde los vehículos circulen en entornos no adaptados a ellos, como estacionamientos, plazas y parques.

En esta ocasión, la precisión obtenida por el clasificador resulta en 65.5 %. Es de esperarse que este resultado sea inferior a la precisión del clasificador de la Sección 3.5, la cual fue de 84.4 %. Uno de los motivos posibles de ello es que, si bien la información obtenida de las cámaras tiene alta correlación con la obtenida directamente de un acelerómetro, no es del 100 % y esto hace que el resultado final sea inferior al alcanzado en la primera instancia.

La Figura 4.17 muestra la matriz de confusión del clasificador con la totalidad de los datos de entrenamiento. En ella se puede observar los aciertos y errores producidos entre ambas clases.

Si se analiza la tasa de aciertos y errores de las clases verdaderas, podemos

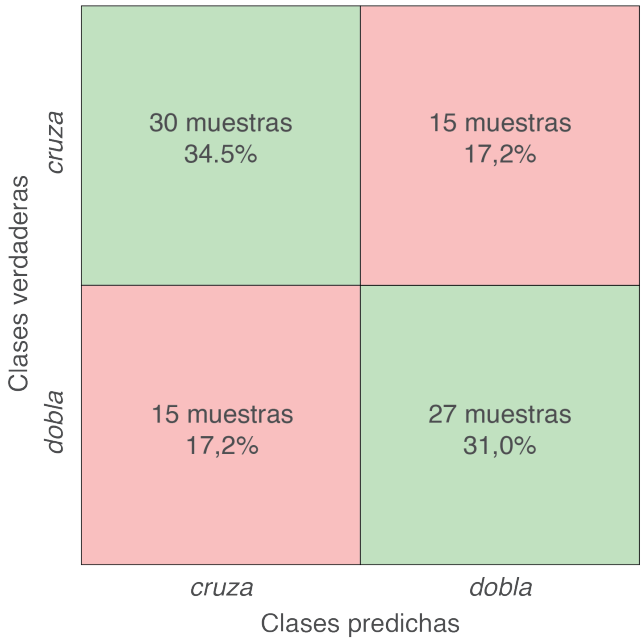


Figura 4.17: Matriz de confusión de la totalidad de los datos utilizados.

observar, en la Figura 4.18, que los verdaderos positivos y falsos negativos entre ambas clases son muy similares. Existe una pequeña ventaja en el acierto de la clase *cruza*, respecto de la clase *dobla*.

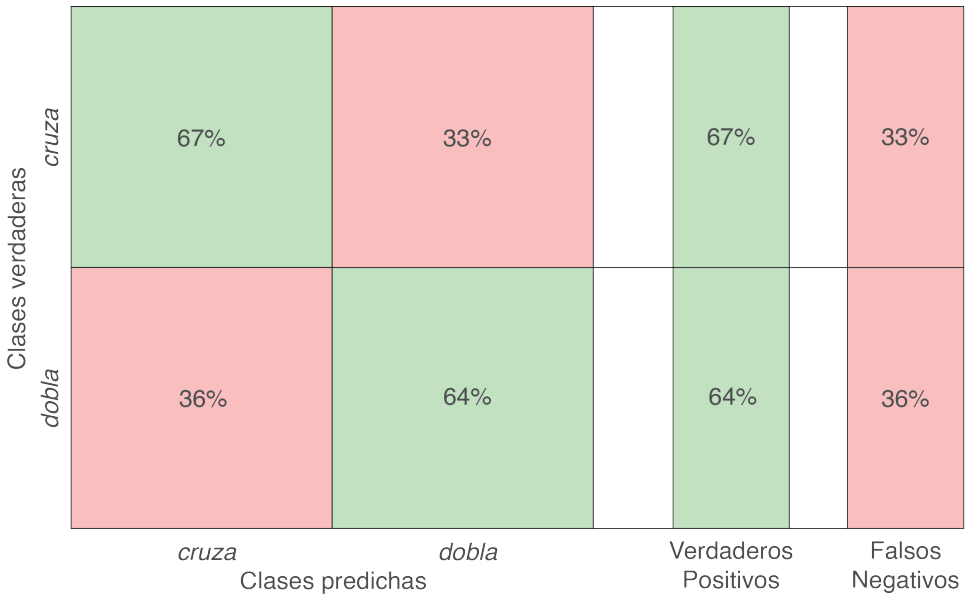


Figura 4.18: Tasa de verdaderos positivos y falsos negativos.

Por el lado de los valores predichos, el 67 % de las predicciones de *cruza* fueron acertadas con la intención verdadera, mientras que el 33 % fueron erróneas. En tanto, se obtuvieron resultados similares con la predicción de la clase *dobla*, donde

el 64 % fue correcto y el 33 % restante, mal predicho. Estos resultados se observan en la matriz de confusión de la Figura 4.19.

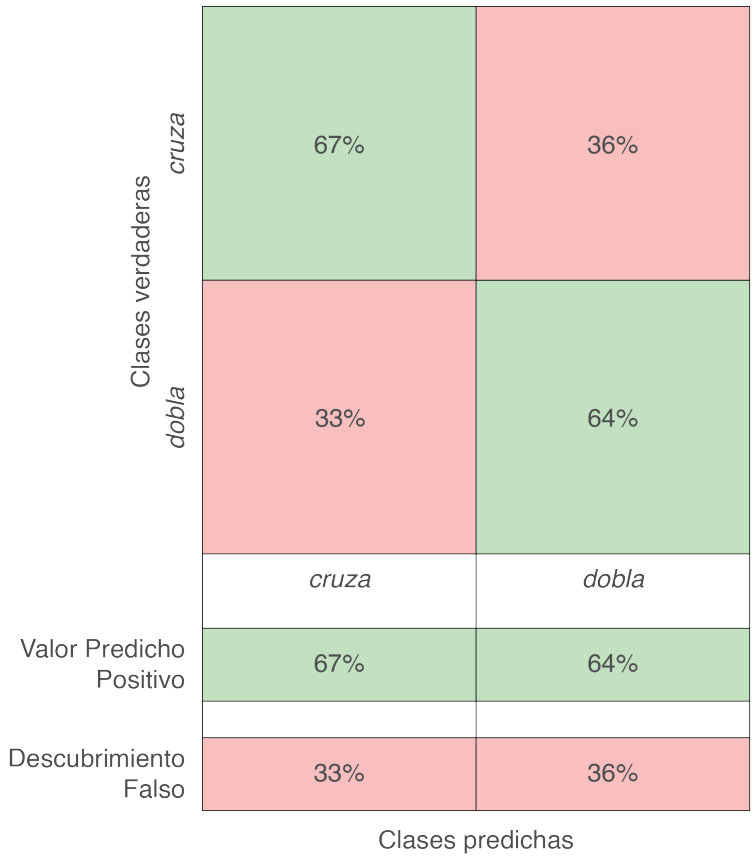


Figura 4.19: Valores predichos positivos y tasa de descubrimiento de falsos.

4.4. Conclusión del capítulo

El objetivo de este capítulo fue mostrar que la información dinámica y cinemática de un peatón puede ser obtenida desde sensores visuales montados en el vehículo. El enfoque presentado es capaz de extraer información dinámica de un peatón o grupo de peatones con una precisión comparable con la de los giróscopos y acelerómetros instalados en los dispositivos portátiles. Sumado a ello, es importante destacar que no es necesario establecer una comunicación directa con los dispositivos portátiles en los peatones ya que la información es obtenida directamente desde las cámaras del vehículo.

Del análisis de los peatones se obtuvo la representación del esqueleto y las posturas, seguido de diferentes acciones como caminar, detenerse, comenzar a

caminar y parar, que son fáciles de reconocer. Esta contribución no sólo mejora la detección de los peatones, sino que a partir de una secuencia de posturas del esqueleto, se puede extraer información más relevante como lo son las velocidades angulares y aceleraciones lineales, así como también las longitudes y los ángulos de diferentes segmentos del cuerpo. En la Sección 4.2.2 se demostró que los datos obtenidos desde los sensores de visión están fuertemente correlacionados con la mediciones hechas por una unidad inercial utilizada por peatones en diferentes partes del cuerpo. La velocidad angular es precisa en más de un 97 % en todos los casos. La aceleración lineal tiene una precisión menor que la velocidad angular, con un resultado del 73 % en el eje X y un 72 % en el eje Y.

El conjunto de datos utilizado fue recolectado usando peatones que seguían una secuencia de acciones definidas para validar las medidas obtenidas desde las imágenes. Un segundo conjunto de datos se compuso de datos de peatones reales que se desplazaban naturalmente y fue usado para generalizar el enfoque presentado, como se describió en la Sección 4.2.3.

Esta información es de fundamental importancia para estimar las actividades e intenciones peatonales. También se mostró que el método propuesto puede ser mejorado con otras técnicas, como semántica y algoritmos que sean capaces de corregir algunas de las fallas propias de la herramienta utilizada para obtener los esqueletos, y así aumentar la robustez y confiabilidad de los resultados. Además, el método propuesto puede ser extendido a otras articulaciones de interés como por ejemplo codos, rodillas y tobillos.

Por último, se evaluó la información adquirida utilizando un clasificador como el descripto en la Sección 3.5. Para ello se extendió el conjunto de datos incorporando más videos de peatones reales circulando por la vía pública. Los resultados de esta evaluación, si bien son inferiores a los del Capítulo 3, dan la pauta de que es necesario incursionar en el uso de redes neuronales más avanzadas, donde otro tipo de información pueda ser incorporada, especialmente aquella referida a sucesos temporales del pasado inmediato. Este enfoque se discute en el siguiente capítulo.

Capítulo 5

Intención del peatón basado en la dinámica del cuerpo

En este capítulo se presenta una estrategia para la detección de la intención peatonal de cruzar o no frente a un vehículo, utilizando el esqueleto virtual de los peatones y una red neuronal. Como ya se ha mencionado, la intención se define como la voluntad de una persona a realizar una acción objetivo, denotado por su movimiento y actitud, previo a realizar la acción, por lo que la estrategia se apoyará en estas características para obtenerla.

En un escenario habitual, el peatón podría tomar decisiones de riesgo mientras circula en un entorno donde interacciona con vehículos. Esto significa que para evitar accidentes, no sólo alcanza con detectarlo mientras camina por la vereda o se encuentra detenido en una esquina, sino que es importante reconocer de antemano cual es la acción que realizará, y esto se logra reconociendo la intención de hacerlo. Además, no sólo es importante saber que cruzará, sino que es de interés estimar si lo hará frente al vehículo, que es la situación de riesgo a la que se puede someter.

Aunque la tarea de reconocer la intención ha sido estudiada con anterioridad por varios autores, como se resume en [82], este tipo de trabajos en general requieren de información detallada de las posturas de los peatones en 3 dimensiones y tienen como objetivo la predicción de su trayectoria, más que la detección de la intención. Y, aunque varios de ellos utilizan redes neuronales que permiten hacer uso de información temporal, necesitan ser alimentadas con datos proveniente de

más sensores además de las cámaras. También se ha avanzado en la estimación de la intención utilizando sólo cámaras monoculares montadas en los vehículos y redes neuronales que permitan realizar estimaciones a partir de información histórica [83]. Sin embargo, el tiempo con el que se obtiene la intención no es suficiente para que un vehículo pueda realizar una acción que evite un posible accidente. Por tal motivo, en este capítulo se ha buscado entrenar redes con memoria temporal alimentándolas con información previa a la acción del peatón, permitiendo así que los resultados no se vean influenciados por el desarrollo de la acción misma. Para enfrentar este desafío se utilizan las cámaras frontales montadas en los vehículos, que entregan información de video, desde cuyas imágenes se extraen los esqueletos virtuales de las personas presentes en la escena. Esta información es procesada y utilizada para alimentar la red neuronal y así predecir la probabilidad de cruzar o no frente al vehículo con suficiente antelación a que el peatón realice la acción.

5.1. Redes neuronales para la determinación de la intención

En general, el proceso de clasificación de las acciones realizadas por las personas se logra a partir del procesamiento de información previa reciente. En otras ocasiones, especialmente aquellas en la que el desarrollo de la acción requiere de un proceso temporal, los clasificadores convencionales como las RNN[1] fallan en obtener resultados satisfactorios. A medida que la brecha temporal entre la información de interés y el momento de reconocer la actividad comienza a crecer, estas redes son incapaces de aprender a conectar la información [84]. Para resolver esta clase de problemas se han diseñado otros tipos de redes que logran almacenar en memoria largos periodos de tiempo, necesarios para concluir en un resultado final, denominadas *Long-Short Term Memory* (LSTM).

[1]Redes Neuronales Recurrentes -*Recurrent Neural Networks*-

5.1.1. Redes LSTM

Las redes LSTM son un tipo especial de RNN, capaces de aprender en base a dependencias de largo plazo. Estas redes fueron introducidas por [85] y en la actualidad son ampliamente usadas ya que funcionan muy bien en una gran variedad de problemas. Las redes LSTM fueron diseñadas para evitar el problema de dependencias de largo plazo, por lo cual se las utiliza para los casos en que se debe recordar información de largos periodos de tiempo. Su estructura es similar a las de las RNN, que son en forma de cadena, con celdas de estructuras simples y una función de activación. Sin embargo, las celdas la red LSTM son más complejas, al punto de interactuar con sus celdas vecinas por medio de 4 compuertas, como se muestra en la Figura 5.1.

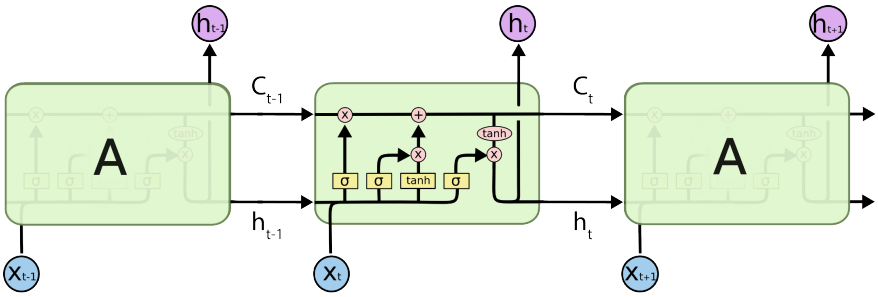


Figura 5.1: Representación del funcionamiento la una celda de una red LSTM.

A modo informativo, las compuertas de entradas a cada celda son:

- x_t el valor de la secuencia en el instante t, por donde ingresan los datos a partir de los cuales se desea que la red aprenda;
- h_{t-1} la salida de la celda LSTM en el paso anterior;
- C_{t-1} el estado de la celda LSTM en el paso anterior.

Y las de salidas:

- C_t como el nuevo el estado de la celda LSTM.
- h_t la salida de la celda LSTM en el instante t.

El funcionamiento de la celda consiste en decidir que tipo de información fluirá a través de ella, regulado por las funciones sigmoides mostradas en la figura. Cada

una de estas funciones es activada en base a información nueva ingresada por x_t y a la ya procesada por la celda anterior, por medio de h_t. Los flujos de datos son los que definen el estado de la celda, que se modifica en la línea de flujo superior, con los aportes del estado de la celda anterior C_{t-1}, los resultados de las funciones sigmoides y las funciones de activación definidas. El nuevo estado C_t es el encargado de transmitir el aprendizaje a la siguiente celda.

Una descripción más detallada de su funcionamiento se puede consultar en el Anexo A.

La red neuronal diseñada para este trabajo incluyó una serie de capas extras para acondicionar la información ingresada y los resultados de salida. La siguiente lista detalla el orden y las características de cada una de ellas.

- Una capa de secuencias de entradas para dar ingreso de la información a la red, utilizando un tamaño de entrada de 72 características, de acuerdo a la cantidad de datos de cada instante de la secuencia;
- Una capa LSTM bi-direccional que permite evaluar la información provista por la primer capa en ambos sentidos de la secuencia y así lograr un entrenamiento más robusto;
- Una capa totalmente conectada, que multiplica la entrada por una matriz de ponderación, luego agrega un vector de polarización y utiliza 2 neuronas como salida, una por cada etiqueta definida (*cruza* y *no-cruza*);
- Una capa de softmax que aplica la función homónima;
- Una capa de clasificación para obtener la decisión final de cada secuencia.

5.1.2. Obtención de información de peatones reales

El conjunto de datos usado en este Capítulo es el mismo que el utilizado para la evaluación de la intención en un clasificador en la Sección 4.3. Los datos fueron recolectados en condiciones de manejo reales, donde el comportamiento de los peatones, el clima y las condiciones de luz fueron variadas. Este punto es importante porque permite asegurar que la variabilidad de los datos brindará resultados con mayor confiabilidad. El conjunto de datos fue capturado en la

ciudad de Sídney, Australia, y se compone de información de video obtenida desde las cámaras montadas en el frente del vehículo. Las cámaras usadas tienen un amplio FOV que permite incluir en las imágenes mayor cantidad de objetos y captar un mayor tiempo de acción peatonal gracias a su amplio ángulo de apertura.

Para alimentar la red, los videos son divididos en fragmentos de 3 segundos de duración. Esta duración se debe a que un peatón promedio realiza al menos dos pasos con cada pierna mientras camina, lo cual es suficiente para obtener la dinámica de la acción. Esta información fue obtenida a partir de los resultados alcanzados en el Capítulo 4. Por otra parte, las cámaras utilizadas poseen una velocidad de captura de 25 cuadros por segundo, lo que significa que los 3 segundos de duración del fragmento serán traducidos a una secuencia de 75 cuadros. Cada uno de los cuadros es procesado por la biblioteca *OpenPose* para extraer el esqueleto de las personas presentes en él. Los esqueletos se componen de 18 puntos, que en general coinciden con las articulaciones del cuerpo, incluyendo aquellas de los brazos y piernas, así como también algunos puntos de la cabeza y el torso. Los puntos mencionados son coordenadas de la imagen, por lo cual poseen un valor sobre el eje X y otro sobre el eje Y.

Con esta información se obtienen los ángulos de cada uno de los segmentos de las extremidades del cuerpo, así como las velocidades angulares y las aceleraciones lineales. El procesamiento para obtener esta información se realiza con el método indicado en el Capítulo 4.

De cada punto de coordenada (X, Y) se calcularon dos ángulos extras a los calculados para cada segmento de las extremidades, lo que permite ayudar a la red a detectar las intenciones. Uno de estos ángulos, α, es el comprendido entre el eje X y la línea que une el vértice inferior izquierdo de la imagen con el punto mencionado, en sentido de las agujas del reloj. El otro, β, es similar pero calculado entre la línea conformada por el vértice inferior derecho y el punto, y el eje X, pero en sentido antihorario. Una representación de los ángulos se muestra en la Figura 5.2, donde se muestran los ángulos y las líneas de referencia para el punto que representa la rodilla derecha del peatón. Cabe mencionar que el par de ángulos $< \alpha, \beta >$ se calcula para todos los puntos del esqueleto mostrados en

la imagen.

La estrategia de utilizar los ángulos en la entrada de la red, junto con los puntos del esqueleto, radica en que el ruido en la detección de un mismo punto devenido en cuadros consecutivos representa pequeños cambios en los ángulos, a la vez que captura el movimiento de traslación del peatón frente al vehículo.

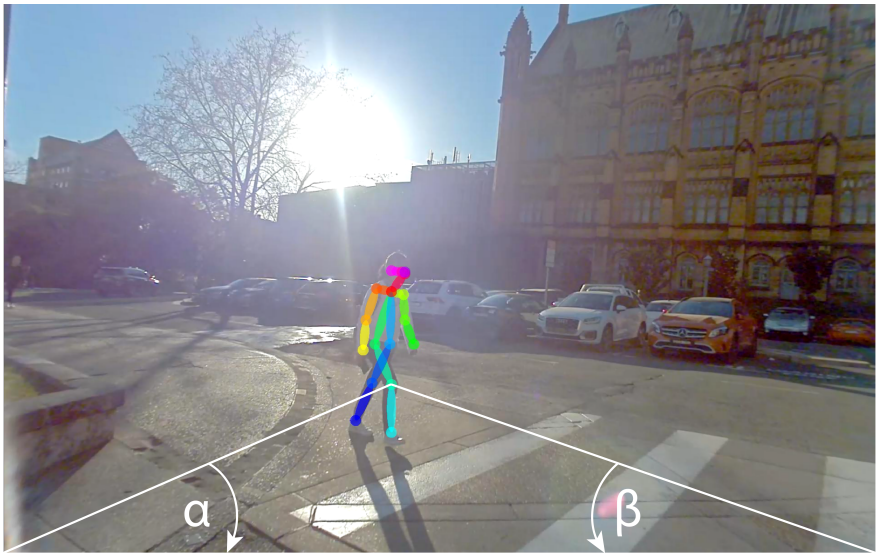


Figura 5.2: Representación de los ángulos usados como complemento a la información que alimenta a la red neuronal. En esta imagen se representan aquellos referidos al punto de la rodilla derecha del peatón. El ángulo α se calcula a partir del vértice inferior izquierdo y el ángulo β a partir del vértice inferior derecho.

5.1.3. Pre-entrenamiento de la red para la evaluación de la calidad de la información utilizada

Los datos obtenidos desde los esqueletos pueden dividirse en dos grupos de acuerdo a las características de la información. El primero de ellos se refiere a los de aceleraciones, velocidades y ángulos de los segmentos del cuerpo referidos a las extremidades del peatón. Esta información fue obtenida mediante las metodologías utilizadas en el Capítulo 4. El segundo grupo son los puntos de los esqueletos como coordenadas (X, Y) en las imágenes, junto con los ángulos calculados desde los vértices inferiores de las imágenes, como se explica en la Sección 5.1.2 de este capítulo.

Ambos grupos fueron utilizados conjuntamente para entrenar la red neuronal propuesta. En primera instancia, los resultados obtenidos no lograron estimar

las intenciones debido a la poca precisión alcanzada por la red durante el entrenamiento. Y, aunque se realizaron ajustes a la red y a los datos, no fueron suficientes para cumplir con los valores de precisión necesarios para estimar la intención correctamente. De este modo, y ante la incertidumbre y desconocimiento del motivo por el que ocurre este desenlace, se optó por evaluar ambos grupos de información de forma independiente, ajustando la red a cada uno de ellos para intentar mejorar los resultados.

Si bien la información que conforma el primer grupo es de calidad, lo que significa que es precisa y comparable con la de unidades inerciales, como se demuestra en el Capítulo 4, tampoco es suficiente para alcanzars lo resultados esperados por si sóla, consiguiendo como mejor precisión una red que puede estimar correctamente el 67 % de las entradas. Este valor no resulta prometedor si se espera poder contar con al menos un 75 % de estimación correcta.

La información del segundo grupo parece proveer de información más útil para la red ya que, entrenada con este grupo, se logró mejorar la precisión respecto a la alcanzada con el primer grupo. Por tal motivo, en las siguientes secciones se desarrollará en mayor detalle el proceso de entrenamiento con este tipo de información, los resultados alcanzados y un análisis de los mismos.

5.1.4. Entrenamiento de la red

El conjunto de datos fue dividido en un 80 % para el entrenamiento y el restante 20 % para la evaluación de la red. El proceso de entrenamiento realiza 3 iteraciones por cada ejecución y 150 ejecuciones, resultando en un total de 450 iteraciones. La Figura 5.3 muestra el avance en la mejora de la precisión del entrenamiento y el decrecimiento de las pérdidas a lo largo de cada iteración. Se destaca que luego de realizadas un poco más de 200 iteraciones, la red logra entrenar la totalidad de la información brindada alcanzando el 100 % del cometido. Luego de esta etapa se observa que prácticamente no existen pérdidas ocasionadas en el manejo de la información.

La Figura 5.4 muestra la matriz de confusión del proceso de entrenamiento, la cual logra un entrenamiento de la red del 100 % con una precisión del 76 % en el proceso de evaluación, como se muestra en la matriz de confusión de la Figura

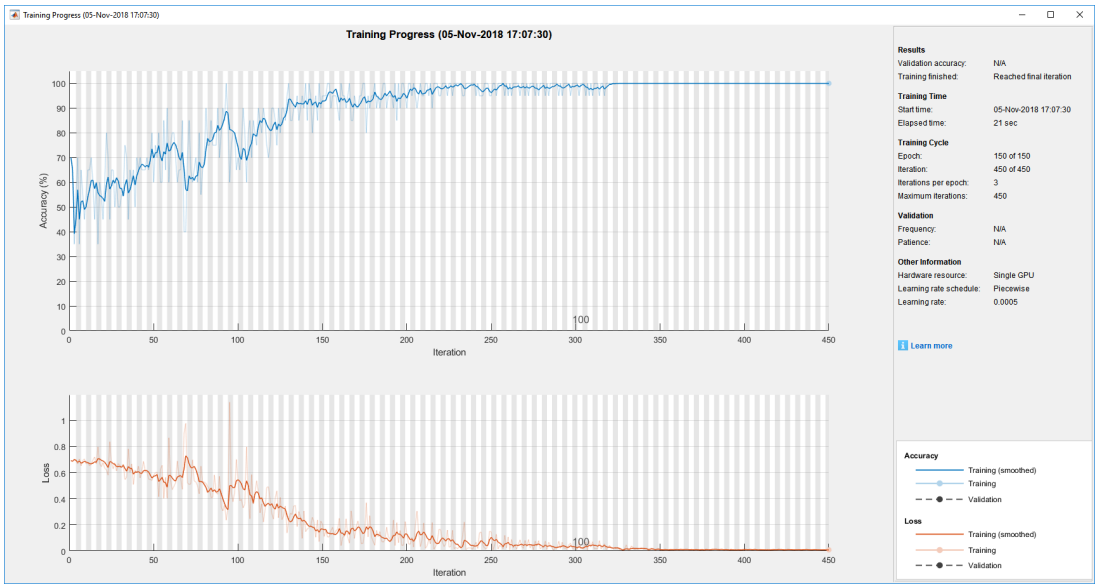


Figura 5.3: Progreso del entrenamiento de la red neuronal.

5.5.

5.2. Estimación de la intención

Para estimar la intención, canalizada por medio de la probabilidad de cruce, se evaluaron videos que no se utilizaron ni para el entrenamiento ni para la evaluación de la red. Si bien son videos del mismo tipo que los anteriores, ya que fueron obtenidos en las mismas condiciones y con el mismo instrumental, son de mayor duración. Su uso es para mostrar la actuación de la red a través de una secuencia de estimaciones y no sólo para un único punto probabilístico, como ocurre durante el entrenamiento y evaluación en los fragmentos de videos de corta duración. La secuencia de estimaciones permite evaluar los cambios de intención a medida que la escena transcurre y así poder observar fácilmente el desarrollo de los resultados a lo largo de las acciones previas a la esperada. Para llevarlo a cabo se utilizó una ventana de tiempo fija de 3 segundos (o 75 cuadros) que se desplazó por el video, obteniendo una secuencia de fragmentos de videos de duración fija y desplazadas 1 cuadro cada una de ellas. De esta forma y mediante el calculo de la probabilidad en cada ventana, se logra generar una secuencia de probabilidades que será analizada a posteriori y que representa el desarrollo de la intención a lo largo de toda la duración del video.

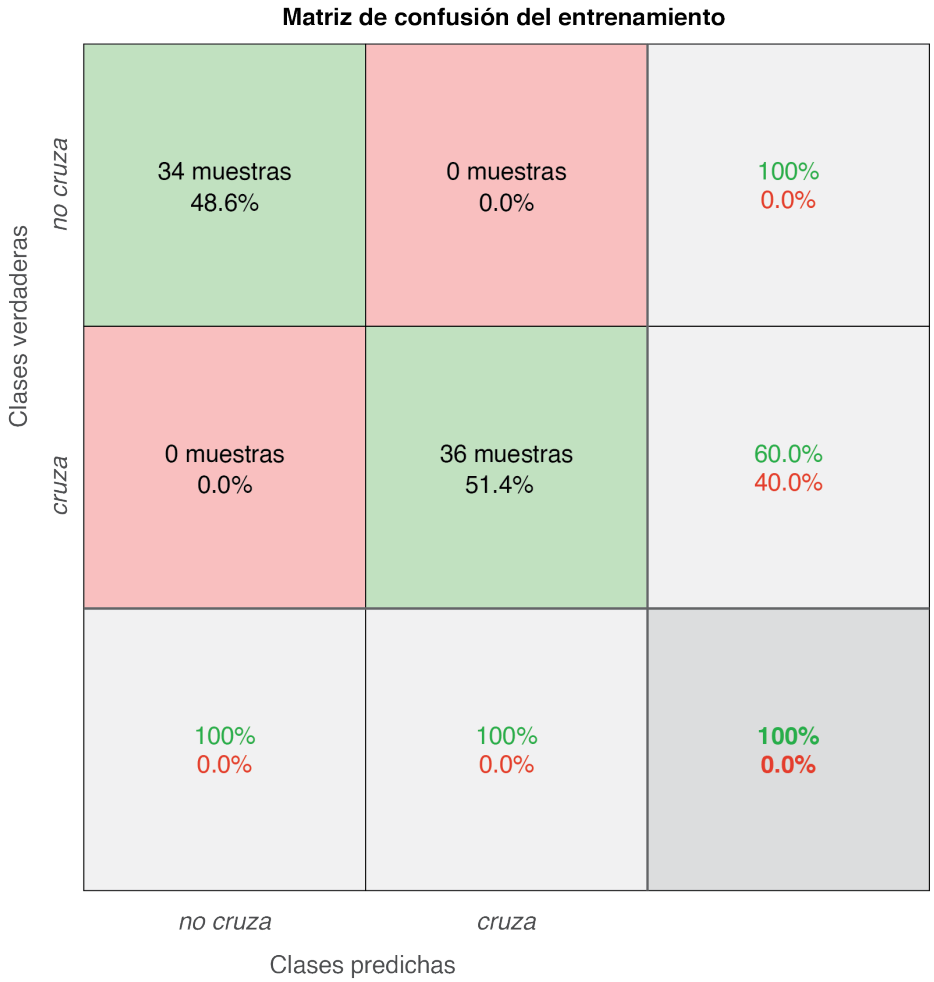


Figura 5.4: Matriz de confusión del proceso de entrenamiento.

Como se ha mencionado, la red necesita al menos 3 segundos para evaluar la primer estimación. Esto significa que luego de procesar la información de video de una ventana de 3 segundos, la red es capaz de estimar la probabilidad de cruce por primera vez. Durante este lapso de tiempo no es posible reconocer cual es la intención del peatón, y si este se encuentra cerca del vehículo, podría ejecutar la acción antes que la intención sea reconocida. Esto motiva a la evaluación de ventanas temporales móviles de menor duración a fin de advertir si es posible obtener la intención del peatón con más anticipación a la conseguida con la duración inicial. Por tal motivo, fueron evaluadas ventanas de 2 segundos (50 cuadros) y 1 segundo (25 cuadros), respectivamente. Para realizar la comparación entre los resultados de las 3 ventanas, se han tomado como videos experimentales

Figura 5.5: Matriz de confusión del proceso de evaluación.

4 en los que los peatones tienen intención de cruzar frente al vehículo, y 4 en los que los peatones no tienen intención de hacerlo.

En la Figura 5.6 se muestran los resultados adquiridos de 2 de los ejemplos evaluados, uno para cada intención de cruce. En la fila superior se encuentran los gráficos correspondientes a la evaluación del primer caso (cruza) y en la inferior los del segundo (no cruza), donde la primer columna corresponde a la ventana temporal de 3 segundos, la segunda columna a la de 2 segundos y la tercer columna corresponde a la ventana de 1 segundo de duración.

El eje X representa el número de cuadros de video totales. Para la primera columna, el gráfico comienza en el valor 75 porque esta es la cantidad de cuadros en la ventana de 3 segundos que la red necesita evaluar para obtener el primer

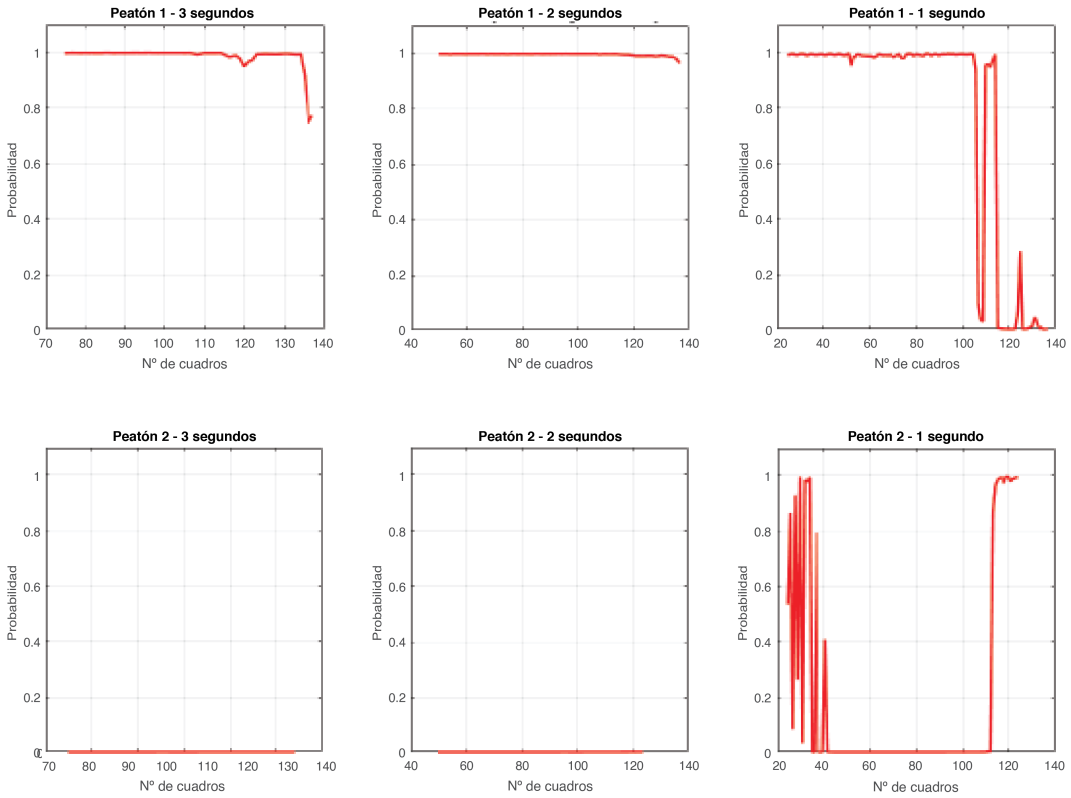


Figura 5.6: Gráficos para mostrar las fallas en el uso de ventanas temporales de diferente duración para obtener resultados de la red. Los gráficos superiores, de izquierda a derecha, son el resultado del uso de las ventanas de 3, 2 y 1 segundos para peatones con la intención de cruzar frente al vehículo (probabilidad ~ 1). Los gráficos inferiores están en el mismo orden del uso de las ventanas pero para peatones sin intención de cruzar (probabilidad ~ 0).

valor de probabilidad. Luego, 50 cuadros son utilizados para obtener el primer valor de probabilidad de la ventana de 2 segundos (segunda columna) y 25 cuadros para la de 1 segundo (tercer columna). El eje Y es la probabilidad de cruzar frente al vehículo, donde el valor 1 es el 100 % de probabilidad de que esto suceda (cruza), y 0 es el 0 % (no cruza).

5.2.1. Análisis de las estimaciones

Los gráficos en la primer columna de la Figura 5.6 son el resultado de la evaluación de la red con la ventana de 3 segundos. Éstos muestran claramente la probabilidad de cruzar frente al vehículo, en la fila superior, y no cruzar, en la inferior, como es esperado.

En la segunda columna están los gráficos de los resultados de la evaluación de la red con la ventana de 2 segundos. Con ésta es posible reconocer la intención

antes de realizar la acción de la misma forma que con la ventana anterior.

La tercer columna muestra los gráficos resultantes de la red evaluada con la ventana de 1 segundo. En este caso, al comienzo del gráfico de la primer fila, la probabilidad de cruzar es precisa, pero a medida que el tiempo avanza, la probabilidad cambia y el valor decrece a 0, significando que el peatón no tiene intención de cruzar, lo cual es erróneo para el caso que se está evaluando. El gráfico inferior de esta columna tiene fallas al comienzo de la estimación, pero no es un problema en este punto ya que al peatón aún le tomará alrededor de 3 segundos (∼75 cuadros) estar lo suficientemente cerca del vehículo como para presentar una advertencia al conductor. El problema real aparece al final del gráfico, cuando la probabilidad cambia al valor 1, significando una probabilidad de cruzar del 100 %, lo cual es un resultado también erróneo y no coincide con el valor esperado, que es 0 %.

De este pequeño análisis se descarta el uso de la ventana de 1 segundo debido a las incertidumbres producidas durante el proceso de reconocimientos de la intención. Sin embargo, las ventanas de 3 y 2 segundos parecen tener una alta correlación entre ambas. Por este motivo, se realiza un análisis extendido utilizando el resto de los videos evaluados con las 3 ventanas.

La Figura 5.7 muestra los resultados de los videos aplicando la ventana de 3 segundos, mientras que la Figura 5.8 lo hace con la ventana de 2 segundos. Cada gráfico individual de estas figuras corresponde a la evaluación de un solo peatón. En algunos videos se muestran dos personas caminando juntas, para este caso se han separado los análisis de cada una de ellas en gráficos diferentes.

Los videos disponibles en [86] y [87] representan toda la secuencia de imágenes utilizadas y los gráficos de intención para cada uno de ellos. Pero con motivo de realizar una descripción más detallada de los mismos, la siguiente lista muestra una descripción de las características de cada uno de los videos utilizados. Entre las más importantes a nombrar está: la cantidad de personas en el video, en los cuales a veces hay una y hasta dos personas en el mismo momento; el tiempo en el cual es importante reconocer la intención del peatón, ya que algunas veces éste se encuentra alejado y otras está muy próximo al vehículo, lo que incrementa la posibilidad de realizar una maniobra riesgosa; la cámara con la cuál se obtuvo el

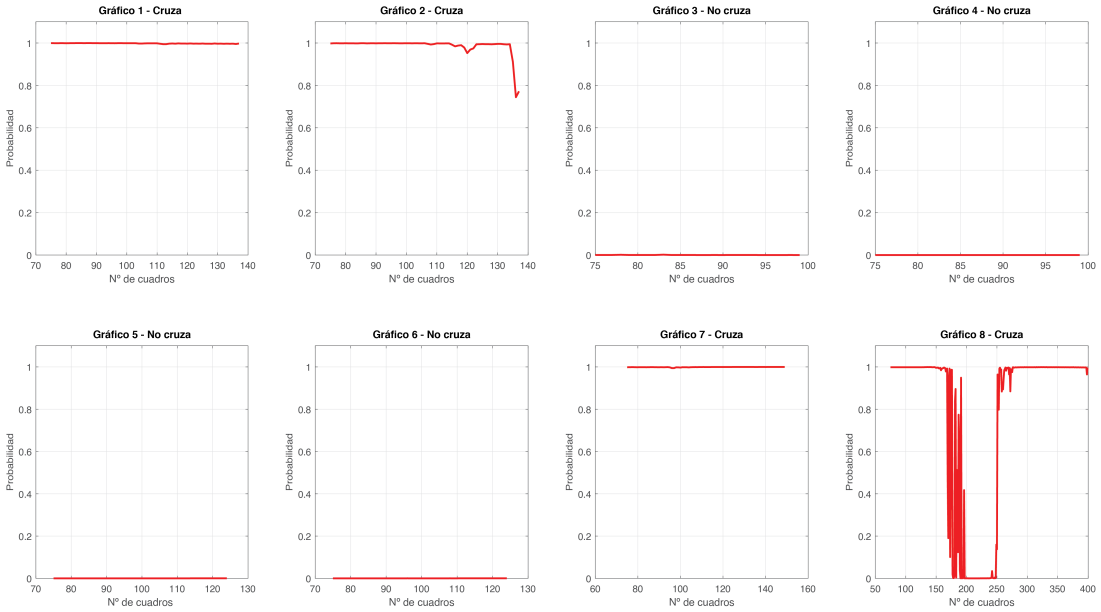


Figura 5.7: Probabilidad de cruzar basada en una ventana de entrada de 3 segundos.

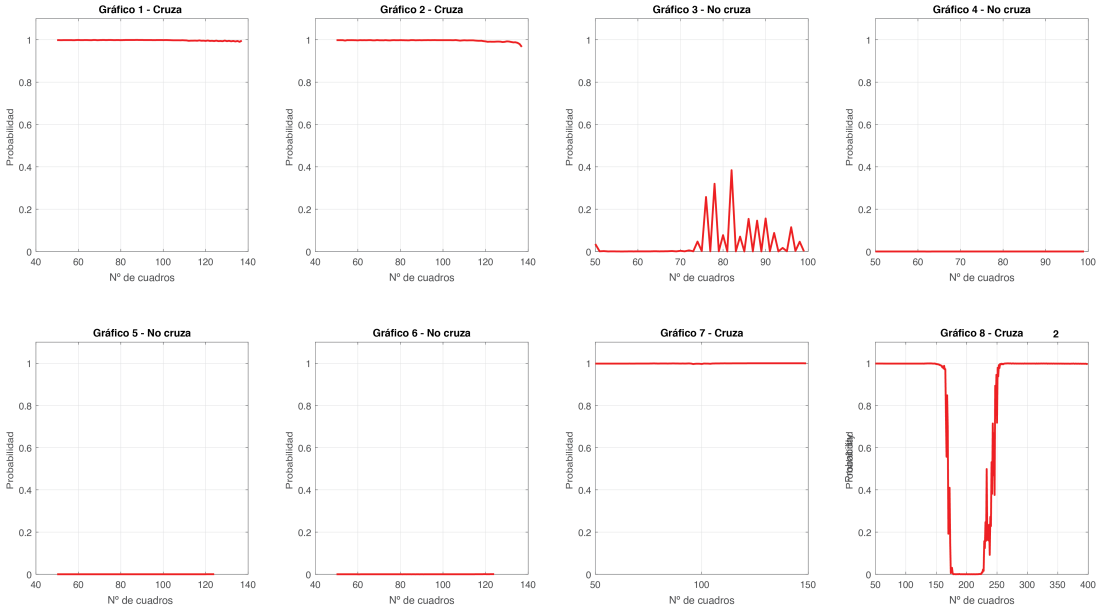


Figura 5.8: Probabilidad de cruzar basada en una ventana de entrada de 2 segundos.

video, que puede ser la frontal izquierda o derecha; el estado del vehículo, ya sea detenido o en movimiento; la acción que está desarrollando (principalmente) el peatón; y la intención que éste tiene de realizar la siguiente acción. Los gráficos individuales mantienen el orden, de izquierda a derecha y de arriba hacia abajo, y corresponden a:

- Gráfico 1: personas: 2 personas en la escena; tiempo: un momento antes

de cruzar frente al vehículo; cámara: frontal derecha; vehículo: detenido; acción: cruzando la calle; intención: cruzar.

- Gráfico 2: personas: 2 personas en la escena; tiempo: un momento antes de cruzar frente al vehículo; cámara: frontal derecha; vehículo: detenido; acción:cruzando la calle; intención: cruzar.
- Gráfico 3: personas: 1 persona en la escena; tiempo: en todo momento; cámara: frontal izquierda; vehículo: en movimiento; acción: parado en la calle; intención: no cruzar.
- Gráfico 4: personas: 1 persona en la escena; tiempo: en todo momento; cámara: frontal izquierda; vehículo: en movimiento; acción: caminando en la vereda; intención: no cruzar.
- Gráfico 5: personas: 2 personas en la escena; tiempo: en todo momento; cámara: frontal izquierda; vehículo: en movimiento; acción: caminando en la vereda; intención: no cruzar.
- Gráfico 6: personas: 2 personas en la escena; tiempo: en todo momento; cámara: frontal izquierda; vehículo: en movimiento; acción: caminando en la vereda; intención: no cruzar.
- Gráfico 7: personas: 1 persona en la escena; tiempo: un momento antes de cruzar frente al vehículo; cámara: frontal izquierda; vehículo: detenido; acción: caminando en una esquina para cruzar la calle; intención: cruzar.
- Gráfico 8: personas: 1 persona en la escena; tiempo: un momento antes de cruzar frente al vehículo; cámara: frontal derecha; vehículo: detenido; acción: caminando, luego esperando en la esquina y por último cruzando la calle; intención: cruzar (principalmente).

En los resultados de la Figura 5.7 la probabilidad de cruce coincide con los resultados esperados. Ésta está bien definida y es tan alta (alrededor de 1) o tan baja (alrededor de 0) como sea posible. Un caso interesante es el último gráfico porque la probabilidad refleja fehacientemente los cambios en la acción del peatón antes que ésta ocurra. En el video de este ejemplo, un peatón arriba

caminando a una esquina, espera un tiempo allí y luego comienza a cruzar la calle en dirección a donde se encuentra el vehículo.

Una imagen más detallada de este ejemplo se presenta en la Figura 5.9 donde se muestra la secuencia desarrollada, representada por 6 cuadros relevantes en distintos momentos. En la parte superior de cada cuadro se encuentra el gráfico de probabilidad de cruce hasta ese momento, traducido en la intención de hacerlo, y en la parte inferior, la imagen correspondiente a ese instante. La secuencia completa puede ser observada en el video disponible en [88], donde la imagen de la izquierda es la cámara del vehículo junto con la representación del esqueleto del peatón, y la gráfica derecha es una reproducción de la probabilidad de cruce. Esto permite demostrar la habilidad de la red en capturar la intención del peatón cuando todavía éste no ha comenzado a desarrollar la acción.

Mediante el análisis de la totalidad de los cuadros que componen esta secuencia, se desprende que la red es capaz de estimar la intención del peatón con al menos 1 segundo de anticipación. Este resultado se observa al momento en que la intención de cruzar del peatón cambia de 0 % a 100 % y, luego de $\sim 1s$ el peatón comienza a moverse con el objetivo de cruzar.

Sin embargo, para este caso, la acción ocurre aún cuando el peatón se encuentra en la vereda, alejado de la posición actual del vehículo. La distancia que los separa le toma al peatón al menos 9 segundos recorrerla. Para este lapso de tiempo, la red aún estima la intención de cruce con el 100 % de confianza.

En la evaluación con la ventana de 2 segundos que se muestra en la Figura 5.8, para los mismos videos, los resultados también son correctos. Sin embargo, aparece una alteración en el tercer gráfico. Aunque en la mayoría del tiempo la intención es bien calculada, en este caso la variación hace incrementar la probabilidad hasta el 30 %, que puede resultar en fallas si se desea utilizar esta información en ADAS.

5.3. Conclusión del capítulo

En un escenario realista, cuando el vehículo se encuentra en movimiento cerca de la vereda, la intención es conocida un momento antes de que el peatón ponga

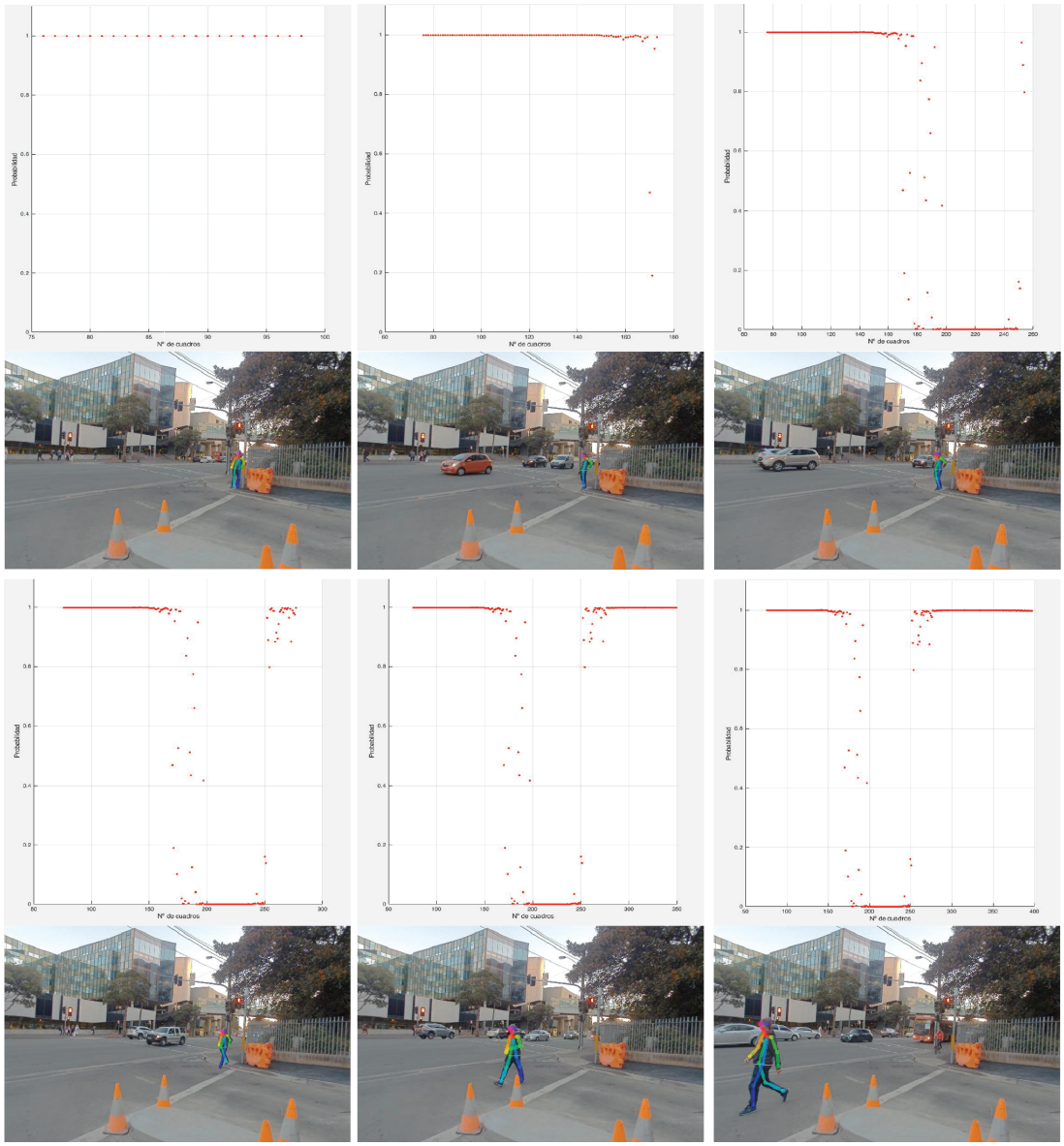


Figura 5.9: Secuencia de uno de los ejemplos más relevantes. En la fila superior se muestra el gráfico de la probabilidad y en la inferior la imagen de video correspondiente a tal, para 6 instantes de tiempo diferentes.

su pie en la calle si él está previamente caminando o parado en la vereda. Para reconocerla, se extrajeron los esqueletos de los peatones de las imágenes capturadas por el vehículo para analizar y se adecuó esta información para alimentar una red neuronal LSTM. Es importante aclarar que la información del contexto donde se encuentra el peatón no se usa. La imagen no es segmentada y los elementos como señales, bordes de la calle/vereda, carriles, etc, no son utilizados como información para alimentar la red.

Se evaluaron dos conjuntos diferentes de información pero relacionados a los

mismos videos, uno compuesto de ángulos de segmentos del cuerpo, velocidades angulares y aceleraciones lineales; el otro compuesto de puntos de la imagen coincidentes con las articulaciones y ángulos calculados en relación a los vértices de la imagen. Si bien el primer conjunto no logró alcanzar los resultados esperados, el motivo de esto podría ser la escasa cantidad de muestras evaluadas, consignadas en los conjuntos de datos adquiridos. Los videos de estas muestras comprenden amplias variaciones en la visibilidad y condiciones de tránsito, y se han capturado en un entorno de conducción realista. Por ello, es posible que, con esta cantidad de muestras, la red no pueda capturar la dinámica del peatón basandose en sus velocidades y aceleraciones para estimar correctamente la intención. Con un conjunto de datos mayor, la red sería capaz de evaluar situaciones y condiciones similares, permitiéndole realizar un entrenamiento más robusto y con mejor precisión en los resultados conseguidos. En tanto, el segundo conjunto sí tuvo un mejor rendimiento y es a partir del cual se obtuvieron los resultados analizados.

La red propuesta cumplió con el objetivo de obtener la probabilidad de cruce previo al momento en que el peatón realiza la acción. En los casos críticos, cuando se producen cambios de intención, la estimación de la misma se obtiene con al menos 1 segundo de anticipación antes de que la acción comience a ejecutarse. Se evaluaron diferentes ventanas temporales sobre los datos de video, concluyendo en que la de 3 segundos es la más apta para el fin propuesto, ya que con ella se entregan resultados correctos y ajustados en todas las pruebas realizadas.

Capítulo 6

Conclusiones

El peatón es un agente de tránsito vulnerable y debe ser protegido mediante los sistemas de prevención de accidentes de los vehículos. No sólo es importante detectarlo y reconocer su actividad, sino que es primordial conocer su acción futura previo a que la realice. Esta estimación no se trata de la predicción de su camino, sino más bien a la estimación de la intención de realizar una acción, particularmente de riesgo, como puede ser cruzar frente a un vehículo en movimiento. Esta tarea muchas veces se encuentra limitada a la información de que se obtiene de las personas que circulan por el entorno del mismo y aún más a la baja posibilidad de transmitirla a los vehículos.

Por ello, se ha comenzado el trabajo con el diseño de una plataforma robusta para la adquisición de datos. Su estructura presenta una arquitectura modular, permitiendo la ejecución de procesos independientes y la comunicación entre plataformas heterogéneas. La adquisición y almacenamiento de datos de los sensores propios del vehículo y de otras fuentes externas se realiza de forma automática, y toda la información es insertada en una base de datos lista para ser utilizada. Este proceso se realiza sin la intervención humana. La arquitectura propuesta ofrece una buena combinación de alta velocidad de ancho de banda y baja latencia, las cuales la aventajan con respecto a otras arquitecturas similares.

Sumado a esta plataforma se ha desarrollado una herramienta de simulación con la capacidad de mostrar los datos recolectados y permitir la evaluación de algoritmos. La ductilidad de un motor de videojuegos permitió la construcción de un simulador donde es posible generar experimentos controlados en contraposi-

ción a los ambientes abiertos, donde pueden ocurrir distracciones no deseadas. El simulador, que comprende un modelo virtual de la zona donde se han recolectado la información que se desea reproducir, permite el análisis de la información y su reproducción indefinida, así como también la evaluación de algoritmos que no pongan en riesgo personas y equipos.

La investigación se inició con la premisa de reconocer si la aceleración de las extremidades de los peatones brinda información relevante que pueda ser utilizada para inferir la intención de una persona que circula caminando, ya que es una de las medidas físicas más utilizadas en la bibliografía para este tipo de estimaciones. Los resultados indican que es posible estimar la acción a desarrollar por el peatón cuando se acerca a un cruce. Del análisis de las aceleraciones se puede estimar el momento en que el peatón disminuye su aceleración antes de cruzar la calle, lo que indica que la atención está enfocada en la acción a desarrollar. De esta forma, se procede a evaluar la posibilidad de capturar el mismo tipo de información, pero originada desde un vehículo, para evitar la problemática de comunicación entre éste y el peatón.

Por tal motivo, se propuso la utilización de las cámaras que actualmente poseen los vehículos inteligentes, con el fin de capturar no sólo las aceleraciones de las extremidades de los peatones, sino lograr el desarrollo de algoritmos que puedan obtener la dinámica del movimiento del mismo. Para cumplir con este objetivo se procesaron las imágenes de video obtenidas para extraer un esqueleto virtual de cada peatón en la escena y a partir del mismo obtener las velocidades y aceleraciones de las principales extremidades. Estas mediciones se validaron con información obtenida de unidades inerciales ubicadas en las mismas extremidades con resultados muy satisfactorios, demostrando así que la información dinámica de un peatón puede ser obtenida desde los sensores visuales montados en los vehículos. La precisión de las mediciones permitieron avanzar en este proceso y la información fue utilizada para las primeras aproximaciones a la estimación de la intención del peatón.

El siguiente paso es la estimación de la intención del peatón, es decir, la probabilidad de que ejecute una acción antes de que la realice, evaluando el historial de su dinámica. De esta forma se llevó a cabo una serie de métodos

destinados a alcanzar el resultado óptimo en la estimación. Se utilizaron redes neuronales con dependencia temporal capaces de recordar información de largos periodos de tiempo y se ajustó la información dinámica del peatón agregando información extra para facilitar el entrenamiento y mejorar los resultados en la evaluación de la red. Los resultados demuestran que la intención de cruce de un peatón por delante del vehículo que lo está evaluando, se puede obtener fehacientemente y con la antelación suficiente a la ocurrencia de la acción.

6.1. Proyección y trabajo futuro

El análisis de los resultados y las conclusiones alcanzadas permiten continuar con el objetivo de esta línea de investigación y plantear metas a alcanzar en la siguiente etapa. Como parte de las tareas necesarias para la continuidad, se deben desarrollar y analizar diferentes metodologías con el fin de hacer énfasis y extender los últimos resultados obtenidos. Para ello:

- Si bien los algoritmos fueron desarrollados en base a datos reales de peatones en situaciones de tránsito, organizados en una arquitectura flexible y confiable, es importante poder extender el análisis de los resultados a situaciones más extremas. Es en este punto que será esencial el uso del ambiente de simulación presentado.

- En todo lo presentado, se trató a la persona aisladamente del entorno. No se incluyeron factores como las condiciones de tránsito, las señales viales o incluso la actitud del conductor. En este punto de la tesis, aún no está definido si estos factores tienen influencia en el comportamiento del peatón. Se privilegió la búsqueda de una solución que no dependa del entorno pese a que es probable que éste, cuando es relevante (como la existencia de sendas peatonales), afecte la intención del peatón.

- Diseñar una estructura de redes neuronales en forma de árbol, utilizando redes LSTM y RNN, que permitan el ingreso de diferentes tipos de datos, no sólo de aquellos que se desarrollan a lo largo del tiempo, sino de otros como condiciones lumínicas y climatológicas, estado del tránsito, características

del entorno, etc. Datos que complementados entre si puedan proveer a estos algoritmos una mayor información sobre el peatón, su actividad y dinámica.

- Aunque en los conjuntos utilizados ya se ha incluido variabilidad en las condiciones en las que fueron tomados los videos, se debe incrementar la cantidad de capturas realizadas incluyendo condiciones climáticas más adversas para aumentar la confiabilidad en la detección de la intención.

6.2. Epílogo

¿Es posible detectar si una persona intentará cruzar delante de un vehículo antes de que su acción represente un riesgo para su integridad física? Esa es la pregunta esencial de esta tesis y el problema fundamental reside en la imposibilidad de medir directamente la intención de cruce. Además, no es evidente si los elementos del entorno, el movimiento del peatón o su actitud nos dará una pista confiable.

Es probable que el entorno por el que circula el peatón tenga un valor importante en una solución general. Si bien el futuro cercano encuentre en las calles vehículos autónomos, donde soluciones como las planteadas sean suficientes, durante mucho tiempo éstos coexistirán con vehículos aún conducidos por humanos y la actitud del conductor y su contacto visual entre éste y el peatón también podrían influir en la decisión.

En esta tesis se avanzó en el desarrollo de herramientas, en la comprensión de los datos disponibles y en las posibilidades de algunos algoritmos para poder inferir la intención y dar una respuesta a este interrogante. Todo lo presentado contribuye a contestar afirmativamente la pregunta original, aún cuando la solución definitiva requiere de continuar con las líneas planteadas y seguir profundizando en el análisis de los datos y de los casos extremos sin poner en riesgo las personas.

Bibliografía

[1] K. Guo, G. Yu, and Z. Li, "An new algorithm for analyzing driver's attention state," in *2009 IEEE Intelligent Vehicles Symposium*, June 2009, pp. 21–23.

[2] R. Mouček and J. Řeřicha, "Driver's attention during monotonous driving," in *2012 5th International Conference on BioMedical Engineering and Informatics*, Oct 2012, pp. 486–490.

[3] Y. Ducrocq, S. Bahrami, L. Duvieubourg, and F. Cabestaing, "A visual attention focusing system using an active stereoscopic vision sensor," in *2010 2nd International Conference on Image Processing Theory, Tools and Applications*, July 2010, pp. 511–516.

[4] P. P. Debnath, A. F. M. R. Hasan, and D. Das, "Detection and controlling of drivers' visual focus of attention," in *2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, Feb 2017, pp. 301–307.

[5] E. Dagan, O. Mano, G. Stein, and A. Shashua, "Forward collision warning with a single camera," in *Intelligent Vehicles Symposium, 2004 IEEE*, Junio 2004, pp. 37–42.

[6] P. Hurney, P. Waldron, F. Morgan, E. Jones, and M. Glavin, "Review of pedestrian detection techniques in automotive far-infrared video," *Intelligent Transport Systems, IET*, vol. 9, no. 8, pp. 824–832, 2015.

[7] D. Gerónimo and A. M. López, *Vision-based pedestrian protection systems for intelligent vehicles.* Springer, 2013.

[8] F. Flohr, M. Dumitru-Guzu, J. F. P. Kooij, and D. M. Gavrila, "A probabilistic framework for joint pedestrian head and body orientation estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1872–1882, Aug 2015.

[9] Z. Fang and A. M. López, "Is the pedestrian going to cross? answering by 2d pose estimation," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1271–1276.

[10] R. Kelley, M. Nicolescu, A. Tavakkoli, M. Nicolescu, C. King, and G. Bebis, "Understanding human intentions via hidden markov models in autonomous mobile robots," in *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, March 2008, pp. 367–374.

[11] S. Koehler, M. Goldhammer, S. Bauer, K. Doll, U. Brunsmann, and K. Dietmayer, "Early detection of the pedestrian's intention to cross the street," in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, Septiembre 2012, pp. 1759–1764.

[12] S. Koehler, M. Goldhammer, S. Bauer, S. Zecha, K. Doll, U. Brunsmann, and K. Dietmayer, "Stationary detection of the pedestrian's intention at intersections," *IEEE Intelligent Transportation Systems Magazine*, vol. 5, no. 4, pp. 87–99, 2013.

[13] Mobileye technologies limited. [Online]. Available: http://www.mobileye.com/

[14] C. Keller and D. Gavrila, "Will the pedestrian cross? a study on pedestrian path prediction," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 2, pp. 494–506, Abril 2014.

[15] K. C. Fuerstenberg, K. C. J. Dietmayer, and V. Willhoeft, "Pedestrian recognition in urban traffic using a vehicle based multilayer laserscanner," in *Intelligent Vehicle Symposium, 2002. IEEE*, vol. 1, Junio 2002, pp. 31–35 vol.1.

[16] (2014) Ibeo Automotive Systems GmbH. [Online]. Available: http://www.ibeo-as.com

[17] H. Ritter and H. Rohling, "Pedestrian detection based on automotive radar," in *Radar Systems, 2007 IET International Conference on*, Octubre 2007, pp. 1–4.

[18] C. Wong, Z. Q. Zhang, B. Lo, and G. Z. Yang, "Wearable sensing for solid biomechanics: A review," *IEEE Sensors Journal*, vol. 15, no. 5, pp. 2747–2760, Mayo 2015.

[19] T. Sztyler and H. Stuckenschmidt, "On-body localization of wearable devices: An investigation of position-aware activity recognition," in *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Marzo 2016, pp. 1–9.

[20] T. Sztyler, "Towards real world activity recognition from wearable devices," in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Marzo 2017, pp. 97–98.

[21] P. Merdrignac, O. Shagdar, and F. Nashashibi, "Fusion of perception and v2p communication systems for the safety of vulnerable road users," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–12, 2017.

[22] J. W. Shin, J. H. Oh, S. M. Lee, J. J. Ko, S. Y. Lee, and S. E. Lee, "In-vehicle can fd network for smart wearable devices," in *2017 IEEE International Conference on Consumer Electronics (ICCE)*, Enero 2017, pp. 45–46.

[23] A. T. M. Nakamura, L. R. T. Horita, and V. Grassi, "A stereo cameras setup for pedestrian detection enhancement," in *2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*, Octubre 2017, pp. 352–357.

[24] O. Yechiel, A. Livne, R. R. Hagege, and H. Guterman, "Performance boosting of a pedestrian detector using multiple cameras," in *2016 IEEE In-*

ternational Conference on the Science of Electrical Engineering (ICSEE), Noviembre 2016, pp. 1–5.

[25] X. Li, L. Li, F. Flohr, J. Wang, H. Xiong, M. Bernhard, S. Pan, D. M. Gavrila, and K. Li, "A unified framework for concurrent pedestrian and cyclist detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 2, pp. 269–281, Febrero 2017.

[26] D. Merad, K. E. Aziz, and N. Thome, "Fast people counting using head detection from skeleton graph," in *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, Agosto 2010, pp. 233–240.

[27] Y. Chen, Q. Wu, and X. He, "Motion based pedestrian recognition," in *2008 Congress on Image and Signal Processing*, vol. 2, Mayo 2008, pp. 376–380.

[28] M. Kilicarslan, J. Y. Zheng, and K. Raptis, "Pedestrain detection from motion," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, Diciembre 2016, pp. 1857–1863.

[29] Q. Zou, L. Ni, Q. Wang, Q. Li, and S. Wang, "Robust gait recognition by integrating inertial and rgbd sensors," *IEEE Transactions on Cybernetics*, vol. 48, no. 4, pp. 1136–1150, Abril 2018.

[30] A. Bender, J. R. Ward, S. Worrall, M. L. Moreyra, S. Gerling Konrad, F. Masson, and E. M. Nebot, "A flexible system architecture for acquisition and storage of naturalistic driving data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1748–1761, Junio 2016.

[31] S. Gerling Konrad, M. L. Moreyra, and F. R. Masson, "The use of game engines and virtual models to build a simulator for intelligent transportation systems," in *2015 XVI Workshop on Information Processing and Control (RPIC)*, Octubre 2015, pp. 1–6.

[32] S. Gerling Konrad and F. R. Masson, "Uso de acelerómetro para la detección de intención de peatones," in *2016 XXV Congreso Argentino de Control Automático (AADECA)*, Noviembre 2016.

[33] S. Gerling Konrad and F. R. Masson, "Pedestrian intention estimation from egocentric data," in *2017 XVII Workshop on Information Processing and Control (RPIC)*, Septiembre 2017, pp. 1–5.

[34] S. Gerling Konrad, F. R. Masson, and E. Nebot, "Analysis of accuracy of pedestrian inertial data obtained from camera's images," in *2018 IEEE Biennial Congress of Argentina (ARGENCON)*, June 2018, pp. 1–5.

[35] S. Gerling Konrad, M. Shan, F. R. Masson, S. Worrall, and E. Nebot, "Pedestrian dynamic and kinematic information obtained from vision sensors," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1299–1305.

[36] H. Guo, J. Pang, L. Han, and Z. Shan, "Flight data visualization for simulation & evaluation: A general framework," in *Computational Intelligence and Design (ISCID), 2012 Fifth International Symposium on*, vol. 1, Octubre 2012, pp. 497–502.

[37] M. Persson and P. Wide, "Using a sensor source intelligence cell to connect and distribute visual information from a commercial game engine in a disaster management exercise," in *Instrumentation and Measurement Technology Conference Proceedings, 2007. IMTC 2007. IEEE*, Mayo 2007, pp. 1–5.

[38] Q. Miao, F. Zhu, Y. Lv, C. Cheng, C. Chen, and X. Qiu, "A game-engine-based platform for modeling and computing artificial transportation systems," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 2, pp. 343–353, Junio 2011.

[39] A. Gregoriades, C. Florides, V. Lesta, and M. Pampaka, "Driver behaviour analysis through simulation," in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, Octubre 2013, pp. 3681–3686.

[40] A. S. Huang, E. Olson, and D. C. Moore, "Lcm: Lightweight communications and marshalling," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Octubre 2010, pp. 4057–4062.

[41] Rabbitmq. [Online]. Available: http://www.rabbitmq.com/

[42] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[43] Zeromq. [Online]. Available: http://zeromq.org/

[44] Multiprocess communications library. [Online]. Available: https://github.com/acfr/mcl

[45] R. Szeliski, *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[46] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[47] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346 – 359, 2008, similarity Matching in Computer Vision and Multimedia.

[48] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, ser. IJCAI'81. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1981, pp. 674–679.

[49] B. Micusik and J. Kosecka, "Piecewise planar city 3d modeling from street view panoramic sequences," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, Junio 2009, pp. 2906–2912.

[50] S. P. Singh, K. Jain, and V. R. Mandla, "Virtual 3d City Modeling: Techniques and Applications," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, no. 2, pp. 73–91, Agosto 2013.

[51] D. Troiano, A. Morro, A. Merlo, and E. Vidal, "From a model of a city to an urban information system: The siur 3d of the castle of pietrabuona," in *Digital Heritage. Progress in Cultural Heritage: Documentation, Preservation, and Protection*, ser. Lecture Notes in Computer Science, M. Ioannides,

N. Magnenat-Thalmann, E. Fink, R. Zarnic, A.-Y. Yen, and E. Quak, Eds. Springer International Publishing, 2014, vol. 8740, pp. 121–130.

[52] K. Ozden, K. Schindler, and L. Van Gool, "Multibody structure-from-motion in practice," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 6, pp. 1134–1141, Junio 2010.

[53] Openstreetmap. [Online]. Available: http://www.openstreetmap.org/

[54] Google maps. [Online]. Available: http://maps.google.com/

[55] J. Bijl and C. Boer, "Advanced 3d visualization for simulation using game technology," in *Simulation Conference (WSC), Proceedings of the 2011 Winter*, Diciembre 2011, pp. 2810–2821.

[56] Simulator for intelligent transportation systems. [Online]. Available: https://youtu.be/qhAyg5tlX0c

[57] M. Liebner, F. Klanner, and C. Stiller, "Active safety for vulnerable road users based on smartphone position data," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, Junio 2013, pp. 256–261.

[58] H. Kimm and H. Kimm, "Improved track path method in real time by using gps and accelerometer," in *2016 15th International Conference on Ubiquitous Computing and Communications and 2016 International Symposium on Cyberspace and Security (IUCC-CSS)*, Diciembre 2016, pp. 77–84.

[59] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *SIGKDD Explor. Newsl.*, vol. 12, no. 2, pp. 74–82, Marzo 2011. [Online]. Available: http://doi.acm.org/10.1145/1964897.1964918

[60] M. A. Ayu, S. A. Ismail, T. Mantoro, and A. F. A. Matin, "Real-time activity recognition in mobile phones based on its accelerometer data," in *2016 International Conference on Informatics and Computing (ICIC)*, Octubre 2016, pp. 292–297.

[61] S.-M. Lee, S. M. Yoon, and H. Cho, "Human activity recognition from accelerometer data using convolutional neural network," in *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Febrero 2017, pp. 131–134.

[62] A. Chowdhury, D. Tjondronegoro, V. Chandran, and S. Trost, "Physical activity recognition using posterior-adapted class-based fusion of multiaccelerometers data," *IEEE Journal of Biomedical and Health Informatics*, vol. PP, no. 99, pp. 1–1, 2017.

[63] L. Bao and S. S. Intille, *Activity Recognition from User-Annotated Acceleration Data*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 1–17.

[64] A. Bujari, B. Licar, and C. E. Palazzi, "Movement pattern recognition through smartphone's accelerometer," in *2012 IEEE Consumer Communications and Networking Conference (CCNC)*, Enero 2012, pp. 502–506.

[65] R. Akhavian and A. H. Behzadan, "Construction activity recognition for simulation input modeling using machine learning classifiers," in *Proceedings of the Winter Simulation Conference 2014*, Dec 2014, pp. 3296–3307.

[66] T. K. Ho, *Nearest neighbors in random subspaces*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 640–648. [Online]. Available: https://doi.org/10.1007/BFb0033288

[67] B. Volz, K. Behrendt, H. Mielenz, I. Gilitschenski, R. Siegwart, and J. Nieto, "A data-driven approach for pedestrian intention estimation," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Noviembre 2016, pp. 2607–2612.

[68] C. Benedek, B. Gálai, B. Nagy, and Z. Jankó, "Lidar-based gait analysis and activity recognition in a 4d surveillance system," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 1, pp. 101–113, Enero 2018.

[69] H. Kataoka, Y. Aoki, Y. Satoh, S. Oikawa, and Y. Matsui, "Fine-grained walking activity recognition via driving recorder dataset," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, Septiembre 2015, pp. 620–625.

[70] N. Abhayasinghe and I. Murray, "Human activity recognition using thigh angle derived from single thigh mounted imu data," in *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Octubre 2014, pp. 111–115.

[71] J. B. Bancroft, D. Garrett, and G. Lachapelle, "Activity and environment classification using foot mounted navigation sensors," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Noviembre 2012, pp. 1–10.

[72] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys, "Accurate 3d pose estimation from a single depth image," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 731–738.

[73] R. Quintero, J. Almeida, D. Llorca, and M. Sotelo, "Pedestrian path prediction using body language traits," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, Junio 2014, pp. 317–323.

[74] M. Andriluka, S. Roth, and B. Schiele, "Monocular 3d pose estimation and tracking by detection," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 623–630.

[75] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[76] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," in *CVPR*, 2017.

[77] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *CVPR*, 2016.

[78] W. Zhou, R. Arroyo, A. Zyner, J. Ward, S. Worrall, E. Nebot, and L. Bergasa, "Transferring visual knowledge for a robust road environment perception

in intelligent vehicles," in *2017 IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2017.

[79] Filtering pedestrian skeletons on images using semantics. [Online]. Available: https://youtu.be/BRcOneJ3vn0

[80] T. N. Do, R. Liu, C. Yuen, and U. X. Tan, "Design of an infrastructureless indoor localization device using an imu sensor," in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Diciembre 2015, pp. 2115–2120.

[81] B. Beaufils, F. Chazal, M. Grelet, and B. Michel, "Stride detection for pedestrian trajectory reconstruction: A machine learning approach based on geometric patterns," in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Septiembre 2017, pp. 1–6.

[82] D. Ridel, E. Rehder, M. Lauer, C. Stiller, and D. Wolf, "A literature review on the prediction of pedestrian behavior in urban scenarios," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2018, pp. 3105–3112.

[83] O. Ghori, R. Mackowiak, M. Bautista, N. Beuter, L. Drumond, F. Diego, and B. Ommer, "Learning to forecast pedestrian intention from pose dynamics," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1277–1284.

[84] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, March 1994.

[85] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[86] Set of videos with a 3-seconds window. [Online]. Available: https://youtu.be/f3_qPjZR0Ls

[87] Set of videos with a 2-seconds window. [Online]. Available: https://youtu.be/A7-litYH6XQ

[88] Demonstration - pedestrian intention using lstm neural network. [Online]. Available: https://youtu.be/FKzZs6PT07k

Anexo A

Redes LSTM

A.1. La idea central detrás de las redes LSTM

La clave de estas redes son las celdas (o unidades) de memoria que pueden conservar información pasada. Este comportamiento es modelado por la línea horizontal que corre sobre la parte superior del diagrama de la Figura A.1. El estado de la celda es similar a una cinta transportadora que recorre toda la cadena con sólo algunas interacciones lineales menores que facilita el flujo de información sin cambios.

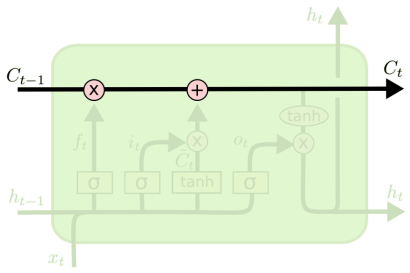


Figura A.1: Representación de la celda enfocada en el flujo de información principal C.

La red LSTM tiene la habilidad de remover o agregar información al estado de la celda, regulado cuidadosamente por estructuras llamadas "compuertas" (Figura A.2). Las compuertas son un camino opcional para proveer información y están compuestas de una función sigmoide y un punto de operación de multiplicación.

Los valores de salida de la función sigmoide son entre 0 y 1, describiendo cuanto de cada componente debe alimentar a la línea principal. Un valor 0 significa que nada debe ser aportado, mientras que un valor 1 significa que toda la

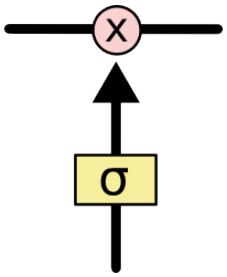
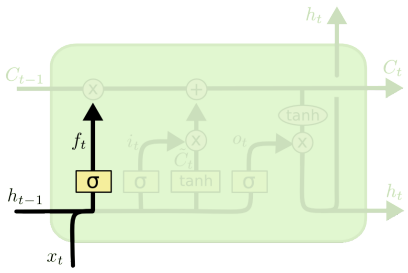


Figura A.2: El círculo rojo claro representa un punto de operación, en este caso una multiplicación, y el rectángulo amarillo representa una compuerta, que en este caso es una función sigmoide.

información debe ser incorporada. Una red LSTM tiene tres de estas compuertas para proteger y controlar el estado de la celda.

A.2. El camino a través de la red

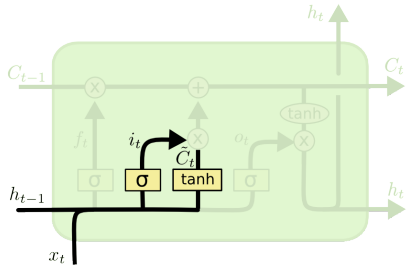
El primer paso es decidir que información fluirá a través de la celda. Esta decisión es llevada a cabo por la función sigmoide denominada "compuerta de olvido"(Figura A.3), cuyas entradas son h_{t-1} y x_t y la salida es un número entre 0 y 1 para cada valor del estado de la celda C_{t-1}. Un valor 1 significa que toda la información debe recordarse, mientras que un valor 0 significa que esa información debe olvidarse.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Figura A.3: Representación de la celda enfocada en la "compuerta de olvido", la cual decide la información que no es relevante y debe olvidarse.

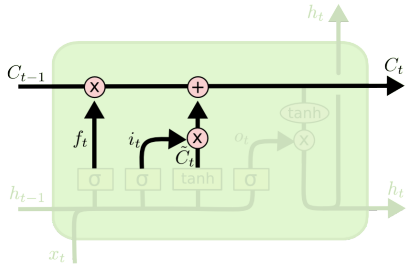
El siguiente paso es decidir que información nueva se almacenará en la celda. Esto tiene dos partes, la primera es por medio de una función sigmoide llamada "compuerta de entrada"(Figura A.4) que decide cual valor será actualizado. Luego, una función *tanh* crea un vector de nuevos valores candidatos, $\tilde{C}_t$, para ser agregado en el estado. Por último, ambos se combinarán para crear una actualización.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Figura A.4: Representación de la celda enfocada en la "compuerta de entrada", la cual decide qué información se utilizará para actualizar el estado actual.

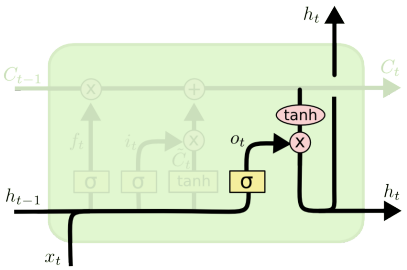
Ahora es momento de actualizar el estado anterior de la celda, C_{t-1}, en el nuevo estado, C_t (Figura A.5). En los pasos previos ya se decidió que información resulta relevante y cual no, por lo que ahora sólo resta hacer la actualización. Para ello se multiplica el estado anterior por f_t, olvidando los valores que se han decidido olvidar, y luego se agrega $i_t * \tilde{C}_t$. Esto resulta en un nuevo valor candidato, escalado por el valor que se decidió para actualizar cada valor del estado.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figura A.5: Los valores candidatos son utilizados para actualizar el estado de la celda.

Finalmente, se necesita decidir qué irá a la salida (Figura A.6), la cual estará basada en el estado de la celda y será una versión filtrada. Primero se aplica una función sigmoide, la cual decide que parte del estado de la celda irá a la salida. Luego, se utiliza el estado de la celda actual, aplicado a una *tanh* (para ponerlo entre valores de -1 y 1) y se lo multiplica por la salida de la función sigmoide, que sólo deja pasar las partes que se han decidido.



$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Figura A.6: La última parte de la celda decide qué valores dispondrá en la salida para que sean utilizado en la siguiente celda.