



Universidad  
de Alcalá

*Campus Universitario  
Dpto. de Teoría de la Señal y Comunicaciones  
Ctra. Madrid-Barcelona, Km. 36,6  
28871 Alcalá de Henares (Madrid)  
Telf: +34 91 885 88 99  
Fax: +34 91 885 66 99*

D. SATURNINO MALDONADO BASCÓN, Catedrático de Universidad del Área de Conocimiento de Teoría de la Señal y Comunicaciones de la Universidad de Alcalá,

#### CERTIFICA

Que la tesis doctoral titulada "**Sistema de detección y reconocimiento de señalización en carretera mediante técnicas de procesado digital de imagen e inteligencia artificial**", presentada por D. Sergio Lafuente Arroyo y realizada en el Departamento de Teoría de la Señal y Comunicaciones bajo mi dirección, reúne méritos suficientes para optar al grado de Doctor, por lo que puede procederse a su depósito y lectura.

Alcalá de Henares, 23 de septiembre de 2014.

Fdo: Dr. D. Saturnino Maldonado Bascón





Universidad  
de Alcalá

*Campus Universitario  
Dpto. de Teoría de la Señal y Comunicaciones  
Ctra. Madrid-Barcelona, Km. 36,6  
28871 Alcalá de Henares (Madrid)  
Telf: +34 91 885 88 99  
Fax: +34 91 885 66 99*

D. Sergio Lafuente Arroyo ha realizado en el Departamento de Teoría de la Señal y Comunicaciones y bajo la dirección del Dr. Saturnino Maldonado Bascón, la tesis doctoral titulada "**Sistema de detección y reconocimiento de señalización en carretera mediante técnicas de procesado digital de imagen e inteligencia artificial**", cumpliéndose todos los requisitos para la tramitación que conduce a su posterior lectura.

Alcalá de Henares, 23 de septiembre de 2014.

EL DIRECTOR DEL DEPARTAMENTO

Fdo: Dr. D. Saturnino Maldonado Bascón





Universidad  
de Alcalá

ESCUELA POLITÉCNICA SUPERIOR

Tesis Doctoral

**SISTEMA DE DETECCIÓN Y  
RECONOCIMIENTO DE SEÑALIZACIÓN EN  
CARRETERA MEDIANTE TÉCNICAS DE  
PROCESADO DIGITAL DE IMAGEN E  
INTELIGENCIA ARTIFICIAL**

Autor: Sergio Lafuente Arroyo

Director: Dr. Saturnino Maldonado Bascón

23 de septiembre de 2014



# Agradecimientos

A mis compañeros del Departamento de Teoría de la Señal y Comunicaciones, especialmente a los miembros del grupo de investigación GRAM. Mi agradecimiento al Dr. Saturnino Maldonado por generar el germen de esta tesis doctoral y realizar el seguimiento del camino hasta llegar al final y al Dr. Hilario Gómez por su disposición y ayuda en innumerables aspectos técnicos. También quiero expresar mi gratitud al Dr. Francisco López Ferreras y al Dr. Manuel Rosa Zurera por su confianza como directores del Departamento de Teoría de la Señal y Comunicaciones desde mi incorporación al mismo.

A mis padres por su apoyo y por haber convertido su vida en una lección de integridad, coraje y entrega. A mi hermana y cuñado. A mis sobrinos (Natalia y David) por su facilidad para llenar de alegría e ilusión los ratos compartidos.

A Mirian por su capacidad para pintar de luz los días más hostiles y su infinita paciencia para saber esperar.



# Resumen

Esta tesis doctoral se enmarca dentro de los Sistemas Inteligentes de Transporte con el objetivo de desarrollar un sistema preciso y eficiente para el reconocimiento de señales de tráfico, basándose en técnicas de procesamiento digital de imagen e inteligencia artificial. Desde hace más de una década, son numerosos los grupos de investigación en el ámbito internacional que han trabajado en este campo intentando resolver los problemas derivados de la detección de objetos en entornos exteriores. En cualquier caso, la mayoría de los trabajos de investigación se han ocupado de etapas concretas del sistema y han trabajado sobre conjuntos de señales reducidos. Por ello, la primera línea de trabajo de esta tesis centra la atención en el estudio de técnicas para un sistema completo de detección e identificación de señalización que puede ser utilizado tanto para el inventariado automático de carreteras como para el desarrollo de vehículos inteligentes. El sistema se basa en una estructura modular correspondiente a las etapas de segmentación, clasificación de forma, reconocimiento del pictograma y seguimiento. Aunque en esta tesis se ha trabajado principalmente con la señalización vial de España, el sistema es fácilmente configurable para la señalización de otro país.

La buena capacidad de generalización de las SVMs ha llevado a introducir esta herramienta estadística en el campo del reconocimiento de señales de tráfico y en esta tesis se incorpora su uso a las etapas de clasificación de forma y de reconocimiento. Los buenos resultados conseguidos implican, no obstante, un elevado coste computacional que supone un ‘cuello de botella’, que viene dado por el alto número de categorías presentes en el problema de multclasificación. Por este motivo, se realiza una investigación en la búsqueda de las mejores técnicas para el módulo de reconocimiento. Esta línea de trabajo conlleva un estudio comparativo de diferentes técnicas paramétricas desde el compromiso de la precisión y la optimización de carga computacional para las tareas de preprocesado, extracción del descriptor y clasificación. Asimismo, se han comparado las principales arquitecturas de multclasificación para la tarea de reconocimiento.

Con el objetivo de reducir la complejidad y lograr que el módulo de reconocimiento basado en SVMs opere en tiempo real, se propone un algoritmo de búsqueda de regiones de información con el doble objetivo de agrupar las señales con similar distribución espacial del pictograma y definir en cada cluster un descriptor adaptativo diferente. El agrupamiento de señales consigue reducir el número de clases implicadas en el problema de multclasificación para cada vector de entrada y por otra parte, el descriptor adapta-

tivo en cada cluster asigna a cada región un número de características proporcional a la cantidad de información contenida. De esta manera, es posible controlar el coste computacional en la etapa de reconocimiento sin que ello suponga una pérdida apreciable de las prestaciones del sistema.

Las propuestas que se describen en esta tesis doctoral requieren disponer de una base de datos para el desarrollo de la fase experimental del trabajo y para el proceso de entrenamiento de los algoritmos de aprendizaje supervisados. Debido a que no existe un conjunto estándar de señalización público, se ha creado una base de datos que incluye la totalidad de señales verticales existentes en España. Con el objetivo de abordar situaciones críticas, las capturas se han realizado en diferentes escenarios con variación de la configuración de las video-cámaras del sistema de adquisición. Los resultados obtenidos avalan la viabilidad y competencia de los métodos propuestos en esta tesis doctoral en el campo del reconocimiento automático de señales de tráfico.

# Abstract

This Ph.D. thesis finds itself in the context of Intelligent Transportation Systems and its main objective is to carry out a study about different techniques in the field of digital image processing and computer vision applied to automatic traffic sign recognition. Even when multiple number research groups dedicated to traffic sign detection had arisen in the last decade, most of them have been concerned with an only stage and their experiments have been developed with reduced sets. The first research line of the thesis focuses on the study and development of a complete traffic sign detection and recognition system, which can be used for road maintenances and for intelligent vehicles. The system is based on a modular structure with four stages that correspond to segmentation, shape classification, ideogram recognition and tracking. Although the dataset we built in this thesis contains all Spanish categories, the system is easily adaptable to the signs of another countries.

The good properties of generalization exhibited by SVMs lead us to introduce them as a statistic alternative, specifically in the shape classification and ideogram recognition stages. However, the process of recognition with SVMs is slow since the system must manage many classes and requires a lot of training data with different image conditions. It demands a high computational complexity and supposes a bottleneck in the complete system. The ideogram recognition task is structured in three steps: preprocessing, descriptor extraction and classification. The second research line in this thesis focuses on the optimization of the ideogram recognition by making a comparative study between existing parametric algorithms. Thus, we search the best architectures and descriptors from the points of view of computational cost and overall performance.

With the aim of reducing the complexity at the recognition stage, we have proposed a novel technique based on the search of spatial distribution of ideograms by finding common similarities that can be shared across the classes. The procedure employed in this work allows us to cluster categories with similar spatial distribution. In this way, the number of potential classes is reduced in the sense that classification process only takes into account the classes grouped within the selected cluster. In addition, an adaptive descriptor is extracted independently in each cluster, where the number of features of each region depends on the information measurement. Thus, it is possible to control the computational cost at the recognition stage while keeping the overall performance of the system. Our experimental results show substantial time reduction with respect to the conventional procedure when all classes are considered in the multi-classification problem.

The proposals described in this Ph.D. thesis have been assessed for a Spanish traffic sign dataset. This has facilitated the development of the experimental stage, while achieving significant results that support the viability of the use of the methods presented. As there does not exist a public standard dataset, we have created one with all the Spanish traffic signs. In order to include scenes and samples under critical situations, the sequences have been acquired by varying the configuration of cameras in different scenarios.

# Glosario de abreviaturas y acrónimos

**ADAS** Advanced Driver Assistance System

**CM** Cluster map

**CNN** Convolutional Neural Network

**CSIC** Consejo Superior de Investigaciones Científicas

**DDAG** Decision directed acyclic graphs

**DECOC** Discriminant Error Correcting Output Codes

**DFT** Discrete Fourier Transform

**DR** Detection Rate

**DTB** Distance to Borders

**ECOC** Error Correcting Output Code

**ERM** Empirical Risk Minimization

**FPR** False Positive Rate

**GA** Genetic Algorithm

**HSI** Hue-Saturation-Intensity Space

**HOG** Histogram of Oriented Gradient

**IM** Ideogram Map

**ITS** Intelligent Transportation Systems

**LBP** Local Binary Pattern

**LESH** Local Energy Based Shape Histogram

**LUT** Look Up Tables

**NN** Neural Network

**OVA** One versus All

**OVO** One versus one

**PCA** Principal Component Analysis

**POI** Pixels of Interest

**RBF** Radial Base Function

**RIFT** Rotation-Invariant Feature Transform

**ROI** Region of Interest

**SA** Simulated Annealing

**SIFT** Scale Invariant Feature Transform

**SMO** Sequential Minimal Optimisation

**SRM** Structural Risk Minimization

**SVF** Simple Vector Filter

**SVM** Supported Vector Machines

**TPR** True Positive Rate

**TSDRS** Traffic Sign Detection and Recognition System

# Índice general

<b>Resumen</b>	<b>V</b>
<b>Abstract</b>	<b>VII</b>
<b>Glosario de abreviaturas y acrónimos</b>	<b>IX</b>
<b>Índice general</b>	<b>XI</b>
<b>Lista de figuras</b>	<b>XV</b>
<b>Lista de tablas</b>	<b>XXIII</b>
<b>1 Introducción</b>	<b>1</b>
1.1 Sistemas ITS . . . . .	3
1.1.1 Clasificación . . . . .	4
1.1.2 Arquitectura . . . . .	7
1.2 Requisitos de los sistemas ITS . . . . .	7
1.3 Métrica de evaluación . . . . .	8
1.4 Descripción de un sistema ITS basado en visión artificial . . . . .	10
1.5 Motivación de la tesis . . . . .	11
1.5.1 Objetivos de la tesis . . . . .	12
1.5.2 Estructura de la tesis . . . . .	13
<b>2 Detección de Señales de Tráfico: Estado del Arte y Perspectivas</b>	<b>15</b>
2.1 Características principales . . . . .	15
2.1.1 Reglamentación internacional . . . . .	16

2.1.2	Reglamentación nacional . . . . .	18
	Señales de advertencia de peligro . . . . .	18
	Señales de reglamentación . . . . .	19
2.1.3	Señales de indicación . . . . .	19
2.2	Descripción del Entorno de Trabajo . . . . .	20
2.3	Revisión de la literatura . . . . .	23
2.3.1	Detección de Señales . . . . .	25
	Segmentación por color . . . . .	25
	Segmentación por bordes . . . . .	30
	Segmentación por bordes y color . . . . .	32
	Otras técnicas: . . . . .	32
2.3.2	Clasificación de formas . . . . .	33
2.3.3	Etapas de Reconocimiento de Señales . . . . .	35
	Descriptores . . . . .	35
	Clasificadores . . . . .	36
2.4	Sistemas Comerciales . . . . .	42
2.5	Resumen . . . . .	44
<b>3</b>	<b>Bases de Datos de Señalización Vertical</b>	<b>47</b>
3.1	Bases de datos existentes . . . . .	47
3.2	Características de una nueva base de datos . . . . .	49
3.2.1	Especificaciones . . . . .	49
3.2.2	Descripción . . . . .	51
<b>4</b>	<b>Sistema para Detección y Reconocimiento de Señalización Vertical</b>	<b>57</b>
4.1	Módulo de captura . . . . .	58
4.2	Módulo de Segmentación . . . . .	59
4.2.1	Variación de color en entornos exteriores . . . . .	60
4.2.2	Invarianza del Tono . . . . .	63
4.2.3	Segmentación basada en HSI . . . . .	63
4.3	Módulo de Clasificación de Forma . . . . .	72
4.4	Módulo de Reconocimiento . . . . .	74
4.4.1	Vector descriptor . . . . .	75
4.4.2	Clasificador . . . . .	77
4.5	Resultados del sistema . . . . .	87
4.6	Módulo de seguimiento . . . . .	91
4.6.1	Seguimiento intra-imagen . . . . .	91
4.6.2	Seguimiento inter-imagen . . . . .	92
	Algoritmo de tracking . . . . .	96
4.7	Análisis de la carga computacional . . . . .	100
4.7.1	Resumen . . . . .	102

<b>5</b>	<b>Arquitecturas de reconocimiento basadas en SVM</b>	<b>105</b>
5.1	Técnicas de preprocesado . . . . .	105
5.2	Extracción del descriptor . . . . .	108
5.3	Análisis comparativo de técnicas de preprocesado y descriptores . . . . .	113
5.3.1	Estudio paramétrico de los descriptores elegidos . . . . .	119
5.4	Arquitecturas de multclasificación basadas en SVM . . . . .	124
5.4.1	Método ‘Uno contra todos’ . . . . .	125
5.4.2	Método ‘Uno contra uno’ . . . . .	125
5.4.3	Códigos correctores de Error . . . . .	128
	Codificación ECOC . . . . .	129
	Decodificación ECOC . . . . .	133
5.4.4	Metodos de evaluación . . . . .	136
5.4.5	Comparativa de clasificadores . . . . .	138
5.4.6	Análisis de fiabilidad y carga computacional . . . . .	139
5.4.7	Análisis de técnicas de decodificación . . . . .	142
5.4.8	Análisis de significancia estadística . . . . .	143
5.5	Resumen . . . . .	144
<b>6</b>	<b>Reconocimiento de señales basado en mapas de pictogramas</b>	<b>147</b>
6.1	Mapa del pictograma . . . . .	147
6.1.1	Proceso de extracción . . . . .	148
6.1.2	Agrupamiento de clases . . . . .	149
6.2	Descripción del algoritmo . . . . .	151
6.2.1	Selección de clusters . . . . .	151
6.2.2	Reconocimiento del ideograma . . . . .	157
6.3	Resultados . . . . .	159
6.3.1	Influencia del número de clusters . . . . .	159
6.3.2	Influencia del tamaño de la rejilla . . . . .	162
6.3.3	Comparativa de métodos . . . . .	165
6.4	Generalización . . . . .	165
6.5	Resumen . . . . .	166
<b>7</b>	<b>Contribuciones y futuras líneas de investigación</b>	<b>169</b>
7.1	Contribuciones originales . . . . .	169
7.1.1	Sistema para Detección y Reconocimiento de Señalización Vertical	169
7.1.2	Arquitecturas de reconocimiento basadas en SVM . . . . .	170
7.1.3	Reconocimiento de señales basado en mapas de pictogramas . . . . .	170
7.2	Futuras líneas de investigación . . . . .	171

<b>A</b>	<b>Publicaciones a las que ha dado lugar la tesis</b>	<b>177</b>
A.1	Revistas . . . . .	177
A.2	Congresos internacionales . . . . .	178
A.3	Congresos nacionales . . . . .	179
A.4	Menciones . . . . .	179
<b>B</b>	<b>Máquinas de Vectores Soporte</b>	<b>181</b>
B.1	Nociones sobre aprendizaje estadístico . . . . .	181
B.2	Clasificación mediante SVM . . . . .	182
B.2.1	Datos separables linealmente . . . . .	183
B.2.2	Datos no separables . . . . .	187
B.2.3	Datos no separables linealmente . . . . .	189
B.2.4	Kernels . . . . .	190
<b>C</b>	<b>Catálogo Oficial de Señales de Tráfico de España</b>	<b>193</b>
	<b>Bibliografía</b>	<b>207</b>

# Índice de figuras

1.1	Ejemplo de curva ROC. . . . .	11
1.2	Esquema de bloques general de un sistema ITS basado en visión artificial. . . . .	12
2.1	Ejemplos de señales americanas con información textual. . . . .	16
2.2	Ejemplos de señales con textos informativos en varios idiomas. . . . .	17
2.3	Variantes europeas del pictograma de una misma señal. . . . .	18
2.4	Cambios de iluminación en entornos viales. (a) Día. (b) Puesta de sol. (c) Anochecer. (d) Noche. . . . .	21
2.5	Brillos y sombras producidos sobre las señales. (a) Sombra proyectada. (b) Puntos de brillo. . . . .	21
2.6	Señales con diferentes niveles de oclusión. . . . .	22
2.7	Señales con deformaciones proyectivas. (a) Rotación 2D. (b) Rotación 3D. . . . .	23
2.8	Señales con diferentes causas de deterioro. (a) Envejecimiento. (b) Deformación del material. (c) <i>Graffiti</i> . . . . .	24
2.9	Señales con colores similares a los de los objetos adyacentes. . . . .	25
2.10	Clasificación de color mediante LUT en [Escalera03]. . . . .	28
2.11	Imagen binaria obtenida en [Aoyagi96] tras aplicar un filtro laplaciano a la imagen para la obtención de bordes. . . . .	30
2.12	Ejemplos de formas ideales para comparar con las formas reales en [Adorni96]. Las formas originales (a1 y b1) se amplían para otras escalas añadiendo píxeles internos en (a2, b2, a3 y b3). . . . .	31
2.13	Simetría radial utilizada en [Loy04]. (a) Para el borde de un punto, los centroides de una posible familia de círculos se extienden a lo largo de la línea que es ortogonal a la dirección del borde. (b) Cada punto del borde de un círculo vota a lo largo de una línea de posibles radios y estas líneas intersectan en el centro del círculo, resultando un pico. . . . .	31

2.14	Ejemplo de búsqueda de señales triangulares mediante GA en [Escalera03]. (a) Imagen Original. (b) Función de ajuste para una señal triangular. (c) Mapa de elevación de (b). (d) Mapa detallado de los máximos. . . . .	32
2.15	Histograma promediado de orientaciones para señales circulares y rectangulares en [Gao06]. Las señales rectangulares muestran predominio de orientaciones verticales y horizontales y las circulares, una distribución más uniforme en todas las orientaciones. . . . .	34
2.16	Máscara para detección de señales circulares en [Escalera97]. . . . .	34
2.17	Relación entre el espacio complejo y el espacio complejo de la imagen definido en [Kang94]. . . . .	37
2.18	Regiones locales discriminatorias para 5 clases en [Paclik06]. . . . .	37
2.19	Regiones discriminatorias con diferentes umbrales en [Ruta09]. . . . .	38
2.20	Arquitectura de una red neuronal convolucional en [Ciresan11], donde las capas de convolución están completamente conectadas. Ambas capas de convolución usan kernels $5 \times 5$ . . . . .	42
2.21	Diseño ECOC para cuatro clases en [Baro09]. (a): Codificación ECOC ‘uno contra todos’. (b): Codificación ECOC ‘uno contra uno’. (blanco:+1, negro:-1 y gris:0). . . . .	43
3.1	Muestras de la base de datos <i>German Traffic Sign Benchmark</i> (GTSRB). . . . .	49
3.2	Muestra de la base de datos de la Universidad de <i>Linköping</i> (STS Data set). . . . .	49
3.3	Muestras de la base de datos <i>KUL Belgium Traffic Sign Dataset</i> . . . . .	50
3.4	Muestra de la base de datos RUG Data set. . . . .	50
3.5	Estructura de directorios por color y forma en la base de datos. . . . .	52
4.1	Arquitectura modular del sistema. . . . .	58
4.2	Vehículo con sistema de captura a bordo. (a) Vehículo de prueba. (b) Sistema de cámaras. (c) Odómetro para medida de distancias. (d) Sistema interno de adquisición. . . . .	59
4.3	Construcción de una escena panorámica a partir de las imágenes captadas por las cámaras. (a) Imagen izquierda. (b) Imagen central. (c) Imagen derecha. (d) Imagen resultante del proceso de construcción. . . . .	61
4.4	Modelo de una escena de tráfico. . . . .	61
4.5	Curva CIE de luz diurna. . . . .	62
4.6	Histogramas H-S en visibilidad diurna. (a) Canal rojo. (b) Canal azul. (c) Canal amarillo. . . . .	65
4.7	Ejemplos de segmentación. De arriba hacia abajo, imágenes originales, máscaras de segmentación del canal rojo, máscaras de segmentación del canal azul, máscaras de segmentación del canal amarillo y máscaras de segmentación del canal acromático. . . . .	67

4.8	Histogramas H-S en visibilidad nocturna. (a). Canal rojo. (b) Canal azul. (c) Canal acromático. . . . .	68
4.9	Histogramas H-S en visibilidad nocturna. (a) Imagen original. (b) Canal rojo. (c) Canal azul. (d) Canal amarillo. (e) Canal acromático. . . . .	69
4.10	Ejemplos de segmentación tras aplicar un filtro de tamaño y relación de aspecto. (a, b, c, d). De arriba hacia abajo, imágenes originales, máscaras de segmentación del canal rojo, máscaras de segmentación del canal azul, máscaras de segmentación del canal amarillo y máscaras de segmentación del canal acromático. . . . .	70
4.11	Distancia a bordes de una forma triangular. . . . .	73
4.12	Invarianza a la traslación. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d). . . . .	74
4.13	Invarianza a la escala. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d). . . . .	75
4.14	Invarianza a la rotación 2D. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d). . . . .	76
4.15	Invarianza a la rotación 3D. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d). . . . .	77
4.16	Ejemplos de clasificación de los objetos detectados en 4.10. . . . .	78
4.17	Firma y módulo de la FFT para un círculo. . . . .	78
4.18	Firma y módulo de la FFT para un triángulo equilátero. . . . .	79
4.19	Firma y módulo de la FFT para un cuadrado. . . . .	79
4.20	Máscaras para eliminación de fondo. (a) Máscara circular. (b) Máscara triangular . . . . .	79
4.21	Obtención del vector descriptor. (a) Blob en color. (b) Conversión del blob a nivel de gris. (c) Eliminación de los píxeles del fondo. (d) Vector descriptor. . . . .	80
4.22	Ejemplos de muestras. (a) Señales. (b) Ruidos. . . . .	80
4.23	Diagrama funcional del sistema completo con un ejemplo de detección y reconocimiento. . . . .	81
4.24	Vectores soporte de la clase ‘Limitación de velocidad a 40 Km/h’. (a) Vectores soporte positivos. (b) Vectores soporte negativos. . . . .	83
4.25	Análisis de la influencia del número de vectores soporte y la dimensión en el tiempo de evaluación de las SVM. (a) Tiempo de testeo en función del número de vectores soporte. (b) Tiempo de testeo en función de la dimensión de los vectores. . . . .	84
4.26	Ejemplo del ajuste de la probabilidad de salida para la clase ‘Incorporación por la derecha’ según el método de Platt[Platt00]. (a) Histogramas de la probabilidad a priori. (b) Probabilidad a posteriori de las muestras positivas. (c) Ajuste de la sigmoide. . . . .	86

4.27	Familia de sigmoides de ajuste para las 10 primeras clases del conjunto de señales rojas triangulares. . . . .	87
4.28	Imágenes de resultados de la secuencia $S_1$ . . . . .	89
4.29	Imágenes de resultados de la secuencia $S_2$ . . . . .	89
4.30	Imágenes de resultados de la secuencia $S_4$ con pérdidas en (b,c) y confusión en (e). . . . .	90
4.31	Imágenes de resultados de la secuencia nocturna $S_5$ . . . . .	90
4.32	Tamaños de la máscara de oclusión con tres diámetros. De izquierda a derecha, diámetros de 1/2, 1/3 and 1/4, respectivamente, de la mayor dimensión del bounding box. . . . .	90
4.33	Desplazamientos de la máscara para la orientación de $0^\circ$ . . . . .	91
4.34	Resultados obtenidos mediante el tracking intra-imagen. . . . .	92
4.35	Ejemplo de oclusión temporal parcial en una video-secuencia. . . . .	93
4.36	Modelo de cámara <i>pin-hole</i> para el cálculo de la máxima distancia de tracking. . . . .	94
4.37	Predicción de posición con un modelo de aceleración constante (predicción 1) y con un modelo de velocidad constante (predicción 2). . . . .	98
4.38	Ejemplo de tracking con oclusión temporal en una secuencia. . . . .	100
4.39	Ejemplo de ‘desenganche’ del tracking debido a la curvatura del recorrido. . . . .	101
4.40	(a) y (b) Detecciones realizadas con diferentes métodos de segmentación. (c) Resultado a la salida del módulo de reconocimiento. . . . .	102
4.41	Características del sistema RESET. De arriba a abajo, deformaciones proyectivas, cambios de iluminación, oclusiones parciales, arrays y señales viales estadounidenses. . . . .	104
5.1	Diagrama de bloques del sistema de reconocimiento. . . . .	105
5.2	Ejemplos de pre-procesado sobre un conjunto de muestras. (a) Conjunto original de muestras en color. (b) Conjunto original en escala de gris tras aplicar la máscara de eliminación de fondo. (c) Muestras tras aplicar normalización del nivel de gris con $K = 120$ . (d) Muestras tras aplicar la técnica de contrast-stretching con $T_{cs1}=0.02$ y $T_{cs2}=0.98$ . (e) Muestras con ecualización de histograma empleando $N_{eq} = 256$ . (f) Muestras tras aplicar la umbralización de Otsu. . . . .	107
5.3	Perfil de transformación Contrast-Stretching. . . . .	108
5.4	Ejemplo de cálculo del gradiente. (a) Muestra original en escala de gris. (b) Mapa de orientación del gradiente de la imagen original. (c) Mapa de orientación del gradiente en la celda seleccionada. (d) Histograma de orientaciones. . . . .	110
5.5	Construcción del descriptor SPIN. Dos puntos son mapeados de la ROI normalizada (izquierda) a las localizaciones del mapa del descriptor (derecho). . . . .	111

5.6	Construcción del descriptor RIFT. Dos puntos son mapeados de la ROI normalizada (izquierda) a las localizaciones del mapa del descriptor (derecha). . . . .	111
5.7	Operador LBP. . . . .	112
5.8	LBP con diferentes escalas variando el radio y los puntos de muestreo. . . . .	112
5.9	Proceso de extracción del descriptor LBP con una partición de la muestra en una rejilla de $3 \times 3$ celdas. . . . .	113
5.10	Curvas ROC para análisis de técnicas de preprocesado con diferentes descriptores utilizando señales de prohibición. (a) Barrido de la intensidad de los píxeles. (b) HOG. (c) RIFT. (d) SPIN. (e) LBP. . . . .	117
5.11	Tiempo de cómputo para análisis de técnicas de preprocesado con diferentes descriptores utilizando señales de prohibición. (a) Barrido de la intensidad de los píxeles. (b) HOG. (c) SPIN. (d) RIFT. (e) LBP. . . . .	118
5.12	Tiempo de procesado por muestra en función de los parámetros $N_b$ y $N_w$ . (a) HOG. (b) LBP. . . . .	123
5.13	Comparativa entre los descriptores HOG y LBP. . . . .	124
5.14	Ejemplo de clasificación OVA con $M = 5$ clases. a) Representación de las 5 clases en un espacio bidimensional. b) Clase 1 contra el resto: $y_1 = +1; y_{2,3,4,5} = -1$ . c) Clase 2 contra el resto: $y_2 = +1; y_{1,3,4,5} = -1$ . d) Clase 3 contra el resto: $y_3 = +1; y_{1,2,4,5} = -1$ . e) Clase 4 contra el resto: $y_4 = +1; y_{1,2,3,5} = -1$ . f) Clase 5 contra el resto: $y_5 = +1; y_{1,2,3,4} = -1$ . . . . .	126
5.15	Ejemplo de clasificación 1-vs-1 con $M = 3$ clases. a) Representación de las 3 clases en un espacio bidimensional. b) Clase 1 contra clase 2: $y_1 = +1; y_2 = -1$ . c) Clase 1 contra clase 3: $y_1 = +1; y_3 = -1$ . d) Clase 2 contra clase 3: $y_2 = +1; y_3 = -1$ . . . . .	127
5.16	Ejemplo de conversión de árbol binario a matriz DECOG. . . . .	132
5.17	Ejemplo de matriz de codificación DECOG para el conjunto de las señales azules circulares. . . . .	133
6.1	Modelos de señales como contribución de partes. (a) Modelo de tres partes correspondientes a orla, fondo y pictograma. (b) Modelo de dos partes: fondo y pictograma. . . . .	148
6.2	Ejemplo de extracción del IM para una señal roja circular con una rejilla de $3 \times 3$ celdas. (a) Blob original. (b) Blob en escala de gris después de eliminar la orla externa mediante la máscara. (c) Gradiente vertical del blob $G_v(x, y)$ . (d) Gradiente horizontal del blob $G_h(x, y)$ . (e) Magnitud del gradiente $G_{mag}(x, y)$ . (f) Mapa del pictograma $IM$ . . . . .	149
6.3	Ejemplos de mapas de ideograma (IMs). (a) IMs de señales rojas circulares ( $20 \times 20$ píxeles) usando una rejilla de $3 \times 3$ celdas. (b) IMs de señales azules rectangulares ( $30 \times 30$ píxeles) usando una rejilla de $3 \times 3$ celdas. . . . .	150

6.4	Ejemplo de agrupamientos para señales rojas circulares con $N = 4$ . (a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. . . . .	152
6.5	Diagrama conceptual del algoritmo, el cual construye el IM de cada muestra de entrada a partir de la magnitud del gradiente. El algoritmo consta de dos etapas: 1) selección de los clusters con mayor probabilidad y 2) identificación del ideograma a partir de los clusters seleccionados. . . . .	153
6.6	Ejemplo de correcta asignación de cluster. (a) Muestra original segmentada. (b) Muestra en escala de gris. (c) Muestra después de eliminar la orla. (d) Máscara de bordes. (e) Mapa del ideograma (IM). (f) Estimaciones de probabilidad a la salida de las SVM. . . . .	155
6.7	Ejemplo de una problemática asignación de cluster. (a) Muestra original segmentada. (b) Muestra en escala de gris. (c) Muestra después de eliminar la orla. (d) Máscara de bordes. (e) Mapa del ideograma. (f) Estimaciones de probabilidad a la salida de las SVM. . . . .	156
6.8	Ejemplo de una problemática asignación de cluster. (a) Muestra original segmentada. (b) Muestra en escala de gris. (c) Muestra después de eliminar la orla. (d) Máscara de bordes. (e) Mapa del ideograma. (f) Estimaciones de probabilidad a la salida de las SVM. . . . .	156
6.9	Tasa de acierto en la asignación de 4 clusters y rejilla de $3 \times 3$ celdas. . . . .	157
6.10	Extracción del descriptor basado en un vector HOG de longitud variable para cada celda. . . . .	158
6.11	Coefficiente de silueta para clustering K-means. . . . .	160
6.12	Tiempo total de procesado por muestra en función del número de clusters con una rejilla de $5 \times 5$ y $n_c = 1$ . . . . .	161
6.13	Curvas ROC en función del número de clusters. . . . .	162
6.14	Tiempo total de procesado por muestra en función del número de clusters asignados con $N = 4$ y una rejilla de $5 \times 5$ . . . . .	163
6.15	Tasa de acierto en función del tamaño de la rejilla con $N = 4$ y $n_c = 1$ . . . . .	163
6.16	Número de características del vector HOG en cada cluster en función del tamaño de la rejilla con $N = 4$ . . . . .	164
6.17	Número total de vectores soporte en función del tamaño de la rejilla con $N = 4$ . . . . .	164
6.18	Tiempo de procesado por muestra para diferentes estructuras de rejilla con $N = 4$ y $n_c = 1$ . . . . .	165
B.1	Problema de separación de patrones con datos separables linealmente. . . . .	183
B.2	Clasificación lineal óptima. . . . .	185
B.3	Ejemplo de clases no separables con puntos que caen dentro del margen de separación. . . . .	187
B.4	Ejemplo de transformación de espacios $\mathbb{R}^2 \rightarrow \mathbb{R}^3$ . . . . .	190
B.5	Arquitectura SVM empleando funciones kernel. . . . .	191

C.1	Señales de advertencia de peligro . . . . .	194
C.2	Señales de reglamentación - Prioridad. . . . .	195
C.3	Señales de reglamentación - Prohibición de entrada. . . . .	195
C.4	Señales de reglamentación - Otras prohibiciones o restricciones. . . . .	196
C.5	Señales de reglamentación - Señales de obligación. . . . .	197
C.6	Señales de reglamentación - Fin de prohibición. . . . .	197
C.7	Señales de indicación - Indicaciones generales. . . . .	198
C.8	Señales de indicación - Carriles. . . . .	199
C.9	Señales de indicación - Servicio. . . . .	200
C.10	Señales de orientación - Preseñalización. . . . .	201
C.11	Señales de orientación - Dirección. . . . .	202
C.12	Señales de orientación - Localización. . . . .	203
C.13	Señales de orientación - Confirmación. . . . .	203
C.14	Señales de orientación - Uso específico de poblado. . . . .	203
C.15	Señales de orientación - Paneles complementarios. . . . .	204
C.16	Señales de orientación - Otras señales. . . . .	204
C.17	Señales de obras - Advertencia de peligro. . . . .	205
C.18	Señales de obras - Prohibiciones. . . . .	206



# Lista de tablas

1.1	Causas de mortalidad a nivel mundial por rangos de edad en el año 2004. . . . .	2
1.2	Número de víctimas mortales en carretera y porcentaje de cambio en la década 2001-2010. . . . .	2
1.3	Matriz de confusión de un sistema de clasificación binario. . . . .	9
2.1	Clasificación de las señales de tráfico según el color y la forma. . . . .	15
3.1	Información de las bases de datos públicas de señales viales. . . . .	50
3.2	Distribución de clases y muestras en la base de datos de señalización vertical. . . . .	53
3.3	Distribución de patrones en la base de datos para las señales rojas circulares. . . . .	55
4.1	Umbrales de los canales cromáticos. . . . .	64
4.2	Umbrales de segmentación para visibilidad nocturna. . . . .	66
4.3	Valores óptimos de los parámetros de las SVM. . . . .	82
4.4	Configuración de los datos para entrenamiento y test. . . . .	82
4.5	Información sobre los vectores soporte de los clasificadores. . . . .	83
4.6	Resultados del módulo de reconocimiento utilizando la base de datos. . . . .	88
4.7	Resultados del sistema de detección y reconocimiento sobre un conjunto de secuencias. . . . .	89
4.8	Distancia máxima de seguimiento para tamaños normalizados de señales viales en España. . . . .	94
4.9	Señales detectadas a la salida del módulo de tracking para una secuencia de ejemplo. . . . .	99
4.10	Falsas alarmas descartadas por el módulo de tracking para una secuencia de ejemplo. . . . .	99
4.11	Resultados del sistema completo usando la información de tracking. . . . .	100
4.12	Carga computacional de los módulos del sistema. . . . .	102
5.1	Valores de variación de los parámetros para las técnicas de preprocesado. . . . .	114

5.2	Valores de variación de los parámetros de los descriptores. . . . .	114
5.3	Valores óptimos de los parámetros de configuración. . . . .	115
5.4	Análisis comparativo de resultados para diferentes técnicas de preprocesado y extracción de descriptores. . . . .	120
5.5	Resultados comparativos de HOG ( $TPR(\%)$ y $T_t(\text{ms})$ ) en función de los parámetros $N_b$ y $N_w$ . . . . .	121
5.6	Resultados comparativos de Normalización+LBP ( $TPR(\%)$ y $N_{sv}$ ) en función de los parámetros $P$ y $R$ . . . . .	122
5.7	Resultados comparativos de Normalización+LBP ( $TPR(\%)$ y $T_t(\text{ms})$ ) en función de los parámetros $N_b$ y $N_w$ . . . . .	122
5.8	Matrices para clasificación de 4 clases con arquitectura OVA(a) y OVO(b). . . . .	129
5.9	Matrices para clasificación de 4 clases con arquitectura de aleatoriedad densa(a) y aleatoriedad dispersa(b). . . . .	131
5.10	Matriz de contingencia del test de McNemar. . . . .	137
5.11	Resultados de clasificación con diferentes arquitecturas basadas en SVM. . . . .	141
5.12	Resultados de clasificación en función de las técnicas de decodificación. . . . .	143
5.13	Tests estadísticos sobre los conjuntos de datos. Para cada pareja de clasificadores enfrentados se aporta el número de victorias (W-Win), empates (T-Ties) y derrotas (L-Loss) del clasificador de la fila frente al clasificador de la columna basándose en el promedio de la tasa de acierto y en la significancia estadística a partir de los tests de McNemar y Wilcoxon. . . . .	144
6.1	Información de los clusters con una rejilla de $5 \times 5$ . . . . .	161
6.2	Promedio de las varianzas de los IMs en cada cluster en función del tamaño de la rejilla con $N = 4$ . . . . .	164
6.3	Análisis comparativo entre el método propuesto (selección cluster+HOG variable) y HOG. . . . .	166
6.4	Análisis comparativo entre el método propuesto (selección cluster+HOG variable) y HOG. . . . .	166

# Capítulo 1

## Introducción

El sector del transporte es un elemento clave en el conjunto de la economía de los países desarrollados y en la Unión Europea [Ue] supone directamente el empleo de unos diez millones de personas, lo que representa cerca del 5% del PIB. Sobre el volumen de crecimiento, es importante destacar que en los últimos treinta años se ha triplicado la distancia total recorrida por los vehículos en carretera y en la última década el volumen de transporte rodado ha crecido en un 35%. Del alto número de usuarios de la vía, es razonable justificar que uno de los principales problemas en términos de movilidad es la congestión, la cual implica cierta ineficiencia de la infraestructura y pérdida de productividad en el transporte de mercancías.

En términos de bienestar, dentro de los principales problemas relacionados con el transporte a los que se enfrenta la sociedad, habría que destacar los accidentes de tráfico. De acuerdo al informe publicado por la Organización Mundial de la Salud (ver [WorldHealth09]), cada año fallecen a nivel mundial, aproximadamente, 1.2 millones de personas y entre 20 y 50 millones de ciudadanos son víctimas de accidentes viales con diferentes tipos de secuelas. De hecho, los accidentes en carretera suponen una de las principales causas de mortalidad en los países desarrollados. Tal es así que según el informe elaborado por [Peden04], los siniestros en carretera suponen una de las tres primeras causas de mortalidad en el rango de edad comprendido entre los 15 y los 44 años (ver Tabla 1.1). La importancia de la señalización en la seguridad en carretera constituye el principal motivo que ha dado lugar al desarrollo de esta tesis doctoral.

Al margen de las innumerables repercusiones sociales, el coste económico de cada siniestro en carretera es elevado (ver [Trafico05]). En el 7.8% de los accidentes en carretera los costes superan los 100 000 euros, el 5.2% entre 50 001 y 100 000, el 12.8% entre 25 001 y 50 000, el 25.6% entre 10 001 y 25 000, el 34.2% entre 5001 y 10 000, y el 14.4% entre 1001 y 5000 euros. Estos costes incluyen, además de la indemnización, los derivados de los tratamientos médicos, los daños materiales e inmateriales, los gastos legales y el tiempo de baja. Es posible cuantificar el impacto del accidente mediante un índice de Dolor Social (IDS), el cual incluye entre sus variables factores tales como los días de

Tabla 1.1: Causas de mortalidad a nivel mundial por rangos de edad en el año 2004.

<i>Posición</i>	<i>5-14</i>	<i>15-29</i>	<i>30-44</i>	<i>45-59</i>	<i>&gt;60</i>
1	Malformaciones	VIH	VIH	Infarto	Infarto
2	<b>Accidentes de tráfico</b>	<b>Accidentes de tráfico</b>	Tuberculosis	Problemas cardiovasculares	Problemas cardiovasculares
3	Infecciones respiratorias	Suicidio	<b>Accidentes de tráfico</b>	Tuberculosis	Problemas pulmonares
4	VIH	Tuberculosis	Ictus cerebral	VIH	Problemas respiratorios
5	Axfisia	Violencia	Suicidio	Problemas pulmonares	Cáncer de pulmón
6	Malaria	Infecciones respiratorias	Violencia	Cáncer de pulmón	Diabetes
7	Enfermedades tropicales	Quemaduras	Enfermedades cerebrovasculares	Cirrosis	Hipertensión
8	Quemaduras	Axfisia	Cirrosis	<b>Accidentes de tráfico</b>	Cáncer de estómago

baja y de hospitalización del accidentado, su incapacidad profesional, la necesidad de una tercera persona y la posible necesidad de adaptación de la vivienda.

Afortunadamente, en los últimos años se ha registrado un descenso considerable en el número de víctimas mortales en carretera. La incorporación de nuevas tecnologías en los vehículos, las mejoras en las infraestructuras, la mayor concienciación de los conductores y las fuertes medidas sancionadoras de los gobiernos contra las infracciones explican esta mejora en la seguridad vial. En la década 2001-2010, el número de víctimas mortales en carretera descendió en un 43 % en los 27 países de la Unión Europea (UE), de acuerdo al informe elaborado por el Consejo Europeo en Seguridad Vial (ETSC-*European Transport Safety Council*)[Jost11]. En la tabla 1.2 se muestra la evolución del número de siniestros registrados en esta década en algunos países miembros. En cualquier caso, cerca de 31.000 personas perdieron su vida en accidentes de tráfico en los países de la Unión Europea en 2010 y es por ello, que el ETSC se ha marcado como objetivo conseguir reducir a la mitad el número de víctimas mortales anuales antes de 2020.

Tabla 1.2: Número de víctimas mortales en carretera y porcentaje de cambio en la década 2001-2010.

<i>País</i>	<i>2001</i>	<i>2002</i>	<i>2003</i>	<i>2004</i>	<i>2005</i>	<i>2006</i>	<i>2007</i>	<i>2008</i>	<i>2009</i>	<i>2010</i>	<i>2010-2001</i>
Alemania	6977	6842	6613	5842	5361	5091	4949	4477	4152	3651	-48 %
Austria	958	956	931	878	768	730	691	679	633	552	-42 %
Bélgica	1486	1306	1214	1162	1089	1069	1067	944	944	840	-43 %
Dinamarca	431	463	432	369	331	306	406	406	303	265	-39 %
España	5517	5347	5399	4741	4442	4104	3823	3100	2714	2470	-55 %
Francia	8162	7655	6058	5530	5318	4703	4620	4275	4273	3.992	-51 %
Grecia	1880	1634	1605	1670	1658	1657	1612	1553	1456	1281	-32 %
Italia	7096	6980	6563	6122	5818	5669	5131	4725	4237	3998	-44 %
Portugal	1670	1668	1542	1294	1247	969	974	885	840	845	-49 %
Suecia	531	515	512	463	423	428	454	380	341	266	-50 %
UK	3598	3581	3658	3368	3337	3300	3056	2718	2337	1943	-46 %

Para prevenir los accidentes de tráfico es necesario conocer las causas de siniestralidad. Según datos de la DGT [Trafico11], las distracciones (23 % de los casos), las infracciones a normas de circulación (17 %) y la velocidad inadecuada (11 %) fueron las principales causas de accidentalidad. Por tipo de accidente, un 32 % fueron salidas de vía, seguido de colisiones frontales (22 %) y frontolaterales (19 %). Un estudio estadounidense sobre el comportamiento del conductor realizado por el *National Highway Traffic Safety Administration* [Hanowski06] encontró que el 99 % de los accidentes analizados eran debidos a errores humanos, siendo la causa más común los errores de percepción. Todos estos datos revelan que el conductor sigue siendo el principal responsable de los siniestros. Además, ponen de manifiesto la necesaria intervención en materia de seguridad activa para ofrecer al usuario los mecanismos suficientes que le ayuden a reducir el riesgo de colisión.

Con la finalidad de mejorar especialmente la seguridad y, además, incrementar la eficiencia y el confort en el transporte terrestre y reducir su impacto medioambiental, surgen los sistemas inteligentes de transporte (*ITS-Intelligent Transportation Systems*), los cuales integran de manera sinérgica las tecnologías de la electrónica y las telecomunicaciones en el sector del transporte. Aunque las mismas tecnologías utilizadas por los ITS pueden aplicarse a otros modos de transporte, su uso en los ITS es orientado al transporte terrestre. Se trata de una disciplina joven y de rápida evolución, lo que dificulta el consenso de una definición única. Según la Sociedad Americana de Transporte Inteligente (ITS América) ITS se define como "gente usando tecnología en transportes para salvar vidas, tiempo y dinero".

## 1.1. Sistemas ITS

Las mejoras que introducen los ITS se consiguen mediante la provisión de información a los usuarios del transporte. Además, permite a los transportistas de mercancías y a las administraciones fronterizas una mayor comodidad y agilidad en las gestiones, sin renunciar a la necesaria seguridad y seguimiento de la mercancía en tiempo real.

La implantación de los ITS contribuye significativamente a resolver problemas presentes en los sistemas de transporte. Sin embargo, no hay que perder de vista los procesos desarrollados en otros países, donde la experiencia acumulada muestra que la explotación de las tecnologías ITS sólo tiene un impacto global significativo si los gobiernos tienen voluntad de colaborar en el desarrollo de las arquitecturas necesarias para ello. Una implantación adecuada de los ITS trata de alcanzar los siguientes objetivos:

1. Mejorar de manera sostenible la seguridad de las personas y mercancías involucradas o afectadas por el transporte y el tráfico.
2. Optimización de la explotación de los recursos de transporte, atendiendo a su capacidad, disponibilidad, fiabilidad, etc., tanto de manera individual para cada modo de transporte como conjuntamente.

3. Armonización, estandarización, definición de compatibilidades entre sistemas y claridad en su presentación al usuario. En este punto se tomarán en cuenta aspectos tanto económicos y socio-culturales como ecológicos del entorno.

Las tecnologías ITS están llamadas a desarrollar una labor fundamental en la reducción del número de accidentes. Son numerosas las iniciativas que se han desarrollado en los últimos años relacionadas con los sistemas de seguridad que incorpora el vehículo para minimizar el daño en caso de accidente. Sin embargo, los diversos actores implicados en las tecnologías ITS son conscientes de la importancia de poder atajar el problema en una etapa anterior. En este aspecto, se vienen desarrollando sistemas capaces de prevenir accidentes mediante el aviso de la presencia de obstáculos o riesgos en la carretera antes de llegar a ellos.

### 1.1.1. Clasificación

Los sistemas ITS pueden subdividirse en dos categorías dependiendo de su aplicación: a) infraestructuras inteligentes, que incorporan sistemas con sensores que se instalan en puntos estratégicos de la red viaria y tratan de mejorar la seguridad y eficiencia de la circulación y b) vehículos inteligentes, que incluyen sistemas embebidos a bordo del vehículo.

El rango de funcionalidades de los sistemas ITS en el marco de infraestructuras inteligentes es muy amplio. Dentro del mismo, se pueden destacar, entre otras, las aplicaciones relacionadas con el control de la velocidad de los vehículos, la estimación de la distancia entre vehículos, la clasificación de vehículos para estadísticas del flujo de tráfico, el reconocimiento de matrículas y la detección de incidentes. En esta última funcionalidad, cabe mencionar, entre otras, situaciones que implican accidentes, congestiones y realización de infracciones por parte de los usuarios. La incorporación de las tecnologías en toda la red viaria nacional supondría un coste prohibitivo, por lo cual su uso queda reducido a vías determinadas.

En el caso de sistemas embebidos en vehículos inteligentes, se pueden destacar tres conocidas aplicaciones:

- Sistemas avanzados de asistencia a la conducción (*ADAS-Advanced Driver Assistance System*): se encuentran en pleno desarrollo actual y son numerosos los proyectos activos relacionados con estos sistemas, tanto a nivel de la industria del automóvil (Citröen, Volkswagen, Mercedes, etc.), como universidades (Universidades Carlos III y Alcalá en España) y centros de investigación (*CSIC-Consejo Superior de Investigaciones Científicas*). De acuerdo a la forma de actuar de estos sistemas, hablamos de seguridad primaria o activa y secundaria o pasiva, entendiendo por seguridad activa la relacionada con los sistemas que se encargan de evitar accidentes potenciales, mientras que la seguridad pasiva comprende los sistemas diseñados para minimizar las consecuencias del accidente en el caso de que éste, finalmente, no pueda ser evitado. Como principales sistemas de seguridad primaria se pueden citar los de

asistencia a la frenada, mejora de la visibilidad nocturna, detección automática de señalización en la vía y detección de peatones en la escena. Entre las mejoras en los sistemas de seguridad secundaria destacan el desarrollo de nuevos materiales para el frontal de los vehículos, los cinturones de seguridad, el reposacabezas, la propia estructura de los capós, los parachoques delanteros y otros más recientes, como los capós activos y los *airbags*. Dentro de los sistemas activos y de acuerdo a los niveles de interacción con el conductor, podemos distinguir entre aquellos que tan sólo incluyen funciones de aviso ante una situación de peligro, ya sea con señales acústicas u ópticas, pero no toman nunca el control del vehículo, y los sistemas que incluyen la interacción sobre uno o varios elementos implicados en la tarea de conducción.

- Vehículos autónomos. Se definen como tal aquellos vehículos con capacidad para tomar el control completo del vehículo bajo determinadas circunstancias. La conducción automática se considera, hoy en día, como la última etapa a desarrollar en el proceso evolutivo de los ITS. Los beneficios de una conducción completamente automática son numerosos. Sin embargo, existen algunas dificultades a nivel técnico, económico y legal que aún deben ser superadas. El sistema debe operar en tiempo real con un grado de fiabilidad muy alto, cercano al 100 % sin que ello suponga incrementos notables sobre el precio final del vehículo. Paralelamente, deben resolverse cuestiones legales para depurar responsabilidades ante posibles siniestros y por otra parte, debe introducirse un cambio de mentalidad en los conductores, que se desplazarán como simples viajeros a bordo.
  
- Sistemas de inventariado. El mantenimiento y optimización de la infraestructura de carreteras supone un punto crítico en la seguridad vial de los conductores. Por ello, se hace necesario realizar periódicamente tareas de mantenimiento con el objetivo de detectar posibles problemas relacionados con el estado del firme y de los diferentes elementos de señalización. Los ITS abren un nuevo horizonte en estas tareas, las cuales tradicionalmente eran desempeñadas de manera manual por operarios. En este sentido, este trabajo de inspección puede ser realizado de manera automática mediante el uso de diferentes tecnologías. Combinando la potencia de un sistema automático para reconocimiento de elementos de señalización con un sistema de información geográfica (*GIS-Geographical Information Systems*), es factible desarrollar una base de datos con información precisa de la localización de los elementos de señalización. No obstante, hasta el momento, existen pocos sistemas encaminados a esta utilidad y además, tienen importantes limitaciones.

El desarrollo de cualquier sistema embebido inteligente, especialmente en aplicaciones orientadas a sistemas de ayuda a la conducción y vehículos autónomos, implica la consecución de numerosas funciones que pueden ser integradas en un sistema conjunto. Entre las más importantes, cabe destacar las siguientes:

- **Detección y reconocimiento de señalización vertical:** se trata de sistemas que de manera automática detectan e identifican las señales presentes en ruta a lo largo de un recorrido. Para ello existen diferentes propuestas, tales como sistemas que extraen información de satélites de navegación alimentados con una base de datos actualizada de las señales viales o sistemas capaces de recibir la información a través de radiotransmisores instalados en los propios elementos de señalización. Sin embargo, los sistemas basados en visión artificial, los cuales analizan la secuencia de imágenes capturada por una cámara integrada en el vehículo, son los más extendidos.
- **Aviso de salida del carril:** el objetivo es conocer la ubicación del vehículo en relación a las marcas delimitadoras de los carriles. Si el vehículo se desvía de su trayectoria natural superando ciertos umbrales de distancia o tiempo para salirse del carril, el conductor es alertado antes de abandonar la vía. Añadiendo otros componentes, el sistema podría ofrecer información correctiva cuando esté circulando en tramos de obras y condiciones de poca visibilidad.
- **Aparcamiento asistido/automático:** se pretende incorporar funciones de búsqueda de espacio de aparcamiento, las cuales deben tener capacidad para localizar un espacio disponible donde poder estacionar y calcular los movimientos necesarios para aparcar de la forma más óptima posible. El conductor únicamente se encarga de pisar los pedales de freno y acelerador para controlar la velocidad del vehículo.
- **Control integral de maniobras:** la función de este módulo es actuar de forma integral sobre todos los componentes activos, tales como dirección, frenos, cambios de marcha y acelerador, en una situación concreta.
- **Sistema de visión nocturna mejorado:** se trata de sistemas que utilizan una unidad de infrarrojos de corto o largo alcance para mejorar la capacidad de percepción del conductor en entornos de baja iluminación.
- **Percepción del entorno:** el objetivo de esta tarea es integrar y fusionar toda la información procedente del entorno del vehículo (360°) para crear un modelo holístico. Esto permite detectar la proximidad de vehículos para evitar colisiones y por otra parte, detectar la presencia de peatones o animales para evitar posibles atropellos.
- **Control del conductor:** el objetivo es determinar si el estado físico y psíquico del conductor es idóneo para conducir un vehículo. Basándose en medidas biomédicas, tales como el movimiento del iris, el ritmo de parpadeo, la apertura del diafragma, la dilatación de la pupila y la coloración de la piel, se puede estimar el grado de fatiga del conductor.

### 1.1.2. Arquitectura

Para cualquier aplicación ITS pueden desarrollarse sistemas basados en muy diferentes tecnologías. Independientemente de cual de ellas sea utilizada, la mayor parte de los ITS constan de una arquitectura basada en tres componentes: el transductor o sensor, un módulo encargado de procesar la señal y un segundo módulo responsable de procesar los datos, a partir de los cuales extrae ciertos parámetros de tráfico. Respecto a los sensores, cabe distinguir entre sensores invasivos y no invasivos. Se consideran de naturaleza invasiva aquellos cuya instalación y reparación supone interrupciones en la circulación de vehículos. Entre los mismos, se encuentran las espiras, los magnetómetros y los sensores para pesado de vehículos en movimiento. Por contrapartida, se habla de sensores no invasivos al referirnos a aquellos que son fáciles de instalar sin interferir el tráfico rodado y entre los mismos, cabe citar las videocámaras para sistemas de visión artificial, los radares de microondas, los radares laser, los sensores ultrasónicos y los sensores infrarrojos. En el caso de los sistemas embebidos en vehículos, lógicamente, es necesario contar con sensores no invasivos y dentro de éstos, las videocámaras, como sensores de sistemas basados en visión artificial, han supuesto una auténtica revolución tecnológica en los últimos años.

Obviamente, mientras que el uso de otro tipo de sensores podría tener la ventaja de extender las capacidades cognitivas más allá de las posibilidades del ser humano, la visión por computador falla en las mismas situaciones en las que los humanos no pueden ver, tales como condiciones de lluvia intensa, niebla y oscuridad sin iluminación específica. No obstante, éste último inconveniente puede solventarse con el uso de otro tipo de sensores. Por ejemplo, puede mejorarse la visión en entornos con poca luz con un sistema de iluminación o con el uso de cámaras infrarrojas.

Para cumplir estos objetivos, un sistema de visión avanzado debe incorporar, además de los dispositivos de captación de imágenes, los equipos y algoritmos de visión artificial que realicen las funciones encomendadas. Normalmente, los equipos empleados para realizar el procesado de video suelen consistir en un ordenador con recursos suficientes, o bien sistemas de microprocesadores diseñados específicamente para dicho fin, mientras que los algoritmos son en realidad, los programas o rutinas que están siendo ejecutados por dichos equipos.

## 1.2. Requisitos de los sistemas ITS

Para que un sistema ITS pueda ser utilizado, se ha de evaluar su grado de eficacia y eficiencia en base a una serie de características que debe cumplir, siendo las más importantes:

- Sencillez y velocidad: como en todo sistema informático, la complejidad computacional de un programa está limitada por los recursos del equipo en el que se ejecuta, ya que cuanto más complejo sea y mayor cantidad de recursos necesite el algoritmo,

más caro será el equipo empleado en el sistema. En contrapartida, cuanto más simple sea el algoritmo, menor velocidad se requerirá del equipo que va a ejecutarlo, con la consiguiente reducción del precio final del sistema. En esta línea cabe destacar que para la mayor parte de las aplicaciones ITS, el tiempo de ejecución del algoritmo es un elemento muy crítico en el diseño, ya que la utilidad de que el sistema alerte al conductor de un cierto peligro es reducida si ya ha ocurrido un siniestro.

- **Fiabilidad:** el objetivo final de la mayor parte de los ITS consiste en determinar en cada instante determinadas incidencias. En este sentido, y suponiendo que el sistema no va a ser perfecto, existirán ciertas probabilidades de error que deben ser evaluadas. Para cuantificar la calidad de un sistema ITS se pueden definir las siguientes probabilidades:
  - Tasa o probabilidad de detección (*DR-Detection Rate*): se define como la probabilidad de detección de los hitos reales. De forma inversa, también se puede definir la probabilidad de pérdida como la probabilidad de que un hito que debería ser detectado, no lo sea.
  - Probabilidad de falsa alarma (*FPR-False Positive Rate*): Debido a errores en los algoritmos, en algunas situaciones el sistema advierte de incidencias que no se dan en la realidad. En estos casos se habla de falsa alarma. Aunque las consecuencias de las falsas alarmas no son tan graves como las pérdidas, una probabilidad de falsa alarma demasiado alta puede llegar a saturar el sistema, haciendo que éste sea inútil.
- **Precio:** Si uno de los objetivos finales del diseño de un sistema ITS es conseguir la mayor fiabilidad posible, otro no menos importante se refiere a la reducción del precio total del sistema. El precio dependerá de los equipos, los algoritmos de procesado empleados y el sensor. Se buscará que tanto los sensores como los medios de transmisión empleados sean lo más sencillos y baratos posibles, por lo que los algoritmos empleados deberían poder corregir estos defectos buscando de nuevo la mayor fiabilidad posible.

### 1.3. Métrica de evaluación

En sistemas ITS que hagan uso de funciones orientadas a la clasificación binaria de patrones existen dos posibles salidas (positivo y negativo), las cuales se dividen en cuatro categorías de acuerdo a la matriz de confusión de la tabla 1.3: verdaderos positivos (*TP-True Positives*) son objetos que han sido clasificados correctamente, falsos positivos (*FP-False Positives*) son falsas alarmas clasificadas erróneamente como objetos, verdaderos negativos (*TN-True Negatives*) son falsas alarmas identificadas como tal y falsos negativos (*FN-False Negatives*) son objetos de interés clasificados como falsas alarmas. A partir de la tabla 1.3 se pueden definir tres parejas de parámetros:

- Precision-Recall: la precisión es la proporción de objetos clasificados como positivos que son realmente positivos y la sensibilidad (Recall) es la proporción de objetos positivos que son clasificados como positivos, definiéndose como:

$$Precision = \frac{TP}{TP + FP} \quad (1.1)$$

y

$$Recall = \frac{TP}{TP + FN}. \quad (1.2)$$

- Sensibilidad-especificidad: la sensibilidad es la proporción de objetos positivos que son clasificados como positivos y la especificidad es la proporción de muestras negativas que son realmente identificadas como tal a partir de:

$$Sensibilidad = \frac{TP}{TP + FN} \quad (1.3)$$

y

$$Especificidad = \frac{TN}{TN + FP}. \quad (1.4)$$

- Tasa de verdaderos positivos-tasa de falsos negativos: la tasa de verdaderos positivos (*TPR-True Positive Rate*) es la proporción de objetos positivos que son clasificados como positivos, mientras que la tasa de falsos positivos (*FPR-False Positive Rate*) o tasa de falsa alarma es la proporción de falsas alarmas que son clasificadas como objetos.

$$TPR(= Sensibilidad = Recall) = \frac{TP}{TP + FN} \quad (1.5)$$

y

$$FPR = \frac{FP}{TN + FP} = 1 - Especificidad. \quad (1.6)$$

		Etiquetas reales	
		Positivo	Negativo
Predicción	Positivo	TP	FP
	Negativo	FN	TN

Tabla 1.3: Matriz de confusión de un sistema de clasificación binario.

Debido al interés por evaluar la capacidad del sistema al rechazo de falsas alarmas, en esta tesis doctoral se trabajará con el par TPR-FPR. En un problema de multclasificación con  $M$  clases, se puede definir la TPR para cada clase  $i$ -ésima como la proporción de muestras pertenecientes a la clase que son identificadas correctamente de acuerdo a la siguiente ecuación:

$$TPR_i = \frac{TP_i}{TP_i + FN_i}, \quad (1.7)$$

donde  $TP_i$  indica el número de muestras de la clase que han sido clasificadas correctamente (verdaderos positivos) y  $FN_i$  las muestras de la clase clasificadas incorrectamente. El parámetro TPR global de un sistema de multclasificación indica el grado de acierto en la categorización de los objetos y se puede calcular a partir del promedio de la TPR de todas las clases como:

$$TPR = \frac{1}{M} \sum_{i=1}^M \frac{TP_i}{TP_i + FN_i}, \quad (1.8)$$

mientras que la FPR viene dada por la ecuación 1.6 donde ahora FP hace referencia a las falsas alarmas que son confundidas con cualquier otra clase.

Como medida de calidad del sistema, también se utiliza la curva característica operativa del receptor (*ROC-Receiver Operating Characteristic*) [Powers11, Davis06], que es una representación gráfica del desempeño de un clasificador, en la que se muestra la evolución de la tasa promedio de aciertos positivos (TPR) en función de la tasa de falsa alarma (FPR), tal como puede verse en la figura 1.1. Una clasificación perfecta dejaría un punto en la coordenada (0,1) del espacio ROC, representando una TPR=100 % y una FPR=0 %. Un clasificador aleatorio daría un punto a lo largo de la diagonal (franja no discriminatoria) que va del punto de coordenadas (0,0) al punto (1,1) y que en la figura 1.1 se representa por una línea discontinua. Los parámetros TPR y FPR se pueden combinar a través de un único parámetro *F - score* denominado factor de mérito, que da idea de la seguridad del sistema y se define como:

$$F = 2 \times \frac{TPR \cdot (1 - FPR)}{TPR + (1 - FPR)}, \quad (1.9)$$

teniendo en cuenta que éste alcanza el valor unidad en el mejor caso (TPR=1 y FPR=0) y 0 en el peor (TPR=0 ó FPR=1). En base a la representación gráfica de la curva ROC, otra medida de calidad empleada es el área encerrada por la misma (*AUC-Area Under Curve*). Este área posee un valor comprendido entre 0.5 y 1, donde 1 representa un desempeño perfecto y 0.5 corresponde a un sistema sin capacidad discriminatoria.

## 1.4. Descripción de un sistema ITS basado en visión artificial

Los sistemas ITS basados en técnicas de procesamiento digital de imagen artificial constan, generalmente, de una estructura modular común para la gran mayoría de las aplicaciones, tal como se muestra en el diagrama de bloques de la figura 1.2. El primer eslabón se corresponde con la detección de objetos, que puede implementarse con diferentes técnicas. Aunque en el esquema de los sistemas clásicos, la detección se ha implementado mediante un proceso de segmentación, en la actualidad existen métodos emergentes que directamente buscan los objetos a partir de la extracción de características [Dalal05]. En

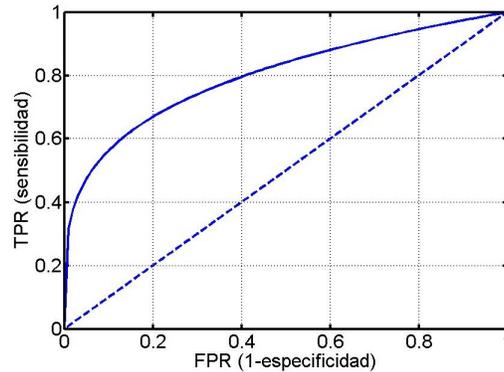


Figura 1.1: Ejemplo de curva ROC.

cualquier caso, la salida del sistema de detección es una lista de objetos y el siguiente paso trata de establecer correspondencias entre los objetos de la imagen actual y los objetos detectados en imágenes previas mediante un proceso de *tracking*. La función de seguimiento permite tener un conocimiento continuo de la trayectoria que ha seguido el objeto y eliminar muchas falsas alarmas.

Dependiendo de la aplicación del sistema, será necesario extraer información variada de los objetos de interés, tal como la distancia a la que se encuentran de la cámara, su posición en el escenario e incluso su categorización, llegando a discernir el tipo de objeto de que se trata. Esta etapa de clasificación será la encargada de determinar el tipo concreto de una señal vial en un sistema de clasificación de señales o el tipo de vehículo en una aplicación orientada a la clasificación de vehículos en una vía. Finalmente, la salida del bloque de reconocimiento puede ser empleada para interpretar el comportamiento de los objetos. En sistemas orientados a aplicaciones de vigilancia en carretera, esta funcionalidad lleva asociado el análisis de trayectorias de los vehículos para detectar situaciones anómalas como maniobras prohibidas o presencia de vehículos averiados.

## 1.5. Motivación de la tesis

Las distintas funcionalidades que pueden ser incluidas en un sistema ITS barren un amplísimo espectro de oportunidades de investigación. Tal y como ha sido explicado anteriormente, existen muy diferentes aplicaciones en el campo de los ITS: desde el reconocimiento de matrículas hasta el análisis físico-psíquico del conductor, por lo que la implementación de un algoritmo concreto va a depender, en gran medida, de cuál sea su utilidad final. De hecho, las publicaciones existentes en la literatura de este campo se basan, por lo general, en una utilidad concreta e incluso en módulos específicos para una determinada utilidad. En concreto, la finalidad de esta tesis doctoral es desarrollar un sistema automático de reconocimiento de señalización vial vertical. A continuación se exponen los objetivos iniciales marcados en esta tesis.

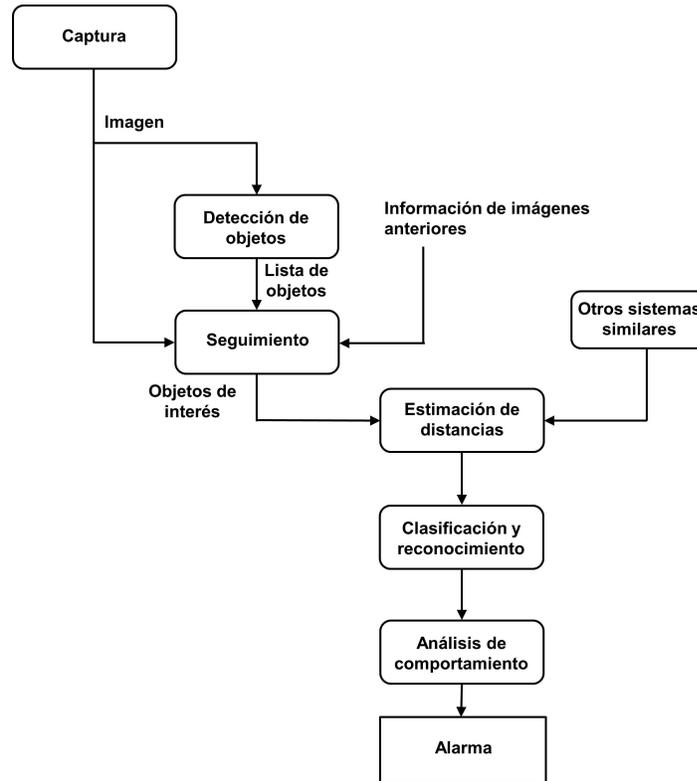


Figura 1.2: Esquema de bloques general de un sistema ITS basado en visión artificial.

### 1.5.1. Objetivos de la tesis

Uno de los principales objetivos de los ITS es la reducción del número de accidentes de tráfico debidos a errores humanos. Las cifras de siniestros reflejan un elevado número de accidentes como consecuencia de que el conductor no respete la señalización o bien que la señalización no sea fácilmente perceptible. En los últimos años la industria automovilística, los proveedores de sistemas eléctricos y los investigadores han desarrollado de forma conjunta sistemas automáticos de detección de señalización con el objetivo de reducir las tasas de accidentes. La necesidad de un sistema que advierta al conductor de la presencia de cierta señalización, tal como una señal de ‘Stop’ en un cruce, es manifiesta. Por otra parte, la percepción de la señalización depende de su estado de conservación y las tareas de inventariado en carretera pueden automatizarse mediante técnicas de visión artificial. Estos dos hechos han motivado el desarrollo de esta tesis, cuyo objetivo final es la implementación de un sistema automático para el reconocimiento de señales de tráfico (TSDRS-(*Traffic Sign Detection and Recognition System*)).

Inicialmente, se plantearon una serie de hitos para la obtención de este objetivo, los cuales se pueden resumir en los siguientes puntos:

1. Construir un módulo para reconocimiento de señales de tráfico que sea fácilmente configurable, contemplando la posibilidad de utilizar el módulo con la señalización

vial vertical de cualquier país. Cabe destacar que la información del exterior es percibida a través de un sistema de cámaras instalado en el vehículo y el objetivo es que el sistema sea robusto a las condiciones de entornos exteriores. La gran dificultad de la detección y el reconocimiento de la señalización estriba en los cambios de iluminación y en la gran variedad de clases posibles.

2. Recopilar una base de datos a partir de las secuencias proporcionadas por un sistema de captura, que posibilite el testeo del módulo para la cuantificación de resultados.
3. Estudio de diferentes propuestas sobre técnicas de procesado, descriptores y arquitecturas de clasificación para el reconocimiento de señalización vial.
4. Construir algoritmos eficaces y fiables de clasificación, para lo cual se intentará optimizar el tiempo de procesado, a la vez que se maximizará la tasa de acierto de reconocimiento y se minimizará la tasa de falsa alarma.

Finalmente, el módulo de reconocimiento de señalización se integrará en la arquitectura modular del sistema completo RESET (Sistema de Reconocimiento de Señales de Tráfico) desarrollado por el grupo de investigación GRAM (*GRAM-Multisensorial Analysis and Recognition Group*) de la Universidad de Alcalá.

### 1.5.2. Estructura de la tesis

El documento se ha estructurado de la siguiente manera:

- El capítulo 2 recoge el estado del arte de los trabajos que pueden encontrarse en la literatura para el reconocimiento de señalización vertical. Se describen las líneas de trabajo más significativas desarrolladas hasta la fecha, poniendo de relieve los puntos débiles y fuertes de cada uno de ellos, así como las perspectivas para el futuro.
- El capítulo 3 describe la base de datos anotada, la cual ha sido creada para testeo del sistema y entrenamiento en técnicas de aprendizaje supervisado.
- El capítulo 4 incluye los algoritmos utilizados en la implementación de un sistema completo para detección, reconocimiento y seguimiento, demostrando que la etapa de reconocimiento del pictograma supone un ‘cuello de botella’ en la carga computacional del sistema.
- El capítulo 5 realiza un estudio de optimización de diferentes arquitecturas basadas en la técnica de Máquinas de Vectores Soporte, evaluando su eficacia y su fiabilidad.
- El capítulo 6 propone un novedoso algoritmo optimizado para la identificación del pictograma con muy baja carga computacional.

- Finalmente, en el capítulo 7 se resumen las principales aportaciones originales de esta tesis y se presentan las líneas futuras de investigación.

# Capítulo 2

## Detección de Señales de Tráfico: Estado del Arte y Perspectivas

La señalización vial cumple la función de informar, prevenir y orientar a los usuarios de la vía. Su importancia se acrecienta por el hecho de que los desplazamientos en carretera implican situaciones novedosas de las que se debe informar al conductor para poder adecuar su respuesta. Con el objetivo de evitar situaciones peligrosas, la señalización no debe exigir esfuerzos de localización, ni de atención ni de comprensión, siendo necesario que el usuario sea capaz de interpretar de manera inmediata la información.

### 2.1. Características principales

Un principio básico a la hora de transmitir información visual es intentar que los medios se adapten lo mejor posible a las características anatómico-fisiológicas del ojo humano. Existen tres características principales que permiten al conductor extraer información de las señales viales: el color, la forma y el pictograma. Mediante la combinación de color y forma se define el tipo de mensaje de la señal (peligro, advertencia, prohibición, etc.) (ver tabla 2.1), mientras que el pictograma permite matizar su significado exacto.

<b>Color y Forma</b>	<b>Tipos de señales</b>
Rojo Circular	Prohibición
Rojo Triangular	Advertencia de peligro
Azul Circular	Obligación
Azul Rectangular	Indicación-Servicio-Carriles
Amarillo	Obras
Blanco	Fin de prohibición

Tabla 2.1: Clasificación de las señales de tráfico según el color y la forma.

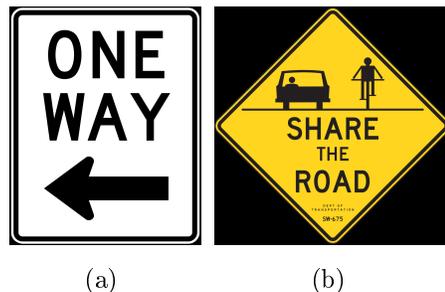


Figura 2.1: Ejemplos de señales americanas con información textual.

La correcta ubicación de las señales logrará, junto a la rigurosa aplicación de las normas de diseño, su correcta visibilidad. Dependiendo de la distancia a la que deban ser observadas, se deberá variar tanto la distancia a la que se sitúan de la calzada como sus dimensiones.

La diversidad de significados de los diferentes tipos de señales de tráfico, combinada con la variedad de contextos y circunstancias en que éstas aparecen, es muy amplia. Diversos estudios realizados por la Dirección General de Tráfico (DGT) ponen de manifiesto que las mismas señales producen diferentes efectos según el lugar en que se encuentran situadas [Trafico00].

### 2.1.1. Reglamentación internacional

La estandarización vial se desarrolló inicialmente en EE.UU. de forma separada en dos documentos: uno para las vías rurales (1927) y otro para entornos urbanos (1929). A partir de ambos surgieron dos organismos diferentes: *American Association of State Highway and Transportation Officials* y *National Conference on Street and Highway Safety*. Finalmente, mediante la publicación en 1935 del Manual sobre dispositivos de Regulación de Tráfico (*Manual on Uniform Traffic Control Devices*) quedaron fijados los criterios principales que se emplean actualmente.

El sistema de señalización estándar vigente en EE.UU., Canadá y Oceanía contiene un elevado número de señales que transmiten la información al conductor de forma textual (ver figura 2.2). Emplear señales de tráfico con estas características puede producir malinterpretaciones en caso de no conocer apropiadamente el idioma local, con la peligrosidad que ello puede conllevar. Este hecho se acrecienta cuando en un mismo país se emplean diferentes lenguas oficiales o cooficiales en función de la región, tal como sucede en EE.UU. donde pueden verse señales de tráfico en inglés, francés, castellano e incluso en algunos idiomas indígenas. En las figuras 2.2(a) y 2.2(b) se muestran dos señales canadienses con un mensaje en inglés y otro en francés, respectivamente, mientras que en la figura 2.2(c) se ilustra una señal de la reserva india de Bonaparte en Columbia Británica (Canadá) con utilización de la lengua minoritaria *Shuswap*.



Figura 2.2: Ejemplos de señales con textos informativos en varios idiomas.

A nivel europeo, el tratado de Viena de 1968 (*Convención de Viena sobre Circulación Vial y Señalización Vial*) generó una normativa común que facilitase la circulación de vehículos e introdujese mejoras en temas de seguridad vial. Una parte importante de dicho tratado fue la relativa a la señalización, donde quedaron definidas las señales y símbolos destinados a regular el tráfico. Como resultado, las señales viales en la parte occidental de Europa quedaron estandarizadas, con la salvedad de algunos países que mantuvieron su propia normativa.

El principio básico del estándar europeo consiste en emplear formas y colores para dotar de significado a la señal y poder organizar el amplio conjunto de las mismas en diferentes subgrupos. La forma debe ser circular, triangular, octogonal y rectangular, mientras que el color debe ser tal que sea fácilmente perceptible por un conductor a larga distancia, evitando en la medida de lo posible confusiones con el fondo de la escena. En función de la capacidad y limitaciones del sistema visual humano se fijaron los siguientes colores: rojo, amarillo, azul y blanco, aunque el negro se utiliza también para los símbolos del interior de las señales.

La forma triangular se utiliza para señales de advertencia de peligro. Adicionalmente, se permite emplear un rombo con el mismo propósito, si bien no es muy común en Europa. Las señales de reglamentación son, generalmente, circulares salvo las de prioridad, que tienen formas especiales dada su importancia sobre el resto, e indican obligaciones, limitaciones o prohibiciones. Se subdividen en señales de prioridad, señales de prohibición de entrada, señales de restricción de paso, señales de obligación y señales de fin de prohibición. Finalmente, las señales de indicación presentan forma rectangular e informan al conductor sobre aspectos de interés, mientras que las señales de dirección no se encuentran reguladas y aparecen con diferencias sustanciales en Europa respecto a criterios de tipografía, tipo de flechas y esquema de colores.

Pese a disponer de una regulación común europea, cada país ha adoptado variantes dentro de sus fronteras. Como un ejemplo, la figura 2.3 muestra la señal ‘Peligro por posible cruce de niños’ con variaciones del pictograma en diferentes países del continente europeo.



Figura 2.3: Variantes europeas del pictograma de una misma señal.

### 2.1.2. Reglamentación nacional

La Dirección General de Carreteras de España publicó el 25 de julio de 1962 la Orden Circular 8.1-I.C en la que se establecen directrices visuales aplicables a los signos tipográficos presentes en las señales de tráfico. Desde entonces, se han editado otras regulaciones para introducir nuevas señales y modificar criterios que recogen los cambios necesarios para coordinarse con los acuerdos de la Convención de Viena de 1968.

El vigente Catálogo Oficial de Señales de Circulación en España forma parte de la regulación básica establecida por la Dirección General de Carreteras del Ministerio de Fomento y está integrado por los siguientes documentos:

- Norma de carreteras 8.1-I.C Señalización vertical.
- Norma de carreteras 8.2-I.C Marcas viales.
- Norma de carreteras 8.3-I.C Señalización de obras.
- Catálogo de señales verticales de circulación tomos I y II.

A su vez el Catálogo Oficial de Señales de Tráfico se encuentra recogido en el Reglamento General de Circulación, el cual fue aprobado por el Real Decreto 1428/2003(BOE núm.306, 23 de diciembre de 2003) [Boe03] y modificado por el Real Decreto 965/2006 (BOE núm.212, 5 de septiembre de 2006) [Boe06], sustituyendo al Real Decreto 13/1992.

A continuación, se recoge el objetivo de las diferentes categorías de señales viales, las cuales son mostradas en el Apéndice C.

#### Señales de advertencia de peligro

##### *Artículo 149 del Reglamento General de Circulación*

Tienen por objeto informar a los usuarios de la vía de la proximidad y naturaleza de un peligro difícil de ser percibido a tiempo, con objeto de que se cumplan las normas de actuación que, en cada caso, sean procedentes.

La distancia entre la señal y el principio del tramo peligroso podrá indicarse en un panel complementario del modelo recogido en el Catálogo Oficial de Señales de Circulación.

## Señales de reglamentación

### *Artículo 150 del Reglamento General de Circulación*

Las señales de reglamentación tienen por objeto informar a los usuarios de la vía sobre las obligaciones, limitaciones o prohibiciones especiales que deben observar.

Se subdividen en:

- *Señales de prioridad*  
Destinadas a poner en conocimiento las reglas específicas de prioridad en las intersecciones o en los pasos estrechos.
- *Señales de prohibición de entrada*  
Para quienes se las encuentren de frente en el sentido de su marcha y a partir del lugar en que están situadas, prohíben el acceso a los vehículos o usuarios en la forma que señalan.
- *Otras señales de prohibición o restricción*  
Regulan el comportamiento de un vehículo en circulación mediante la prohibición de ciertas acciones o la imposición de límites concretos.
- *Señales de obligación*  
Son aquellas que señalan una norma de circulación de obligado cumplimiento.
- *Señales de fin de prohibición o restricción*  
Acotan el tramo de vía en que se debe cumplir una prohibición o restricción. Sirven para anular el cumplimiento de una señal de prohibición o restricción.

Las obligaciones, limitaciones o prohibiciones especiales establecidas por las señales de reglamentación son aplicables a partir del lugar donde estén colocadas, salvo que mediante un panel complementario debajo de ellas se indique la distancia a la sección donde empiecen a regir.

### 2.1.3. Señales de indicación

#### *Artículo 158 del Reglamento General de Circulación*

Las señales de indicación tienen por objeto facilitar al usuario de la vía ciertas indicaciones útiles. Se desglosan en los siguientes tipos:

- *Señales de indicaciones generales.*  
Brindan información útil a los usuarios.
- *Señales de carriles.*  
Indican una reglamentación especial para uno o más carriles de la calzada.

- *Señales de servicio.*  
Informan de un servicio de posible utilidad.
- *Señales de orientación.*  
Se subdividen a su vez en señales de preseñalización, señales de dirección, señales de identificación de carreteras, señales de localización, señales de confirmación y señales de uso específico en poblado.
- *Paneles complementarios y otras señales.*  
Precisan el significado de la señal que complementan.

Los paneles complementarios situados debajo de una señal de indicación pueden expresar la distancia entre dicha señal y el lugar señalado.

## 2.2. Descripción del Entorno de Trabajo

El diseño de un sistema para la detección y el reconocimiento de señales de tráfico mediante técnicas de visión artificial no es una tarea trivial. Debido a la ubicación de las señales en entornos exteriores, éstas aparecen en entornos complejos (con gran cantidad de elementos) y no-estructurados (los elementos en la escena no ocupan un lugar determinado en ella, sino que aparecen de forma aleatoria y no controlada). Además, las condiciones de iluminación son cambiantes y difícilmente controlables. Así, la fiabilidad de un sistema automático para detección y reconocimiento de señalización viene dada por su robustez frente a las dificultades que se enuncian a continuación:

1. Variaciones de luminosidad (ver figura 2.4). La luz en entornos naturales es difícil de controlar pues cambia constantemente de intensidad, dirección y color en función de la hora del día, de las estaciones del año y de las condiciones meteorológicas. Las variaciones en las condiciones de iluminación se traducen en cambios en la percepción del color que son complicados de controlar de manera automática. Asimismo, se pueden producir sobre las señales puntos de brillo debido a la reflexión de la luz solar y sombras proyectadas por otros objetos de la escena (ver figura 2.5).
2. Oclusiones (ver figura 2.6). De forma generalizada, las señales aparecen en la escena junto a múltiples objetos, tanto estáticos como dinámicos. Cualquiera de estos objetos, tales como árboles, vehículos, edificios y peatones, pueden interponerse entre la cámara y la señal provocando una oclusión. Se pueden establecer dos niveles de oclusión: oclusión parcial, en el caso de que el objeto interferente oculte la visibilidad de una región de la señal y oclusión total, cuando el objeto ocluye por completo la visibilidad de la señal.
3. Deformaciones proyectivas (ver figura 2.7). Idealmente, las señales se colocan de forma perpendicular a la trayectoria de los vehículos para garantizar su correcta

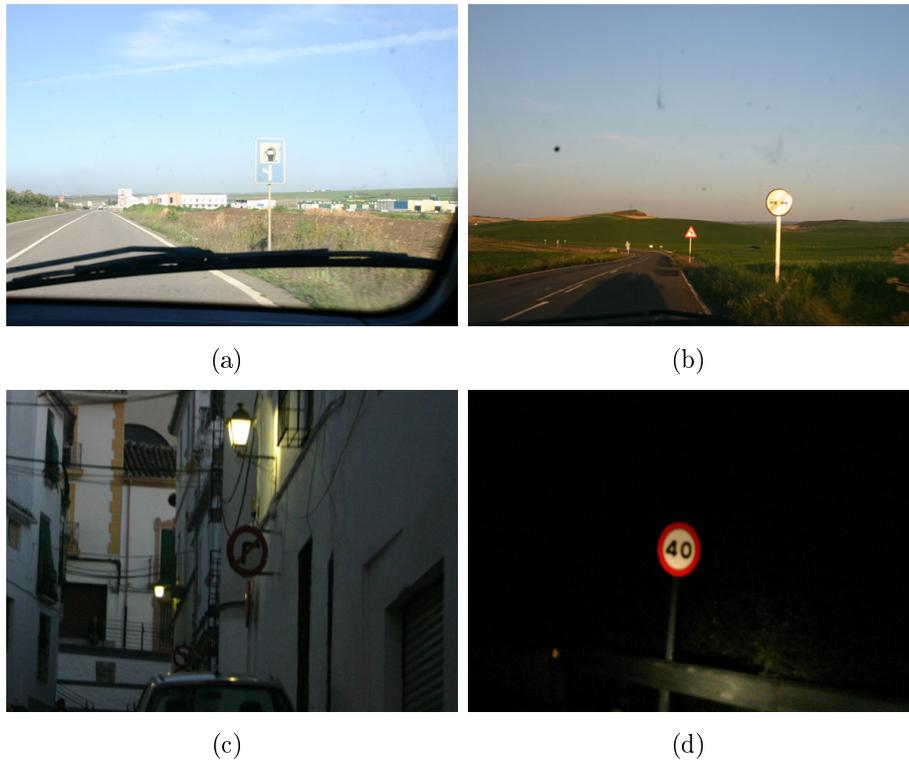


Figura 2.4: Cambios de iluminación en entornos viales. (a) Día. (b) Puesta de sol. (c) Anochecer. (d) Noche.



Figura 2.5: Brillos y sombras producidos sobre las señales. (a) Sombra proyectada. (b) Puntos de brillo.

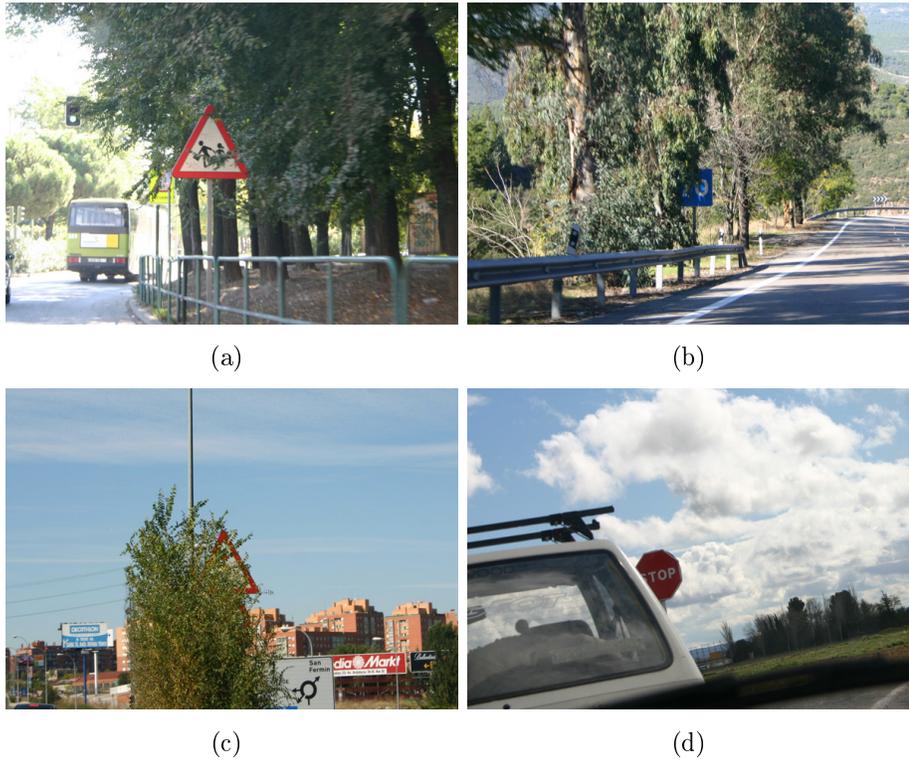


Figura 2.6: Señales con diferentes niveles de oclusión.

visibilidad. Sin embargo, en entornos reales no se cumple siempre esta premisa, debido a la influencia de múltiples factores que pueden provocar alteraciones sobre la orientación de las señales. A efectos prácticos, es posible distinguir dos tipos de rotaciones diferentes.

- Rotaciones 2D: La señal se encuentra en un plano perpendicular a la trayectoria del vehículo, pero rotada un determinado ángulo respecto a la situación ideal. Circunstancias tales como el deterioro de la señal, inclinación del firme de la vía o actos vandálicos pueden provocar situaciones como ésta (ver figura 2.7(a)).
- Rotaciones 3D: En este caso, el plano que contiene la señal no es perpendicular a la trayectoria del vehículo. Es decir, el ángulo entre el eje óptico de la cámara y el vector perpendicular al plano de la superficie de la señal no es nulo. Generalmente, este tipo de rotación se produce cuando las señales se colocan a varios metros de distancia de los márgenes delimitadores de la vía y en intersecciones giratorias. Llevar a cabo un correcto tratamiento de este tipo de rotaciones supone aplicar técnicas de proyección geométrica para tratar de recuperar la forma original de la señal. Sin embargo, es posible que el sistema de reconocimiento funcione de forma incorrecta identificando señales correspondientes a otra vía diferente a la de circulación del vehículo. En la figura

2.7(b) se presenta esta situación, donde la señal ‘Ceda el paso’ es aplicable a los vehículos de la vía de incorporación a la intersección giratoria.



Figura 2.7: Señales con deformaciones proyectivas. (a) Rotación 2D. (b) Rotación 3D.

4. Envejecimiento de las señales. Con el paso del tiempo, las señales pierden intensidad en su coloración debido a la incidencia de la luz solar, la lluvia y el polvo sobre los materiales de las mismas. Asimismo, este efecto de envejecimiento provoca como consecuencia que las señales sean menos distinguibles del entorno en que se encuentran (ver figura 2.8(a)).
5. Señales alteradas y dañadas. La acción humana provoca en algunas situaciones alteraciones y daños que distorsionan la forma o apariencia de las señales, ya sea por colisiones (ver figura 2.8(b)) o por actos vandálicos como *graffities* (ver figura 2.8(c)).
6. Señales difusas. La captura de imágenes de video en movimiento mediante un sistema de captura instalado en un vehículo puede verse distorsionada en ocasiones debido al movimiento y a la vibración. Este efecto se conoce en la literatura como *blurred images*.
7. Presencia de objetos con colores y formas similares a las señales. A veces, aparecen en la escena edificios, postes publicitarios o vehículos con colores y formas similares a las señales viales. La figura 2.9 muestra algunos ejemplos.

## 2.3. Revisión de la literatura

Los sistemas automáticos para detección y reconocimiento de señales de tráfico (*TSDRS-Traffic Sign Detection and Recognition System*) basados en técnicas de visión artificial incluyen, por lo general, dos etapas: una para detección, que extrae regiones de interés (*ROIs-Regions of interest*), y otra de reconocimiento, que identifica el prototipo de señal que se corresponde con la ROI. No obstante, en muchos sistemas se incluye entre ambas

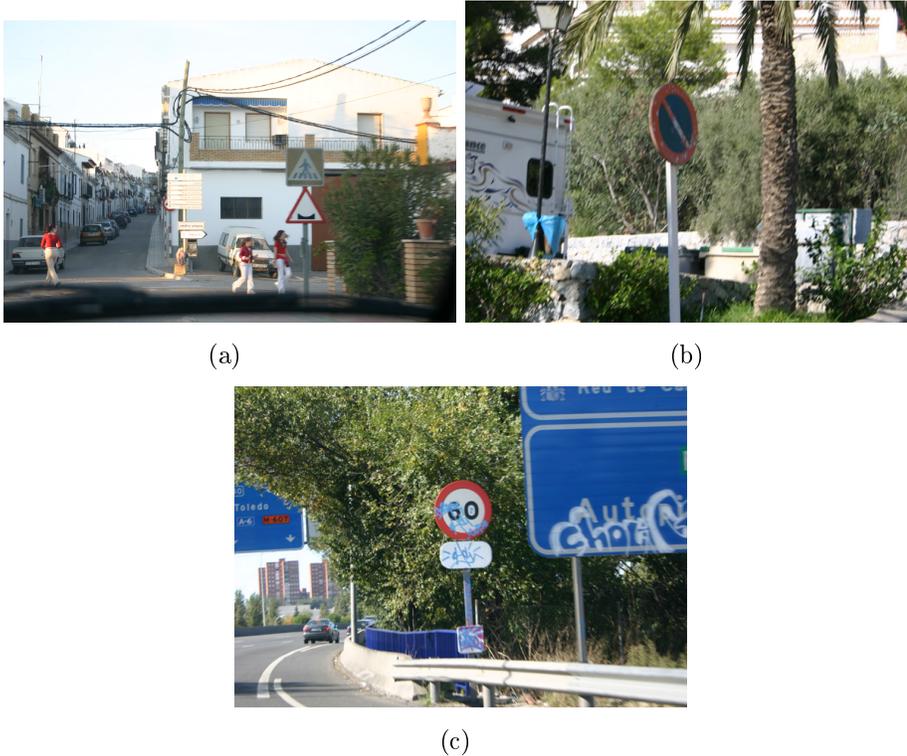


Figura 2.8: Señales con diferentes causas de deterioro. (a) Envejecimiento. (b) Deformación del material. (c) *Graffity*.

etapas un módulo auxiliar para clasificar las ROIs de acuerdo a su forma geométrica. La ventaja de clasificar los objetos candidatos por su geometría es doble: en primer lugar, ayuda a descartar posibles falsas alarmas y por otra, reduce el número de clases a discriminar en la etapa de reconocimiento de acuerdo a la forma geométrica de la ROI. Esto queda claro si se tiene en cuenta que tan sólo son posibles ciertas combinaciones de color y forma. Así, por ejemplo, no existen señales triangulares de color azul.

Los primeros trabajos sobre detección automática de señales datan de la década de los años sesenta, aunque los primeros avances significativos son de 1984 [Paalik99], cuando el campo de visión artificial se introdujo en la asistencia a la conducción. Desde entonces y hasta el momento, se han desarrollado múltiples algoritmos que han ido mejorando en fiabilidad y eficacia.

Esta sección cubre la revisión del estado del arte en este campo. La revisión es dividida en tres partes, de acuerdo a la detección, a la clasificación y al reconocimiento.

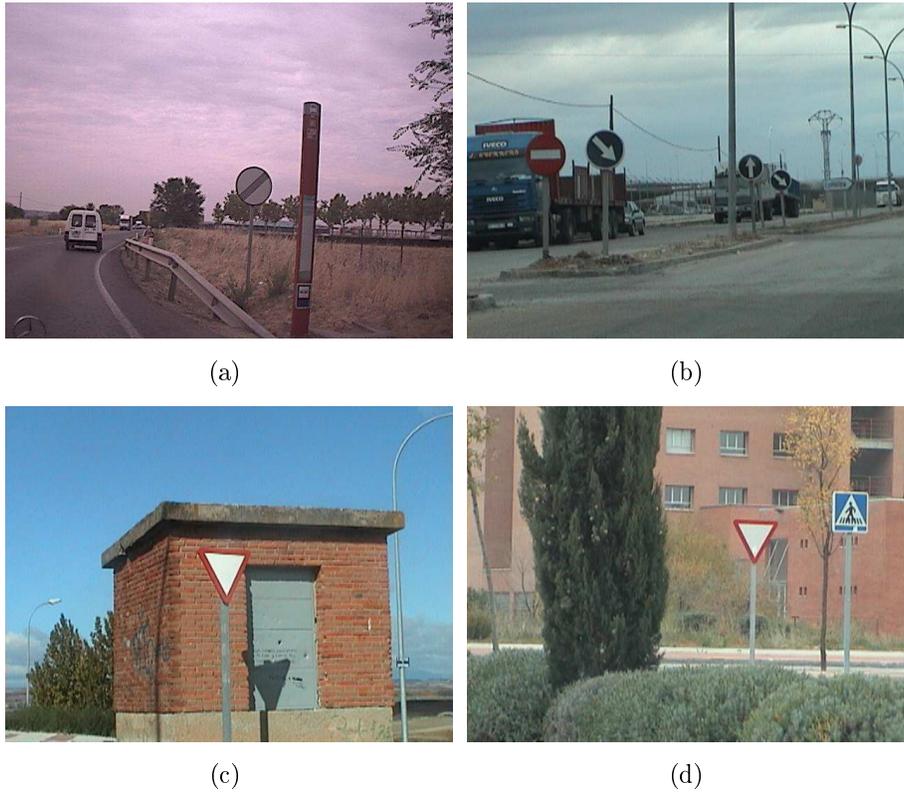


Figura 2.9: Señales con colores similares a los de los objetos adyacentes.

### 2.3.1. Detección de Señales

Aprovechando las características más distinguibles de las señales de tráfico, la mayor parte de los TSDRSs basados en visión artificial emplean algoritmos que utilizan el color y los bordes como criterios de búsqueda.

#### Segmentación por color

El color es, probablemente, la característica visual más significativa de una señal de tráfico. Si bien, de forma generalizada, suelen emplearse los colores primarios de luz (rojo, verde y azul) o secundarios (amarillo), la normativa permite que la coloración de las señales se encuentre dentro de un determinado margen de tolerancia, de acuerdo a la distribución espectral de colorimetría bajo iluminación diurna [Judd64]. Para ello se especifica un rango de longitudes de onda del espectro visible que no debe ser rebasado en ningún caso. La segmentación de imágenes basada en color realiza una búsqueda de regiones de acuerdo a los colores definidos para las señales viales. Es importante considerar la dificultad de trabajar en entornos exteriores, ya que las variaciones luminosas se traducen en cambios del color aparente de los objetos. Teniendo en cuenta que la percepción del color de un objeto depende de la fuente luminosa (color, intensidad y posición), de las propiedades

de reflexión del objeto y de las propiedades de la cámara, la mayoría de las técnicas basadas en color presentan problemas cuando la fuente de iluminación es cambiante. Este es el principal motivo que ha desencadenado la propuesta de diferentes algoritmos [Skarbek94, Busin08], desde aquellos que utilizan espacios de color estándar hasta los que crean espacios de color propios. El objetivo es buscar un método de segmentación por color que sea robusto a los cambios de iluminación. Entre los espacios de color utilizados en la literatura, encontramos:

- **RGB-(*Red-Green-Blue*):**

El espacio RGB es el más directo en cuanto que es utilizado por las cámaras de captura y está basado en la teoría tricromática que asume que cualquier color puede obtenerse por mezcla de los tres colores primarios. Por analogía con el ojo humano, los colores primarios utilizados son el rojo (R), el verde (G) y el azul (B). Así, el color de cada píxel es definido por los tres canales correspondientes a los colores primarios y una primera aproximación para la segmentación de señales de tráfico en la escena consiste en emplear directamente el espacio RGB. Así, en [Zadeh97] se buscaron subespacios del espacio RGB que encerrasen las variaciones de cada color de las señales. Estos subespacios fueron definidos en un proceso de entrenamiento para determinar los rangos de color. En [Ruta09] se construyeron mapas binarios para los canales R y B que evalúan la diferencia entre cada una de estas dos componentes y las otras componentes restantes del espacio RGB. Debido a que el color amarillo se encuentra presente tanto en el rojo como en el verde, pero no en el azul, para el canal amarillo se evaluó la distancia R-B y G-B.

Existen otras propuestas que, en vez de trabajar directamente con el espacio RGB, incluyen modificaciones sobre el mismo. Tal es el caso de [Escalera97], donde se empleó una medida de la relación entre componentes RGB para umbralizar la escena por color. De hecho, el problema de efectuar la umbralización directamente sobre el espacio de color RGB presenta el inconveniente de que las tres componentes se encuentran altamente correladas y no son independientes entre ellas, tal como quedó demostrado en el análisis de componentes principales de [Ohta80]. Para minimizar la sensibilidad del sistema ante variaciones de luminosidad, diversos investigadores propusieron normalizar el espacio de coordenadas RGB respecto a la intensidad [Kamada90, Janssen93, Perez94], generando un nuevo espacio de color conocido como NRGB (*Normalized RGB*).

La utilización de umbrales fijos en la segmentación acarrea, a veces, problemas de mala generalización en cuanto que los resultados son muy dependientes de las características de las imágenes a procesar. Como el objetivo de la segmentación es el etiquetado de los píxeles de una imagen, otros investigadores han utilizado clasificadores estadísticos. Entre los mismos, cabe destacar el trabajo de [Priese93], en el que se utilizaron redes neuronales. Con el objetivo de acelerar el proceso de etiquetado, algunos investigadores han recurrido al empleo de tablas de búsqueda

(*LUT-Look-up-tables*) que asocian *off-line* todas las posibles combinaciones de valores R,G,B con su correspondiente etiqueta de clasificación, consiguiendo reducir el coste de la clasificación de cada píxel a un simple acceso a la tabla. Entre los trabajos que han hecho uso de LUTs se encuentran [Akatsuka87, Janssen93, Ghica95]. En [Akatsuka87, Janssen93] se empleó el espacio de color RGB normalizado y en [Ghica95] se hizo uso de las componentes RGB.

Otra variante de uso del espacio RGB puede ser encontrada en [Kellmeyer94], donde se consideró que en la percepción del color existe una influencia del entorno. Por este motivo, se clasificaron los píxeles a través de una red neuronal en la que las entradas son alimentadas por las características de color correspondientes tanto al píxel como a los píxeles vecinos.

Finalmente, en [Tsai08], se utilizó la transformación de Karhunen-Loeve a partir del espacio RGB. Para ello, se obtuvo la matriz de covarianza formada con la distribución de los tres canales R, G y B a partir de las muestras de entrenamiento. En base a los autovectores correspondientes a los mayores autovalores, se mapeó el problema a un nuevo espacio de color.

- **HSI-*Hue-Saturation-Intensity*:**

El espacio de color HSI tiene tres componentes, dos de las cuales: el tono o matiz(H) y la saturación(S), están relacionadas con la percepción del color, mientras que la componente de intensidad(I) está vinculada al brillo. La componente de tono representa el color dominante y la saturación el grado de pureza del color, de tal modo que los valores de saturación altos corresponden a colores muy puros y los valores bajos a colores con una gran mezcla de blanco. Las componentes HSI pueden obtenerse a partir de las componentes RGB [Gonzalez93] mediante transformación, aunque las expresiones de conversión entre ambos modelos no son lineales y por tanto, existe un coste computacional asociado a la transformación. El espacio HSI presenta dos ventajas que hacen que sea el modelo de color más utilizado en la segmentación: 1) el espacio HSI presenta gran similitud con el comportamiento del sistema visual humano, y 2) la información de color queda codificada en dos componentes (H y S).

El trabajo de Kehtarnavaz [Kehtarnavaz93] fue uno de los pioneros en la segmentación de señales viales utilizando el espacio HSI. En concreto, realizó un mapeo del espacio RGB al espacio HSI y a partir del análisis estadístico de muestras, se determinaron los umbrales de las componentes HSI para la segmentación de píxeles rojos. En la misma línea, en [Nguwi06] se fijaron umbrales para las componentes HSI empleando un conjunto mayor de muestras.

Aprovechando que el color queda codificado en tan sólo dos componentes del espacio HSI, la mayor parte de la literatura trabaja únicamente con las componentes H y S. Así, [Arnoul96] utilizó ambas bandas para extraer los píxeles azules y rojos de la imagen. En [Escalera03] se empleó una transformación no lineal sobre las

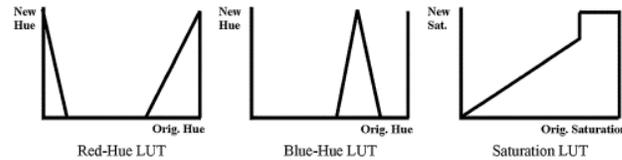


Figura 2.10: Clasificación de color mediante LUT en [Escalera03].

componentes de matiz y saturación y se obtuvo una máscara binaria aplicando la operación lógica *AND* sobre ambas componentes. La transformación se realizó para corregir los posibles errores de segmentación generados al trabajar con una umbralización fija sobre las componentes por separado. Por cada color a segmentar, la transformación se implementó mediante dos LUTs, correspondientes al matiz y a la saturación (ver figura 2.10).

En [Fleyeh06] se hizo un estudio sobre la sensibilidad de los espacios RGB, NRGB y HSI a las sombras y los reflejos, demostrándose que la componente H consigue la mayor invarianza, aunque las componentes S e I se utilizaron para definir el subespacio cromático. La imagen segmentada fue dividida en sub-imágenes y en cada una de ellas se determinaron las posibles semillas de un posterior algoritmo basado en crecimiento de regiones.

En [Piccioli96] se trabajó de forma estructural, en vez de píxel a píxel. Para ello se determinaron las ROIs de la imagen en las que el porcentaje de píxeles en un determinado rango de la componente H es mayor que un determinado umbral.

En cualquier caso, el espacio HSI admite pequeñas variantes que definen otros espacios alternativos como el HSV (*Hue-Saturation-Value*). La componente V del espacio HSV y la componente I del HSI representan casi lo mismo. Mientras que en HSV, un valor máximo de V significa que ese es el color más brillante, en HSI un valor máximo de I significa que el color es blanco, independientemente de los valores de H y S. En concreto, el espacio HSV fue utilizado para la segmentación en [Vitabile01, Vitabile02, Malik07]. En los dos primeros trabajos se trató la inutilización de la componente H cuando la intensidad(I) y/o la saturación(S) presentan valores muy altos o muy bajos. Por estos motivos, se definieron tres áreas en el espacio de color HSV mediante umbralización de las componentes S y V: área acromática, área cromática inestable y área cromática. También en [Paclik00] se empleó el espacio HSV separando el blanco, el negro, el rojo, el azul, el verde y el amarillo, considerando que para la segmentación de colores acromáticos se realizó una umbralización de la componente V.

- **Otros espacios de color:**

Aunque la mayor parte de los estudios de segmentación basados en color para la detección de señales viales se fundamentan en los espacios RGB y HSI, existen otras alternativas. El modelo CIECAM97, recomendado por la CIE (*Commission Internationale de l'Éclairage*), permite medir el color aparante de forma invariante a las condiciones de luminosidad y extraer características cromáticas de las señales [Fairchild01]. Este modelo está basado en la emulación del sistema visual humano extrayendo del color las características de luminosidad, cromaticidad y matiz. Para ello, intrínsecamente, el modelo realiza la transformación desde el espacio RGB al CIE XYZ y posteriormente al LCH (Iluminación, Cromaticidad y Matiz). Así, en [Gao06] se estimaron los rangos que presentan las componentes de cromaticidad y matiz de las señales rojas y azules, considerando diferentes condiciones de iluminación. El proceso de segmentación dividió recursivamente la imagen hasta obtener bloques homogéneos.

En [Tominoga90] se empleó el espacio de color uniforme  $L^*a^*b$  ( $L^*a^*b$  uniform color space). Este modelo fue propuesto por la CIE para poder determinar la diferencia perceptual entre colores cuantificándola mediante la distancia euclídea. Básicamente, consta de tres canales:  $a^*$  (diferencias rojo-verde),  $b^*$  (diferencias azul-amarillo) y  $L^*$  (luminosidad del color). También Siogkas [Siogkas09] utilizó el espacio  $L^*a^*b$  para efectuar a posteriori una segmentación basada en el color. El empleo del algoritmo de Otsu permite la obtención de cinco imágenes binarias resultantes de las posibles combinaciones positivo-negativas de los canales  $a^*$  y  $b^*$  y de la umbralización del canal  $L$ . A partir de la comparación de las imágenes binarias resultantes de la umbralización con la imagen binaria del canal de luminosidad es posible obtener un conjunto de regiones de interés.

Shadeed [Shadeed03] propuso un algoritmo de segmentación basado en dos espacios de color: HSV e YUV, incluyendo este último una componente de luminancia (Y) y dos de cromaticidad (UV). La imagen RGB fue mapeada en los espacios HSV e YUV, aplicando una umbralización por niveles a las componentes cromáticas H y UV. Los resultados de sendas umbralizaciones fueron fusionados mediante la operación lógica *AND*.

Zhang [Zhang08] propuso el empleo del espacio de color CBH (*Color Barycenters Hexagon*), que en términos generales consiste en realizar una correspondencia entre un triángulo cuyos baricentros representan las tres componentes del espacio de color RGB y un hexágono formado por seis regiones de color (Rojo, Verde, Azul, Magenta, Amarillo y Cyan). Mediante la representación de cada uno de los píxeles de la imagen en dicho hexágono se construye un histograma que sirve para filtrar aquellos píxeles cuyo color no sea de interés.



Figura 2.11: Imagen binaria obtenida en [Aoyagi96] tras aplicar un filtro laplaciano a la imagen para la obtención de bordes.

En [Eichner08] se usó el espacio YCrCb, trabajando con el canal Cr mediante umbralizado. El espacio YCrCb se define mediante dos componentes de crominancia: Cr y Cb, a través de rojo y azul, respectivamente, e Y como componente de luminancia.

### Segmentación por bordes

Aunque el color es una fuente muy importante de información para detectar las señales, ningún espacio cromático es totalmente inmune a factores como los cambios de luz, el envejecimiento de material o la distancia de observación. Por esta razón, algunos investigadores argumentan que una segmentación basada en color no es totalmente fiable y prefieren trabajar con imágenes en escala de gris.

De esta manera, en [Aoyagi96] se usó la información estructural para separar los objetos del fondo. El objeto fue extraído del fondo de la escena con un filtro laplaciano. Para obtener la imagen binaria se realizó una umbralización (ver figura 2.11) y la detección corrió a cargo de algoritmos genéticos. Las imágenes en escala de gris también fueron usadas con un detector de Canny en [Garcia06] y en [Piccioli94], donde en este último trabajo se realizó un agrupamiento de los bordes identificados. La correlación de los bordes con máscaras circulares permitió detectar círculos, mientras que los triángulos fueron detectados por agrupamiento de bordes en segmentos verticales, horizontales y oblicuos. Un método alternativo se presentó en [Sandoval00], usándose una detección de bordes basada en la dependencia de ángulo. El método está basado en la generación de máscaras de convolución que usan la posición angular de los píxeles bajo consideración y se usa para detección de formas circulares.

En [Adorni96] se empleó un operador de gradiente para extraer los bordes, seguido de un umbralizado que elimina los gradientes de bajo valor. La aproximación está basada en la detección de forma y en su localización en la escena, basándose en un *matching* mediante conocimiento previo de la forma a detectar (ver figura 2.12). Un filtro CNN (*Cellular Neural Networks*) es utilizado para realizar la preselección de píxeles con alto valor de gradiente, los cuales son candidatos a formar parte del borde de una forma.

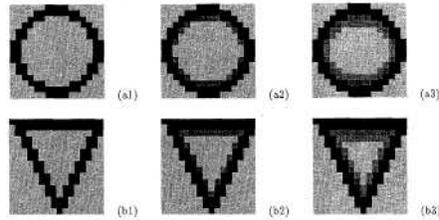


Figura 2.12: Ejemplos de formas ideales para comparar con las formas reales en [Adorni96]. Las formas originales (a1 y b1) se amplían para otras escalas añadiendo píxeles internos en (a2, b2, a3 y b3).

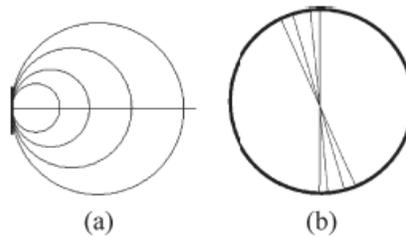


Figura 2.13: Simetría radial utilizada en [Loy04]. (a) Para el borde de un punto, los centroides de una posible familia de círculos se extienden a lo largo de la línea que es ortogonal a la dirección del borde. (b) Cada punto del borde de un círculo vota a lo largo de una línea de posibles radios y estas líneas intersectan en el centro del círculo, resultando un pico.

En [Buker95] los bordes, detectados en una estructura piramidal, fueron procesados por un algoritmo genético (*GA-genetic algorithm*) para detectar las señales de límite de velocidad, mientras que en [Gavrila98] se generaron modelos *offline* que representan los bordes de la señal teniendo en cuenta los cambios de escala. Los bordes extraídos son reforzados con una transformación de distancia (distancia de Chamfer). En [Paclik00], la detección de detección de bordes se utiliza para encontrar señales circulares y triangulares en la imagen.

Un método rápido de detección basado en bordes fue desarrollado en [Loy04] para la detección de señales. En concreto, se utilizó una extensión de la transformación de simetría radial presentada en [Loy03] para detectar polígonos regulares y círculos. El método, que es invariante a las rotaciones 2D y puede ser empleado en aplicaciones de tiempo real, se basa en considerar todos los posibles círculos de los que puede formar parte un píxel del borde (ver figura 2.13). En la misma línea, en [Barnes08] se adaptó la detección de formas mediante la transformada de simetría radial para las formas circular, triangular, cuadrada, octogonal y rómbica.

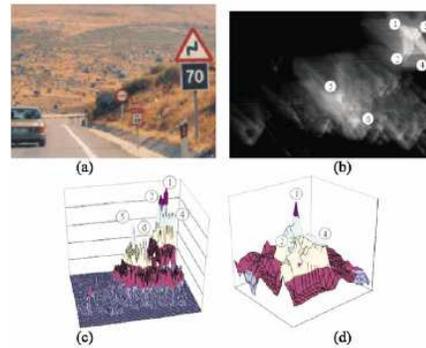


Figura 2.14: Ejemplo de búsqueda de señales triangulares mediante GA en [Escalera03]. (a) Imagen Original. (b) Función de ajuste para una señal triangular. (c) Mapa de elevación de (b). (d) Mapa detallado de los máximos.

### Segmentación por bordes y color

Con el objetivo de reforzar la fiabilidad de la detección, algunos investigadores han combinado el color y la forma tomando ventaja de ambas técnicas. No obstante, un método híbrido permite llamar a una u otra técnica dependiendo de las circunstancias. El uso combinado de color y forma tiene la ventaja de utilizar la información disponible de ambos criterios. Tal es el caso de [Fang03-b], que desarrolló un sistema de detección con características de forma y color mediante una red neuronal y la integración de ambas se realizó a través de un clasificador borroso. Para mejorar la capacidad de detección del sistema, se añadió una etapa de integración temporal.

En [Liu02] se extrajeron los bordes con un filtro laplaciano y el color de la imagen con otro filtro SVF (*Simple Vector Filter*).

También, en [Won08] se construyó un mapa de notabilidad usando las características de color y bordes. Las ROIs fueron seleccionadas por máxima energía local utilizando una ventana deslizante de tamaño fijo.

En [Escalera03] los algoritmos genéticos fueron usados para un análisis de forma sobre los blobs detectados por color (ver figura 2.14), mientras que en [Escalera04] se utilizaron dos algoritmos para la detección de forma: algoritmos genéticos y SA (*SA-Simulated Annealing*) a partir de la energía de color, energía acromática, energía de gradiente y energía de distancia.

### Otras técnicas:

Los métodos de detección basados en la técnica de ventana deslizante, sin necesidad de un proceso previo de segmentación, han ganado popularidad en los últimos tiempos. No obstante, el análisis de regiones deslizantes a través de la imagen completa implica un alto coste computacional, ya que para encontrar el objeto hay que escanear un gran

número de regiones a diferentes escalas. En [Bahlmann05] se implementaron detectores de señales de tráfico libres de parámetros usando el algoritmo AdaBoost y una estructura en cascada de Viola-Jones [Viola01], que permite seleccionar las características de Haar más discriminatorias. Las mejores características fueron seleccionadas a partir de un conjunto de múltiples representaciones de color (planos R, G y B) y canal en escala de gris. En esta misma línea, en [Baro09] se utilizó también la metodología AdaBoost para la búsqueda de señales viales en la imagen, pero en vez de con características de Haar, empleando dipolos disociados. Anteriormente, en [Luo92] y [Kim97] ya se propuso el uso de texturas para detectar las señales en la escena. El análisis de la imagen completa demanda mucho tiempo de procesado, incluso con detectores muy ligeros desde el punto de vista computacional, tales como los basados en las características de Haar. En [Overett11] se proponen dos variantes de HOG aplicadas sobre una cascada de clasificadores débiles: características simples tomando los intervalos discretos del HOG de manera independiente en las regiones y linealización de características mediante el uso del análisis discriminante de Fisher. En [Rajesh11] se trabaja con el vector de coherencia de gradiente orientado en el que se toma en cuenta el número de píxeles coherentes, considerando un píxel como coherente si éste forma parte de un grupo de píxeles con la misma orientación dominante que la suya. Recientemente, varios investigadores han demostrado que el método de bolsas de palabras (*BoW-Bag of Words*) es una interesante elección para problemas de detección. En [Nguyen13] se propone acelerar la técnica BoW para la detección de señales viales mediante una búsqueda aproximada a través de descriptores SIFT, de tal manera que las regiones del fondo se eliminan del espacio de búsqueda en la primera pasada y en la búsqueda refinada se busca la localización precisa de la señal en las regiones que hayan resultado positivas en la primera etapa.

### 2.3.2. Clasificación de formas

La clasificación de formas permite diferenciar la geometría de los objetos candidatos, teniendo en cuenta que las formas geométricas que puede presentar una señal vial son las siguientes: círculo, triángulo, octógono y rectángulo.

En [Shi08] fue desarrollado un sistema de clasificación por formas utilizando Máquinas de Vectores Soporte (*SVM-Support Vector Machines*), distinguiendo entre siete diferentes formas geométricas y utilizando dos características: las imágenes binarias y los momentos de Zernike. Otros clasificadores de forma emplean los histogramas que codifican las orientaciones locales de los bordes [Gao06] (ver figura 2.15) o características tales como el perímetro, el centro de gravedad, el área y el código Freeman [Blancard92], que sirven de entrada a una red neuronal.

En [Vitabile02] se calculó una medida de similitud entre la forma de la ROI y un conjunto de formas, contemplando posibles rotaciones, mediante la distancia de Tanimoto. En [Escalera97] se definieron máscaras de ángulos específicos de acuerdo a las formas a detectar y se realizó un proceso de convolución entre las máscaras y la imagen (ver

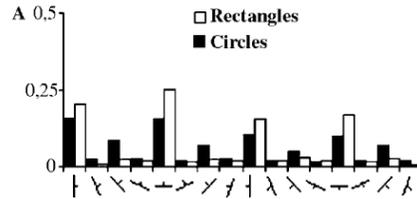


Figura 2.15: Histograma promediado de orientaciones para señales circulares y rectangulares en [Gao06]. Las señales rectangulares muestran predominio de orientaciones verticales y horizontales y las circulares, una distribución más uniforme en todas las orientaciones.

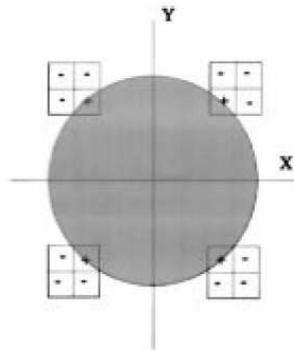


Figura 2.16: Máscara para detección de señales circulares en [Escalera97].

Figura 2.16). [Piccioli94] se basó en principios geométricos para efectuar la detección de formas circulares y triangulares e implementó un algoritmo para cada forma. Para señales triangulares, se combinó la detección de bordes empleando el algoritmo de Canny, con una aproximación poligonal y un filtrado de segmentos, conservando aquellos que eran horizontales o con una pendiente determinada. Para la detección de señales circulares se recurrió al uso de la transformada de Hough.

Algunas aproximaciones buscan explícitamente formas específicas. Tal es el caso de [Eichner08], donde se detectaron señales circulares a partir de los objetos segmentados mediante la técnica RANSAC. Esta técnica se basa en seleccionar aleatoriamente un conjunto de puntos del contorno del objeto e intenta ajustar un modelo de forma. Esta técnica es robusta a oclusiones parciales en las que el contorno queda roto. En [Shuang-dong05] se implementó un sistema de detección de señales de tráfico triangulares basado en un algoritmo de extracción de bordes mediante la transformada LOG (Laplaciana de la gaussiana) e incluyó un módulo adicional de detección de vértices. Mediante una red neuronal se estimó el ángulo de orientación en que se encontraba cada uno de los vértices y mediante la ley de los senos se determinó si se correspondían con un triángulo equilátero o no.

Besserer [Besserer93] desarrolló un sistema de detección de formas para integrarlo en un sistema real de reconocimiento de tráfico [Estable94]. Para ello utiliza un método de clasificación basado en un código similar al código cadena, que emplea la teoría de Dempster-Shafer. Adicionalmente, incorpora otros elementos tales como un detector de esquinas, un detector de formas circulares o un sistema de análisis de direcciones principales basado en el histograma.

En [Priese94] se abogó por la utilización de un modelo que predefine, mediante polígonos de 24 lados, las formas básicas de círculos, triángulos y rectángulos.

La transformada de Hough es una técnica que permite detectar formas geométricas mediante una transformación paramétrica del espacio. Como ventaja principal es que tolera discontinuidades en el contorno de las regiones pero, como contrapartida, su coste computacional y de memoria es muy alto. La transformada de Hough fue utilizada en [Kehtarnavaz93] para caracterizar la forma de los objetos y a partir de un proceso *offline* de detección local se analizaba la presencia de señales dentro de una escena.

### 2.3.3. Etapa de Reconocimiento de Señales

La etapa de reconocimiento tiene como objetivo identificar el pictograma de las señales de tráfico. En el caso de tratar con clasificadores supervisados, esta tarea implica averiguar a qué clase pertenece cada una de las regiones candidatas (ROIs) y el reconocimiento se estructura en dos sub-tareas: 1) extracción de vectores descriptores y 2) clasificación de las ROIs mediante alguna técnica de clasificación.

#### Descriptores

En la literatura existen dos principales alternativas para extraer características de una señal vial con la finalidad de identificar el pictograma: 1) obtener algunos descriptores de la imagen y usarlos como características de entrada de los clasificadores y 2) presentar directamente la imagen mediante muestreo de las intensidades de los píxeles.

- **Descriptores:**

Entre los diferentes métodos para la extracción de características, cabe citar la utilización de histogramas [Vicen07], [Hsien06] y [Simon09]. También en [Zakir10], los autores emplearon histogramas de forma basados en energía local (*LESH-Local Energy Based Shape Histogram*), teniendo en cuenta que la energía local facilita información para extraer puntos de interés de una imagen de una manera invariante a la iluminación y al ruido. En [Bargeton08] se trabajó sobre el reconocimiento de los dígitos en señales de limitación de velocidad a partir de histogramas de la imagen binarizada y el uso de los códigos cadena. En los últimos años, los descriptores basados en el paradigma de Histograma de Gradientes Orientados (*HOG-Histograms of Oriented Gradient*) han cobrado gran popularidad, tal como en [Xie09], [Zaklouta11],

[Zaklouta12] y [Overett09], donde en este último caso, cada celda HOG fue empaquetada como un píxel en un histograma y posteriormente, se realizó una proyección unidimensional del espacio de 8 dimensiones para aplicar técnicas de boosting. Otros posibles descriptores están basados en: *wavelets* [Nguwi08], donde se utilizaron las *wavelets* de Gabor; el paradigma SIFT (*Scale Invariant Transform*) [Kus08]; y la transformada de Fourier [Kang94] y [Hibi96], donde se implementó una transformación del espacio de coordenadas al plano log-polar con el objetivo de conseguir inmunidad a los cambios de escala y a la rotación (ver figura 2.17) y posteriormente, se efectuó una conversión al dominio de la frecuencia. También las características de Haar fueron utilizadas en [Bahlmann04].

Algunos trabajos aplican técnicas de procesado para reducir la dimensión de los descriptores. Tal es el caso de [Paletta02], donde se emplearon técnicas de análisis de componentes principales (PCA-*Principal Component Analysis*) para transformar el espacio imagen a otro espacio de menores dimensiones. Recientemente, en [Zaklouta11] se ha probado con un descriptor basado en la transformada de distancia, donde el valor de cada píxel es la distancia euclídea de cada píxel a su vecino más cercano que no sea nulo en la imagen binaria. De esta manera, se consigue una segmentación de los pictogramas y caracteres.

- **Muestreo de la imagen:**

Entre las propuestas que trabajan directamente con la imagen, bien en color o en niveles de gris, podemos encontrar diferentes tamaños de normalización de las ROIs. El compromiso entre una buena resolución para identificar el contenido y el coste computacional asociado, el cual es proporcional al número de características, determina el tamaño de escalado. Algunos ejemplos de tamaño son:  $50 \times 50$  píxeles [Broggi07],  $48 \times 48$  [Ciresan11],  $40 \times 40$  [Huang08],  $32 \times 32$  [Xie09, Sermanet11],  $30 \times 30$  [Kouzani07] y  $18 \times 18$  [Aoyagi96]. Algunos autores no trabajan directamente con los niveles RGB, tal como [Ciresan11] donde se trabaja con las componentes YUV. Otra variante en esta aproximación es la binarización de la imagen, en vez de presentar directamente las intensidades de los píxeles. Así, en [Kellmeyer94] los objetos candidatos fueron reescalados a un tamaño de  $10 \times 10$  píxeles y binarizados, distinguiendo los colores de interés del resto.

## Clasificadores

Múltiples son los métodos que se han empleado en la literatura para el diseño de clasificadores, buscando que sean lo más eficientes y robustos posibles. A continuación, se recogen los más empleados en el campo de las señales de tráfico:

- **Medidas de distancia:**

La técnica más simple de los sistemas basados en la apariencia consiste en medir

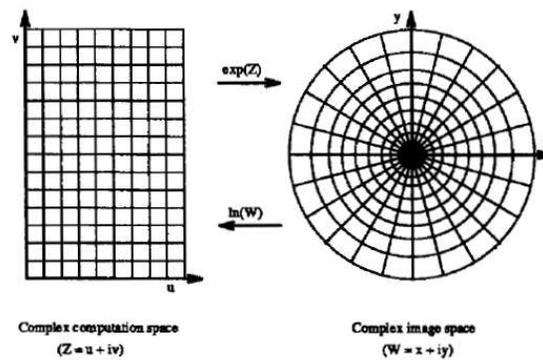


Figura 2.17: Relación entre el espacio complejo y el espacio complejo de la imagen definido en [Kang94].



Figura 2.18: Regiones locales discriminatorias para 5 clases en [Paclik06].

distancias directamente sobre el espacio imagen. Esta técnica se conoce como *template matching* y en este caso, se almacenan los prototipos de las diferentes clases en una base de datos. Durante la clasificación, cada objeto se compara con cada uno de los prototipos buscando la mayor similitud. Este método requiere que el objeto a testear esté bien alineado con los prototipos a comparar, lo cual no es frecuente debido a los problemas de segmentación.

Una de las medidas de distancia más empleada es la correlación, que al tratarse de un método de medida de similitud global, es sensible a la presencia de píxeles sin información relevante (fondo de la escena), a las oclusiones y a desalineamientos severos entre los objetos de test y los patrones de entrenamiento.

En [Escalera04] se empleó un algoritmo basado en la correlación cruzada de objetos de interés, mediante comparación con el patrón de cada clase. La correlación se realizó considerando una tolerancia a desalineamientos de hasta  $\pm 3$  píxeles.

En [Paclik06] se propuso un método consistente en la correlación sobre un conjunto de regiones discriminatorias (ver figura 2.18) para reducir el tiempo de cómputo respecto al cálculo de la correlación sobre todo el objeto. La ventaja de esta técnica es la robustez frente a oclusiones parciales. En esta misma línea, en [Ruta09] se extrajeron las regiones más discriminatorias para cada señal ponderándolas con un peso específico (ver figura 2.19).

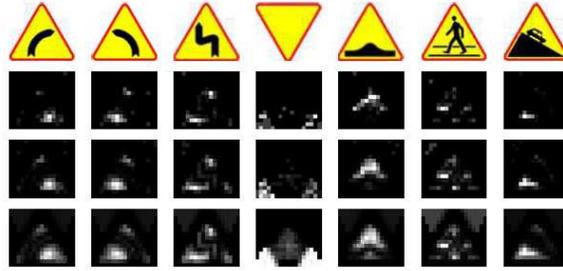


Figura 2.19: Regiones discriminatorias con diferentes umbrales en [Ruta09].

En [Akatsuka87], [Piccioli94] y [Miura00] se implementaron algoritmos de clasificación basados en la correlación cruzada normalizada con varios umbrales de ajuste. En [Malik07] se intentó conseguir cierta invarianza a las deformaciones proyectivas. Para ello, todas las muestras, tanto patrones como las de test, fueron rotadas en sentidos dextrógiro y levógiro y se calculó la correlación con las versiones rotadas de los patrones.

Una variante de la correlación se encuentra en el uso de táblas de búsqueda [Lalonde95b], donde se almacenaron los histogramas de las señales en una tabla. Mediante la comparación del histograma de la imagen de test con los histogramas almacenados en la base de datos, se clasificaron las señales de acuerdo al grado de similitud.

En [Perez02] se implementó un sistema de reconocimiento teniendo en cuenta que la correlación en el dominio espacial se convierte en una multiplicación en el espacio de Fourier. Así, en el dominio de la frecuencia se aplicaron técnicas no lineales de filtrado para aumentar la robustez ante oclusiones y rotaciones.

En [Gao06] se fijaron unos puntos sobre una rejilla polar de radios concéntricos. Para cada uno de estos puntos se obtuvieron las orientaciones locales dominantes en un entorno de vecindad y se concatenaron en un vector descriptor. La medida de similitud entre una señal de test y un patrón de referencia se realizó a través de una distancia que compara las orientaciones en todos los puntos de la rejilla. Otras distancias, tales como la de Hausdorff son utilizadas en [Nguwi08].

Si bien es posible unificar los procesos de detección y clasificación mediante el método de correlación basado en plantillas, no es práctico cuando se trabaja con imágenes de gran tamaño debido a la alta carga computacional que supone comparar cada patrón de la base de datos con todas las posibles ubicaciones de la señal dentro de la imagen. A modo de ejemplo, cabe destacar la propuesta de [Betke94], en la que se implementó un algoritmo con capacidad de detección y reconocimiento de señales de forma simultánea. En [Huang00] se emplea otro método de correlación basado en plantillas para la detección de señales en la escena, haciendo uso de la información de color. El algoritmo pone de manifiesto todas las dificultades, tales

como la lentitud relativa del sistema o la necesidad de disponer de varias muestras de cada señal con diferentes distorsiones (escalado, orientación, rotación, etc).

■ **Redes neuronales(NN-*Neural Network*):**

Las redes neuronales constituyen una de las soluciones más adoptadas en el entorno de reconocimiento de señales de tráfico debido a su potencia de clasificación. Combinan el conocimiento acerca de las clases entrenadas en una estructura basada en pesos. La tarea de entrenamiento de una nueva clase puede conllevar cambios en el número de elementos de proceso(neuronas) y capas. La función de activación de las neuronas refleja, en cierto modo, la relación entrada-salida existente en la red.

Cuando se emplean NNs en tareas de clasificación, éstas pueden ser entrenadas para reconocer características de las señales de tráfico a partir del conocimiento aportado por un conjunto de patrones. Para estudiar el comportamiento de las redes, se suele recurrir a diferentes implementaciones de diferente tamaño y comparar los resultados que generan, debido a que su capacidad de reconocimiento depende en gran medida del tipo de estructura neuronal empleada, de la estrategia de aprendizaje e incluso, del orden en que se introducen los patrones durante el proceso de aprendizaje. Los tipos de red neuronal más empleados son:

- *Red de perceptrón multicapa (MLP)*: está formada por múltiples capas, por lo que puede resolver problemas que no son linealmente separables en contraposición al perceptrón simple. El perceptrón puede ser totalmente o localmente conectado: en el primer caso, cada salida de una neurona de la capa  $i$  sirve de entrada a todas las neuronas de la capa  $i + 1$ , mientras que en el segundo caso, cada neurona de la capa  $i$  actúa como entrada de una serie de neuronas (región) de la capa  $i + 1$ .
- *Red de base radial (RBFN)*: tiene en cuenta el carácter localizado de las neuronas y emplea funciones radiales para la activación de las mismas. Calcula la salida en función de la distancia a un punto determinado y sólo genera una salida cuando la entrada se encuentra dentro del área de respuesta de las neuronas.
- *Red de retropropagación del error (BP)*: hace referencia a un algoritmo de entrenamiento, más que a una arquitectura. La retropropagación consiste en propagar el error hacia atrás, es decir, de la capa de salida hacia la capa de entrada, pasando por las capas ocultas intermedias y ajustando los pesos de las conexiones con el fin de reducir dicho error. Hay distintas versiones o reglas del algoritmo de retropropagación y distintas arquitecturas conexionistas a las que pueden ser aplicadas.
- *Red tipo ART*: se basa en principios de agrupamiento o clustering de patrones. Cuando a la red se le presenta un patrón de entrada, éste se hace resonar con

los prototipos de las categorías conocidas por la red. Si el patrón entra en resonancia con alguna clase, entonces es asociado a ésta y el centro del cluster es desplazado ligeramente para adaptarse mejor al nuevo patrón que le ha sido asignado. En caso contrario, si el patrón no entra en resonancia con ninguna clase, pueden suceder dos cosas: si la red posee una capa de salida estática entrará en saturación pues no puede crear una nueva clase para el patrón presentado pero tampoco puede asignarlo a una clase existente, mientras que si la red posee una capa de salida dinámica se creará una nueva clase para dicho patrón.

- *Red recurrente (RNN)*: es una red monocapa con habilidad para añadir *online* nuevos patrones sin necesidad de reentrenar toda la red. Durante el proceso de entrenamiento se hacen actualizaciones periódicas de normalización para mejorar la estabilidad de la red.

Pese a todo, las redes neuronales tienen una serie de inconvenientes a tener en cuenta. Entre ellos, se puede destacar la posible sobrecarga que puede suponer la imposibilidad de emplear determinadas redes multicapa en aplicaciones de tiempo real o que al disponer de una arquitectura fija, un aumento del número de clases suponga un rediseño de la arquitectura y un reentrenamiento de la red entera. Además, la existencia de mínimos locales en la función de error dificulta considerablemente el entrenamiento, pues una vez alcanzado un mínimo el entrenamiento se detiene aunque no se haya alcanzado la tasa de convergencia fijada. Cuando caemos en un mínimo local sin satisfacer el porcentaje de error permitido, se puede considerar cambiar la topología de la red (número de capas y número de neuronas), comenzar el entrenamiento con unos pesos iniciales diferentes, modificar los parámetros de aprendizaje, modificar el conjunto de entrenamiento o presentar los patrones en otro orden.

En [Kang94] y [Aoyagi96] se emplearon redes neuronales basadas en el algoritmo BP y en particular, en [Aoyagi96] se clasificaron los objetos candidatos en tres categorías: señales de limitación de velocidad, señales distintas a las de limitación de velocidad y elementos diversos considerados como ruido.

En [Escalera97] la clasificación se realizó empleando una red MLP con una capa de salida formada por 10 neuronas, correspondiente a los 10 tipos de señales que identifica. En [Vitabile02, Vitabile01] se sometieron diferentes conjuntos de imágenes a un proceso de segmentación por color y forma. El conjunto resultante de regiones de interés fue clasificado por una red neuronal MLP, cuyas entradas eran los canales RGB de las ROIs. También se empleó una red MLP en [Nguwi06] para determinar si cada ROI se corresponde con una señal de tráfico o no. En caso de tratarse de una señal se reescaló y se convirtió al espacio de color YCbCr. En la clasificación se utilizó una red MLP para cada clase con el objetivo de conseguir una arquitectura

extensible, de forma que la inclusión de una nueva clase no suponga el reentrenamiento de toda la arquitectura. Otra red MLP se empleó en [Bargeton08] para el reconocimiento de señales de velocidad, entrenándose con muestras de dígitos de secuencias reales.

Las redes RBF fueron utilizadas en [Estable94] y en [Paletta02], apuntándose que las redes RBF mejoran las prestaciones de las redes MLP en aquellos problemas en que se dispone de pocos datos o bien cuando la dimensión del vector de datos es muy grande. El carácter localizado de las funciones de base radial junto con la complejidad de las señales obliga a utilizar redes de gran tamaño con elevado coste computacional.

Algunas de las más recientes aproximaciones con NNs se encuentran en [Sermanet11, Ciresan11], donde los autores aplicaron Redes Neuronales Convolucionadas (*CNN-Convolutional Neural Network*) (ver 2.20), las cuales son redes jerárquicas que alternan capas de convolución con capas de sub-muestreo. En la arquitectura de las CNNs, la salida de cada capa consiste en un conjunto de imágenes denominado mapa de características en el que las neuronas de un mapa están conectadas con neuronas de la capa siguiente a través de máscaras de convolución. La configuración de una CNNs viene dada por la forma en que se realiza el proceso de entrenamiento y en cómo se implementan las capas de muestreo y de convolución. En [Ciresan11] se aplican diferentes técnicas de preprocesado para aumentar los niveles de contraste en las imágenes y el número de capas ocultas de la CNN se varía entre 4 y 7.

- **Otras técnicas:**

Existen en la literatura otras técnicas de clasificación empleadas para la clasificación de señales viales, que no se corresponden con las anteriormente mencionadas. Entre las mismas, cabe destacar el diseño en [Baro09] de un código corrector de error de salida (*ECOC-Error Correcting Output Code*), el cual está basado en una estrategia de codificación y decodificación con la que asignar una palabra-código a cada una de las clases. Además, el proceso de entrenamiento implica la combinación de esta propuesta con la implementación de un árbol de acuerdo a la metodología Random Forest (*Forest-ECOC*). La técnica de clasificación Random Forest ha cobrado fuerza en los últimos años y consiste en un árbol de decisión en el que por cada nodo se elige aleatoriamente la mejor partición a partir de un subconjunto de características. En el reconocimiento de señales de tráfico ha sido aplicada en [Kouzani07], [Zaklouta11] y [Zaklouta12]. Otro tipo de árbol utilizado es el K-d (abreviatura de árbol K-dimensional) utilizado en [Zaklouta11] y [Zaklouta12], donde el árbol puede verse como una estructura de datos que particiona el espacio mediante planos perpendiculares a uno de los ejes del sistema de coordenadas. Cada nodo del árbol K-d divide los datos en dos subespacios a partir de la característica con mayor varianza en ese nivel. Otras propuestas para el reconocimiento de señales viales se centran en los modelos ocultos de Markov (ver [Hsien06] y [Nguwi08]), donde se utilizan

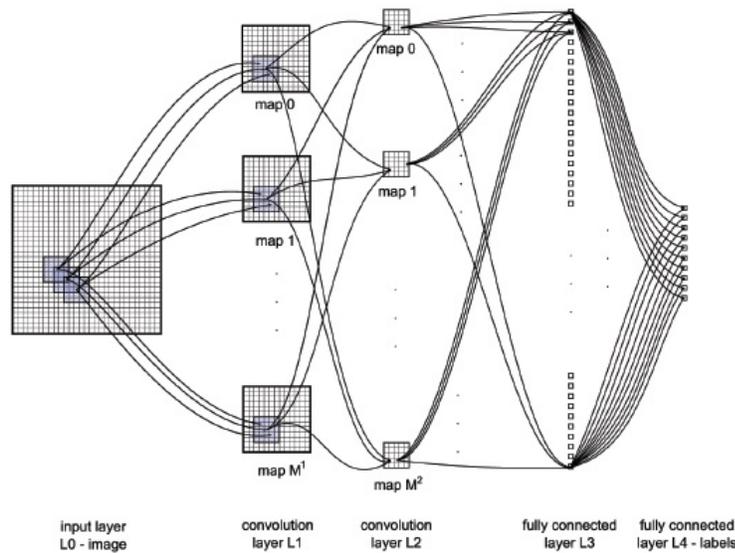


Figura 2.20: Arquitectura de una red neuronal convolucional en [Ciresan11], donde las capas de convolución están completamente conectadas. Ambas capas de convolución usan kernels  $5 \times 5$ .

mapas auto-organizativos para agrupar las señales. Si bien la técnica de AdaBoost fue propuesta para la detección, se ha empleado también para el reconocimiento [Ruta11] mediante una variante denominada SimBoost.

Otra alternativa es el uso de un clasificador de Bayes en [Bahlmann05] para fusionar las observaciones individuales de diferentes instantes de tiempo, asumiendo una distribución gaussiana del vector descriptor e independencia entre observaciones consecutivas. El clasificador otorga más peso a las últimas detecciones que a las primeras, ya que a medida que la cámara se acerca a una señal, ésta última se visualiza con mejor calidad. En [Gao13] el reconocimiento de señales se realiza mediante técnicas BoW, donde a partir del agrupamiento de las características SIFT, se genera un código y las señales se describen mediante histogramas utilizando el modelo BoW.

## 2.4. Sistemas Comerciales

Las principales empresas vinculadas al sector automovilístico están trabajando o participando, directamente o a través de sus empresas fabricantes de componentes, en la construcción de sistemas de ayuda a la conducción. La detección de elementos de señali-

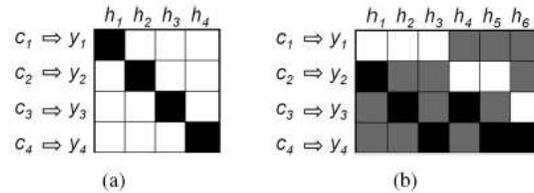


Figura 2.21: Diseño ECOC para cuatro clases en [Baro09]. (a): Codificación ECOC ‘uno contra todos’. (b): Codificación ECOC ‘uno contra uno’. (blanco:+1, negro:-1 y gris:0).

zación en estos sistemas es una tarea integrada a la que, en muchos casos, se ha buscado solución mediante visión por computador.

Entre las empresas que han presentado sistemas de detección de señalización, se encuentran los siguientes:

- Intel (*Connected car*). El sistema fue presentado en 2010 y en la actualidad INTEL se encuentra en conversación con fabricantes automovilísticos para estudiar la comercialización del mismo. El ordenador de a bordo reconoce las señales de tráfico gracias a una cámara exterior.
- Centro Tecnológico de Automoción de Galicia (CTAG). Trabajan en un prototipo de vehículo inteligente que incorpora, entre otras novedades, un sistema de identificación de señales de tráfico que no sólo las lee y alerta al conductor de su incumplimiento, sino que además frena el coche si persiste la infracción. Fue presentado en 2008.
- Opel. En 2009 Opel presentó el sistema *Opel Eye* con capacidad para detectar señales de limitación de velocidad y realizar seguimiento de las líneas delimitadoras de los márgenes de la carretera para advertir al conductor en caso de salirse de la calzada. La tecnología permite procesar hasta 30 imágenes por segundo e identificar el contenido de las señales que se encuentran a una distancia máxima de 100 metros del conductor.
- BMW. Su sistema de reconocimiento para señales viales ha sido el mejor de acuerdo a un estudio realizado por el ADAC (*Germany's largest auto club*). Este análisis comparativo fue realizado en 2010, evaluando los sistemas de un BMW 740d, un Mercedes S 500 CGI, un Audi A8, un Opel Insignia y un VW Phaeton. Con la excepción de Opel, todos los sistemas combinan cámaras y mapas de localización para detectar señales de limitación de velocidad. Los datos son ‘vistos’ por las cámaras y verificados con mapas de navegación, a excepción del sistema de Opel que se apoya únicamente en las cámaras del vehículo. Todos los sistemas mostraron problemas con la variabilidad de las señales, especialmente con las basadas en tecnología LED.

- Nissan. La promesa de la marca automovilística japonesa es que el modelo comercial de su Nissan Leaf con sistema de conducción asistida pueda estar en 2020 en la calle. Entre las ayudas a la conducción que incluye hay que mencionar el asistente para situarse en el centro del carril correspondiente, control de velocidad, parada automática en caso de peligro o cambio de carril automático, incluyendo por supuesto un detector de señalización.

## 2.5. Resumen

En este capítulo se han descrito las principales técnicas utilizadas para detectar y reconocer la señalización vertical en carretera con una arquitectura basada en tres etapas: detección, clasificación y reconocimiento. Respecto a la etapa de detección, existen dos grandes aproximaciones: las basadas en color y las basadas en bordes. Respecto a la primera línea, ningún espacio cromático es totalmente inmune a factores como los cambios de luz, el envejecimiento de material o la distancia de observación. Entre los problemas apenas tratados en los algoritmos por segmentación de color se encuentran los siguientes:

- Escasa iluminación en condiciones de visibilidad nocturna, lluvia y niebla.
- Señales expuestas a diferentes niveles de iluminación en su superficie, como por ejemplo en la proyección de sombras que hay partes más iluminadas que otras.
- Señales con color similar al entorno que las rodea.
- Condiciones de brillo en las que las señales actúan como un espejo que refleja luz de la fuente hacia la cámara.

La detección basada exclusivamente en color limita la internacionalización del sistema, ya que al no existir una estandarización de colores entre las señales de los diferentes países, los sistemas que se apoyan en el color necesitan ser reajustados al cambiar de un país a otro.

Otra alternativa para la detección es el uso de bordes, donde los aspectos más problemáticos son:

- Detección de objetos con formas similares a las señales de tráfico, tales como edificios y automóviles.
- Detección de señales dañadas u ocluidas por otros objetos, donde las formas quedan distorsionadas.
- Como la distancia entre la cámara y las señales es variable, el tamaño de la señal también varía. Cuando la señal es muy pequeña, puede no ser reconocible por su forma.

- Detección de formas ante deformaciones proyectivas.

Para la clasificación de formas geométricas son varios los algoritmos utilizados basados en la geometría de las formas habituales de las señales. Entre los métodos propuestos, caben destacarse aquellos basados en la búsqueda de máscaras de ángulos específicos de acuerdo a las formas a detectar mediante un proceso de convolución, los inspirados en aprovechar la simetría radial de las formas de las señales y los que emplean la transformada de Hough. En cualquier caso, la descripción de los métodos propuestos apenas analizan su robustez ante oclusiones y rotaciones.

Respecto a la etapa de identificación del pictograma, cabe diferenciar los algoritmos en base al descriptor utilizado y a la técnica de clasificación. El descriptor puede ser un vector formado por las componentes en el dominio espacial o bien una proyección en otro espacio diferente. En cuanto a la técnica de clasificación, la diversidad es amplia: desde técnicas de correlación hasta redes neuronales. No obstante, apenas existen trabajos que pongan a prueba un sistema con un conjunto de señales completo, sino que la mayoría de los trabajos prueban los algoritmos con un conjunto reducido.

Aunque casi todos los sistemas propuestos siguen una estructura modular, no siempre es el caso. Así, cabe destacar aquellos que no realizan una segmentación previa, sino que desarrollan una búsqueda de los elementos de interés a través de la imagen completa. Finalmente, es importante incidir en el hecho de que la mayor parte de los trabajos se centran en una parte del sistema y son escasos los que describen un sistema completo de detección y reconocimiento.



# Capítulo 3

## Bases de Datos de Señalización Vertical

La implementación de un sistema para detección y reconocimiento de señalización vertical exige la necesidad de disponer de una base de datos con escenas de tráfico reales que permitan testear el nivel de calidad del mismo. Si además el sistema incluye un módulo de reconocimiento basado en algoritmos supervisados es necesario disponer de muestras de entrenamiento para el aprendizaje de los correspondientes clasificadores. La identificación del pictograma de las señales mediante técnicas de inteligencia artificial es un problema que se puede categorizar en el contexto del reconocimiento de patrones, donde el objetivo es la clasificación dentro de un conjunto de categorías o clases. Así, la necesidad de disponer de una base de datos anotada en esta aplicación es doble. En primer lugar, si se utilizan técnicas de clasificación supervisadas es necesario disponer de patrones anotados para que el sistema sea capaz de aprender. Por otra parte e independientemente de la técnica de clasificación (supervisada o no) que se utilice, es necesario cuantificar la calidad del sistema y para ello es necesario disponer de un conjunto de patrones etiquetados.

### 3.1. Bases de datos existentes

A diferencia de otras aplicaciones basadas en el reconocimiento de patrones, tales como la identificación de rostros, en el campo del reconocimiento de señales de tráfico no existen bases de datos en uso suficientemente generalizadas. Este hecho supone un problema en el sentido de que los algoritmos propuestos en este campo han sido comparados escasamente entre sí con el objetivo de realizar un análisis comparativo. No obstante, desde 2011 se han comenzado a dar algunos pasos en este sentido y en la actualidad se encuentran publicadas las siguientes bases de datos sobre señales viales:

1. **German Traffic Sign Benchmark (GTSRB)** [Houben13]. Se trata de una base de datos anotada del Instituto Neuroinformático de Alemania para el reconocimiento que contiene más de 50 000 imágenes repartidas entre un total de 43 clases de señales. Las imágenes se subdividen en dos sub-conjuntos: uno de entrenamiento, con 39 209 imágenes y otro de test, con 12 630 imágenes. Las muestras varían el

rango de su tamaño desde  $15 \times 15$  a  $250 \times 250$  píxeles y presentan la señal incluyendo además, un pequeño margen de, al menos, 5 píxeles alrededor de la misma. La base de datos aporta tres descriptores para cada una de las muestras: histograma de gradientes orientados (HOG-*Histogram of oriented gradients*), características de Haar e histograma de matiz. En la Figura 3.1 se visualizan ejemplos de patrones de esta base de datos. En los últimos meses, se ha publicado una base de datos en paralelo para la detección de señales que incluye 900 imágenes de escenas de tráfico (600 para el entrenamiento y 300 para testeo), donde cada una contiene de 0 a 6 señales.

2. **Swedish University Traffic Sign Dataset (STS Data set)** [Larsson11]. Esta base de datos, perteneciente a la Universidad de Linköping, incluye aproximadamente 20 000 imágenes extraídas de secuencias de video en entornos urbanos, las cuales incluyen 3488 señales viales. En este caso, las imágenes no contienen únicamente señales, sino que engloban la escena del entorno en que aparecen las señales. Esta base de datos no tiene todas las imágenes anotadas. En la anotación, además de indicarse las coordenadas de la señal en el plano imagen, se incluye la situación de visibilidad tal como una posible oclusión o visibilidad difusa (ver Figura 3.2).
3. **KUL Belgium Traffic Sign Dataset (KUL Data set)** [Timofte11]. Este conjunto, perteneciente a la Universidad Católica de Lovaina, abarca 62 clases de señales utilizadas en Bélgica y es dividido en un conjunto de entrenamiento con 4591 imágenes y otro de test con 2534 muestras. También en este caso, cada imagen de muestra incluye una señal con un margen de la escena alrededor de la misma. En la Figura 3.3 se visualizan muestras de esta base de datos. La base de datos incluye cuatro secuencias que pueden ser utilizadas para algoritmos de seguimiento y aporta información de las cámaras para cada imagen.
4. **(RUG Data set)** [Grigorescu03]. Esta base de datos contiene tan sólo 48 imágenes de tamaño  $360 \times 270$  píxeles, donde cada una de ellas representa una escena de tráfico que incluye una o más señales. Las imágenes están agrupadas en 3 clases correspondientes a una categoría distinta (paso de peatones, intersección y carril-bici). En la figura 3.4 se muestra una imagen de esta base de datos.

A modo de resumen comparativo, en la tabla 3.1 se muestran las características de las bases de datos descritas. Las bases de datos STS y RUG incluyen escenas de tráfico y pueden ser utilizadas para la detección y el reconocimiento. Sin embargo, al no aportar conjuntos de patrones, obligan a extraer muestras a partir de las imágenes para la tarea de reconocimiento. En cualquier caso, el número total de imágenes con señales en ambas bases de datos es reducido, en concreto 3488 y 48 en STS y RUG, respectivamente. Esta circunstancia imposibilita efectuar una partición de entrenamiento y test significativa. Por otra parte, las bases de datos GTSRB y KUL están orientadas al reconocimiento



Figura 3.1: Muestras de la base de datos *German Traffic Sign Benchmark* (GTSRB).



Figura 3.2: Muestra de la base de datos de la Universidad de *Linköping* (STS Data set).

y subdividen las muestras en dos conjuntos: uno de entrenamiento y otro de test. Para poder evaluar la capacidad de detección del sistema en la escena original, la base de datos GTSRB hizo pública en 2013 un conjunto de imágenes anotadas para la detección de señales de tan sólo 1200 muestras. Por tanto, se puede concluir que la única base de datos utilizable para evaluar la capacidad de un TSDRS completo es la GTSRB, a pesar de estar limitada por los siguientes aspectos: 1) el número de clases es limitado y no aborda un conjunto completo de señales, 2) el número de imágenes para la detección no es representativo, y 3) no se incluyen muestras de ruido, las cuales son necesarias para evaluar la capacidad de rechazo a las falsas alarmas.

## 3.2. Características de una nueva base de datos

### 3.2.1. Especificaciones

Para el desarrollo de esta tesis doctoral se ha creado una base de datos específica bajo los siguientes requisitos:



Figura 3.3: Muestras de la base de datos *KUL Belgium Traffic Sign Dataset*.



Figura 3.4: Muestra de la base de datos *RUG Data set*.

	GTSRB	STS	KUL	RUG
Nº. Clases	43	7	62	3
Nº. Muestras	51 839	-	7125	-
Tamaño muestras	15×15 a 250×250 píxeles	-	21×21 a 328×446 píxeles	-
Nº. Imágenes	900 con/sin señales	20 000	-	48 con señales
Tamaño imágenes	1360×800	1280×960 píxeles	-	360×270 píxeles
País de origen	Alemania	Suecia	Bélgica	Holanda
Información adicional	Diferentes imágenes anotadas de la misma secuencia.	Señales anotadas con condiciones de visible/ocluída/blurred.	Incluye anotaciones de señales, calibración de cámaras y poses.	No incluye anotaciones.

Tabla 3.1: Información de las bases de datos públicas de señales viales.

1. Inclusión de imágenes de tráfico. Para poder evaluar la capacidad de detección y reconocimiento de los sistemas TSDRS, la base de datos aporta un número significativo de imágenes de escenarios viales y la correspondiente anotación de las coordenadas de las señales presentes en la imagen, la forma geométrica de la señal para algoritmos de clasificación de forma y la categoría de la señal para algoritmos de reconocimiento.
2. Inclusión de patrones. Con el objetivo de centrarse en algoritmos orientados exclusivamente al reconocimiento de las señales, la base de datos incluye un conjunto significativo de patrones a partir del cual se generarán los correspondientes subconjuntos para entrenamiento y test. En este conjunto de muestras es necesario incluir patrones de ruido.
3. Tamaño realista. Para poder aportar resultados representativos de un TSDRS, se hace necesario trabajar con la totalidad de señales viales de un país y no con conjuntos limitados de clases. Además, en la creación de la base de datos se ha intentado incluir por cada clase un número de patrones significativo que recoja, en la medida de lo posible, diferentes condiciones de iluminación, rotaciones y variantes del pictograma. Para conseguir muestras con diferentes condiciones ambientales, los procesos de grabación se han realizado con diferentes cámaras y además, se han ajustado los parámetros de cada una de ellas con diferentes configuraciones con la finalidad de aportar el máximo grado de variabilidad posible.

### 3.2.2. Descripción

La base de datos se vertebra en una estructura de directorios organizada jerárquicamente por color y forma geométrica. La estructura se organiza de acuerdo a la figura 3.5, teniendo en cuenta que las señales pueden ser segmentadas por cuatro colores típicos: azul('BSg'), rojo('RSg'), blanco ('WSg') y amarillo ('YSg'), y que pueden presentar tres formas geométricas: 'Círculo', 'Triángulo' y 'Rectángulo'. Debido a que las señales de advertencia de peligro y numerosas señales de reglamentación contienen una orla externa roja alrededor de un fondo blanco o amarillo, éstas pueden ser segmentadas por dos colores. De esta manera, podemos encontrar muestras de una misma señal en diferentes directorios (como un ejemplo, encontraremos patrones de señales de límite de velocidad dentro del directorio 'Círculo' de color rojo('RSg') y en el directorio 'Círculo' de color blanco ('WSg')). En cualquier caso, por cada muestra se generan tres archivos con la siguiente información:

- **Imagen de la escena completa.** Imagen de la escena vial completa que incluye la señal.

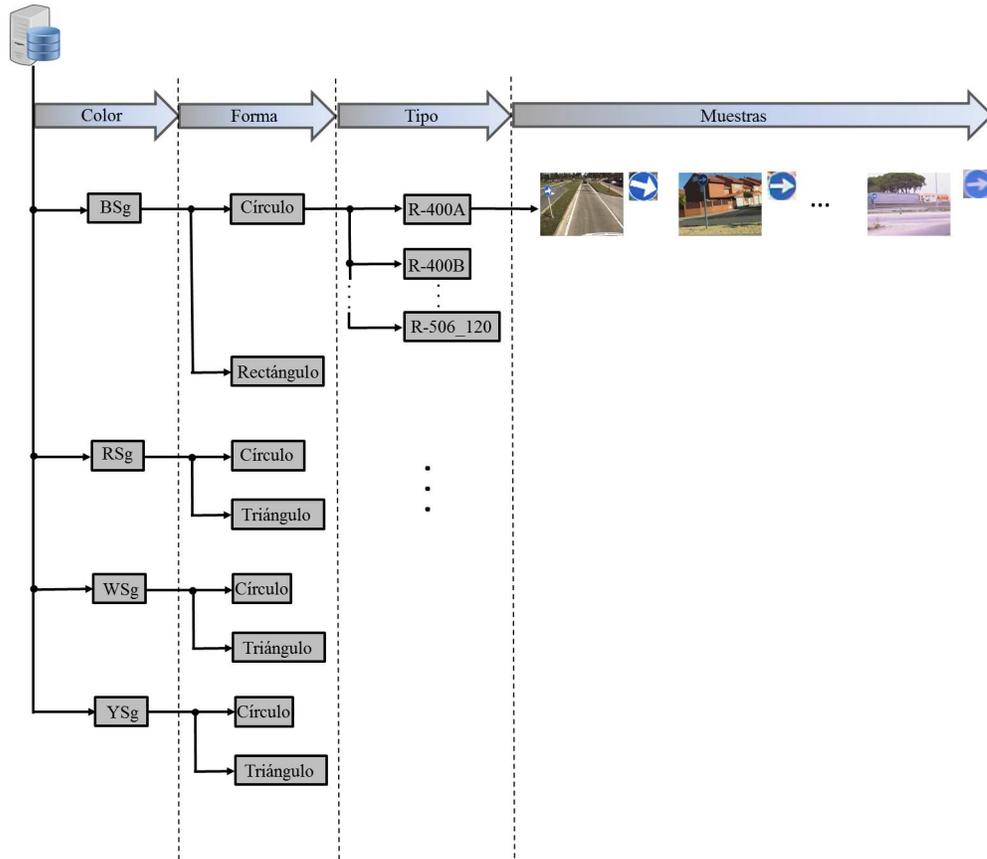


Figura 3.5: Estructura de directorios por color y forma en la base de datos.

- **Patrón de la señal.** Imagen de la muestra recortada de la señal de tráfico, la cual es extraída de la imagen completa. Cada blob es normalizado en tamaño, en concreto a  $31 \times 31$  píxeles.
- **Anotación.** Archivo que contiene información complementaria asociada a la muestra. Incluye la ruta y el nombre de la imagen original a partir de la cual ha sido extraída, la forma geométrica de la señal, el color por el que ha sido segmentada, la matriz de transformación que permite realizar la extracción del blob a partir de la imagen de la escena completa y el tipo de señal.

En la tabla 3.2 se describe la distribución de patrones en la base de datos creada, detallando para cada combinación de color y forma el número de clases consideradas, el número de muestras totales, el número medio de muestras por clase, la desviación estándar en el número de muestras por clase y el número total de muestras ruidosas. A tenor de la tabla 3.2, se puede afirmar que las señales que aparecen con mayor frecuencia son las de prohibición (circular con orla roja y fondo blanco) y las de advertencia de peligro (triangular con orla roja y fondo blanco). Por el contrario, las señales de tramos de obra con fondo amarillo son excepcionales. El hecho de que la base de datos no sea balanceada

se refleja a dos niveles: 1) el número de muestras totales entre combinaciones de color y forma es muy diferente y 2) el número de muestras entre clases dentro de una combinación de color y forma es también muy diferente. Es previsible que las clases con un número de muestras poco significativo crearán modelos de aprendizaje de escasa generalización y los resultados tendrán validez limitada. A modo de ejemplo de distribución de muestras, en la Tabla 3.3 se detalla el número de patrones de cada clase para las señales rojas circulares.

Combinación (Color-Forma)	# Clases no ruidosas	#Muestras no ruidosas			# Muestras ruidosas
		Total	Promedio	Desv. Estándar	
Rojo Circular 	62	14 536	230	406	3895
Rojo Triangular 	45	9246	204	345	3066
Azul Circular 	54	2160	40	180	610
Azul Rectangular 	99	2709	27	64	890
Blanco Circular 	114	6695	59	235	1574
Blanco Triangular 	44	8133	181	492	778
Amarillo Circular 	47	374	8	19	1162
Amarillo Triangular 	26	415	16	62	152

Tabla 3.2: Distribución de clases y muestras en la base de datos de señalización vertical.

Clase	#Muestras	Clase	#Muestras	Clase	#Muestras
 R-100	68	 R-101	1141	 R-102	164
 R-103	146	 R-104	138	 R-105	120
 R-106	136	 R-107	148	 R-108	178
 R-109	45	 R-110	44	 R-111	94
 R-112	107	 R-113	140	 R-114	27
 R-115	87	 R-116	169	 R-117	122
 R-2	1218	 R-200	185	 R-201	3
 R-201	1	 R-201	2	 R-201	107
 R-201	1	 R-202	95	 R-203	87
 R-204	74	 R-205	65	 R-205	50
 R-205	2	 R-300	51	 R-301	65
 R-301	87	 R-301	93	 R-301	188
 R-301	137	 R-301	2138	 R-301	448
 R-301	909	 R-301	625	 R-301	170
 R-301	287	 R-301	189	 R-301	1

Continúa en la página siguiente.

Clase	#Muestras	Clase	#Muestras	Clase	#Muestras
 R-301	191	 R-302	298	 R-303	275
 R-304	114	 R-305	1934	 R-306	130
 R-307	123	 R-308	186	 R-308a	41
 R-308b	55	 R-308c	21	 R-308d	47
 R-308e	431	 R-310	87	 R-5	227
 R-202	1	 RUIDO	3895		

Tabla 3.3: Distribución de patrones en la base de datos para las señales rojas circulares.



# Capítulo 4

## Sistema para Detección y Reconocimiento de Señalización Vertical

Una vez enmarcada esta tesis dentro de los Sistemas Inteligentes de Transporte, se puede afirmar que la detección de señales en carretera es un reto desafiante y un campo que, a pesar de haber sido muy explotado en la última década, se encuentra aún abierto a la investigación al no contarse a día de hoy con un TSDRS definitivo. En esta tesis se ha apostado por desarrollar un sistema para detección y reconocimiento de señalización basado en visión artificial, denominado RESET (*Reconocimiento de Señales de Tráfico*).

El sistema en el que se basa la investigación de esta tesis doctoral se ha estructurado en varias etapas de acuerdo a una arquitectura modular. De esta manera, se garantiza el funcionamiento independiente de cada tarea y se facilita la configuración de la fase experimental para la ejecución de diferentes algoritmos. Asimismo, la estructura modular permite trabajar con cada una de las etapas por separado a pesar de la estrecha relación que existe entre ellas, ya que cada módulo alimenta la entrada del siguiente.

En la figura 4.1 se presenta el diagrama de bloques del trabajo propuesto, el cual está estructurado en los siguientes módulos:

- Adquisición para captura de información.
- Detección.
- Clasificación de acuerdo a las características geométricas.
- Reconocimiento del pictograma.
- Seguimiento.

En las próximas secciones se irán describiendo los algoritmos en los que se basa esta tesis doctoral, donde las SVM han cobrado gran relevancia en este trabajo de investigación como herramienta de clasificación. En el apéndice B se describen los fundamentos

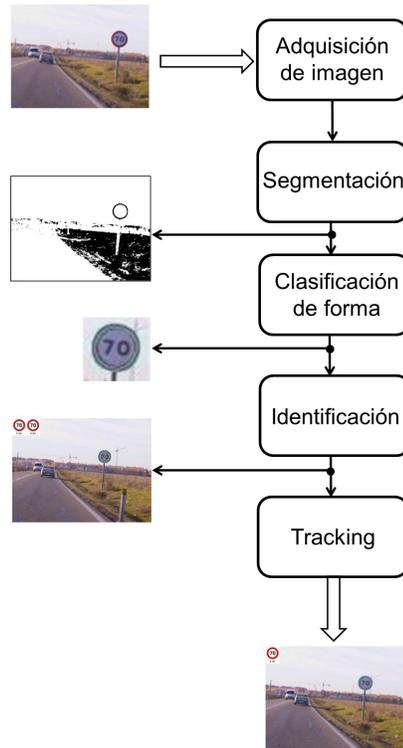


Figura 4.1: Arquitectura modular del sistema.

matemáticos en los que están basadas y sus propiedades para un sistema de clasificación binario.

## 4.1. Módulo de captura

El módulo de adquisición se encuentra instalado a bordo de un vehículo experimental para la captura y almacenamiento de imágenes de entornos viales de forma eficiente y sincronizada (ver figura 4.2). Debido a la gran cantidad de datos a manejar en la aplicación, el diseño del hardware de adquisición debe mantener una eficiente vía de comunicación entre las cámaras (ver figura 4.2(b)) y el soporte de almacenamiento, tanto de las imágenes capturadas como de la información del GPS. El sub-sistema de adquisición consta de varias cámaras *firewire*, ordenadores portátiles, una placa con un microcontrolador, un receptor GPS y un odómetro (ver figura 4.2(c)), donde este último permite obtener medidas de la distancia cubierta por el vehículo en un trayecto. En realidad, el odómetro genera pulsos eléctricos a medida que el vehículo se desplaza y estos pulsos son contabilizados por la placa.

La información del odómetro obtenida por el microcontrolador sincroniza las cámaras, las cuales son controladas mediante los ordenadores portátiles. Debido a que el vehículo en el que se encuentra instalado el módulo de captura está en movimiento mientras se produce la grabación, las cámaras deben estar perfectamente sincronizadas para garantizar

que ambas capturan la misma escena. Todos los parámetros de la grabación son ajustados mediante software y cada imagen es almacenada junto con la correspondiente posición GPS y la posición relativa del odómetro.

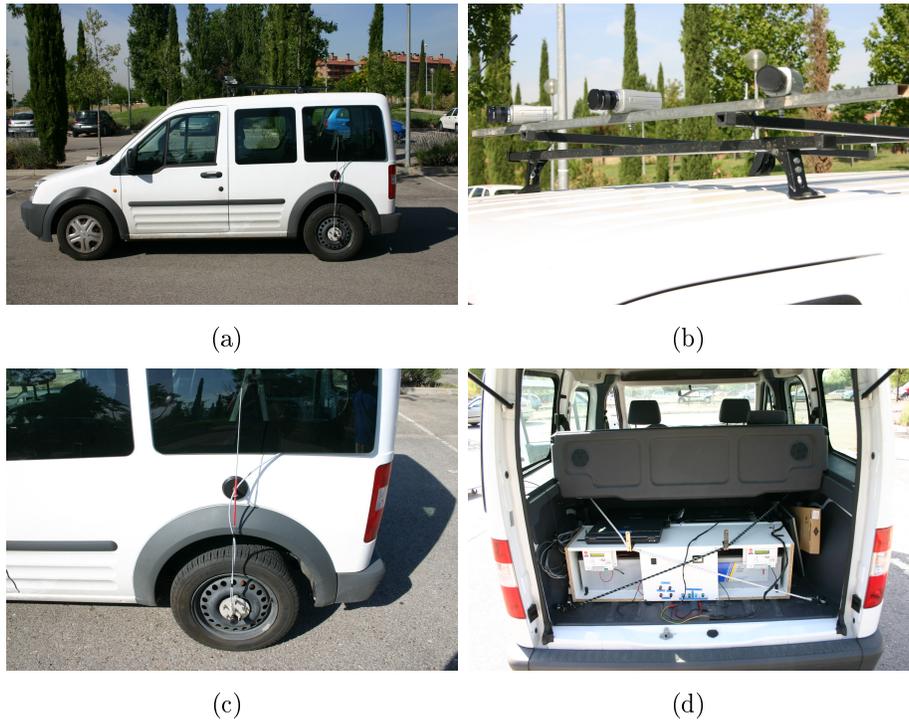


Figura 4.2: Vehículo con sistema de captura a bordo. (a) Vehículo de prueba. (b) Sistema de cámaras. (c) Odómetro para medida de distancias. (d) Sistema interno de adquisición.

La captura de la escena mediante varias cámaras sincronizadas permite construir una imagen panorámica a partir del establecimiento de píxeles de correspondencia. Si se trabaja, por ejemplo, con tres cámaras, la correspondencia se establece entre cada par de imágenes: par izquierda-central y par central-derecha. Las zonas de solapamiento entre imágenes permiten conocer la correspondencia mediante la transformación de las coordenadas de una y otra imagen. En la figura 4.3 se muestra la escena panorámica construida a partir de las imágenes captadas por las cámaras.

## 4.2. Módulo de Segmentación

Diferentes espacios de color han sido utilizados en la literatura para segmentar señales viales basándose en técnicas de visión artificial. Uno de los mayores problemas en utilizar segmentación por color en entornos exteriores es la variación cromática de la luz, que causa cambios en el color de los objetos. Probablemente, una de las más importantes funciones del sistema visual humano es el reconocimiento de objetos en el sentido de que

la apariencia su color persiste a pesar de cambiar la iluminación. Este fenómeno físico es conocido como ‘constancia del color’ y se define como la habilidad de un sistema visual de reconocer el color de un objeto bajo variaciones luminosas.

En esta tesis doctoral se propone un modelo de detección basado en una umbralización del espacio de color HSI adaptada para condiciones diurnas y nocturnas. Con el objetivo de estimar los valores óptimos de los umbrales, se realiza un análisis de la distribución de las componentes del espacio de color.

### 4.2.1. Variación de color en entornos exteriores

La figura 4.4 representa un modelo para la captura de la imagen. La luz de la fuente incide en una superficie, es reflejada y entra a la cámara, donde es captada por los fotoreceptores. En esta figura,  $\mathbf{n}$ ,  $\mathbf{s}$  y  $\mathbf{v}$  son los vectores unitarios que representan, respectivamente, la dirección del vector normal a la superficie de la señal vial, la dirección de la fuente de iluminación y la dirección del observador.

El color percibido de cualquier objeto en una imagen en color depende de tres factores:

- Color, intensidad y posición de la fuente de luz.
- Reflectancia espectral de las superficies de la escena, las cuales determinan cómo las superficies reflejan el color.
- Respuesta espectral de los sensores del sistema de captura

Una de las mayores dificultades a la hora de trabajar con colores en imágenes de entornos exteriores es la variación de los colores según la hora del día. Los cambios a lo largo del día varían de acuerdo a la posición del sol, la nubosidad y las condiciones climatológicas. Numerosos estudios realizados en la década de 1960-69 [Judd64] determinaron un modelo aproximado para el color de la luz diurna mediante la curva característica mostrada en la figura 4.5, que responde a la siguiente ecuación:

$$y = 2.87x - 3.0x^2 - 0.275, \quad \text{para } 0.25 \leq x \leq 0.38 \quad (4.1)$$

donde  $x$  e  $y$  son las coordenadas del modelo de color CIE. La ecuación muestra que la variación de color de la luz diurna se controla con una única variable, denominada ‘temperatura de color’, que es independiente de la intensidad.

De acuerdo a [Ebner04], la medida de luz de cada sensor del sistema de captura en la posición  $\mathbf{p}_s$  a partir de la luz reflejada en la posición  $\mathbf{p}_0$  sobre una superficie lambertiana viene dada por:

$$\rho_i(\mathbf{p}_s) = \int_{\lambda} R(\lambda, \mathbf{p}_0) E(\lambda) S(\lambda) d\lambda \quad \text{para } i = \{R, G, B\}, \quad (4.2)$$



Figura 4.3: Construcción de una escena panorámica a partir de las imágenes captadas por las cámaras. (a) Imagen izquierda. (b) Imagen central. (c) Imagen derecha. (d) Imagen resultante del proceso de construcción.

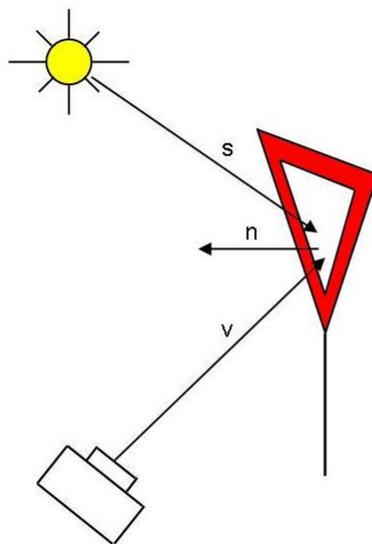


Figura 4.4: Modelo de una escena de tráfico.

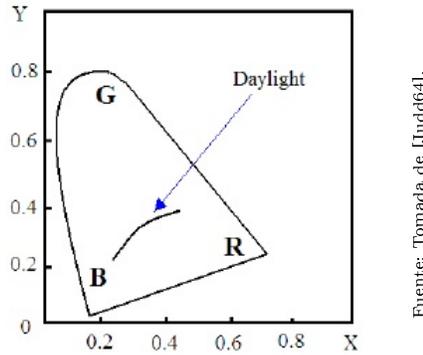


Figura 4.5: Curva CIE de luz diurna.

donde  $\rho_i$  es la medida de cada sensor,  $R(\lambda, \mathbf{p}_0)$  es la reflectividad, la cual especifica el porcentaje de luz con longitud de onda  $\lambda$  reflejado por la superficie en la posición  $\mathbf{p}_0$ ,  $E(\lambda)$  es la densidad espectral de la luz ambiental que incide en la superficie y  $S(\lambda)$  representa la sensibilidad espectral del sensor. La integral es evaluada sobre todas las longitudes de onda del espectro visible a las cuales responde el sensor. En general, el sistema de captura dispone de tres sensores con respuestas que cubren las regiones espectrales de rojo, verde y azul (espacio RGB) y por tanto, cada sensor crea un canal. Una de las claves de la ecuación 4.2 es que el color aparente de cualquier objeto en imágenes captadas en entornos exteriores varía de acuerdo a los cambios de luz, a la geometría de iluminación, a la geometría de visión y a los parámetros de configuración de las cámaras.

Asumiendo que los sensores son ideales para la luz roja, verde y azul ( $S_i(\lambda) = \delta(\lambda - \lambda_i)$ ) con  $i = \{R, G, B\}$ , la ecuación 4.2 se simplifica como:

$$\rho_i(\mathbf{p}_s) = R(\lambda_i, \mathbf{p}_0)E(\lambda_i), \quad (4.3)$$

y por tanto, la luz ambiental de la escena simplemente escala la reflectancia del objeto.

Desde la teoría del color, las diferencias percentuales en las densidades espectrales son captadas mediante el tono, la saturación y el brillo (luminancia). El tono es determinado por la longitud de onda dominante en la densidad espectral de la luz, teniendo en cuenta que los tonos espectrales son longitudes de onda únicas en el espectro visible. Por ejemplo, los colores primarios (rojo, verde y azul) están localizados en posiciones específicas del espectro. La saturación es la magnitud del tono respecto al resto de longitudes de onda:

$$S = \frac{s_1}{s_2}, \quad (4.4)$$

donde  $s_1$  es la cantidad de luz en la longitud de onda dominante y  $s_2$  es la cantidad de luz en todo el espectro. El brillo es una medida global de la cantidad de luz que pasa a través de todo el espectro y puede verse como un factor de escala aplicado al mismo. La localización y forma del pico en el espectro aportan las componentes de tono y saturación y determinan la percepción del color.

### 4.2.2. Invarianza del Tono

El significado del tono o matiz, simbolizado por la componente H en el espacio HSI, está vinculado directamente con el color de un objeto. El sistema visual humano percibe con el mismo tono superficies visualizadas en muy diferentes contextos. De acuerdo a la Comisión Internacional de Iluminación (CIE), el tono es el atributo de una sensación visual por la que un área parece similar a uno de los colores percibidos: rojo, amarillo, verde y azul, o una combinación de dos de ellos [CIE89]. Como ya ha sido demostrado en [Gevers99], el tono es altamente invariante a ciertas peculiaridades como el brillo y los reflejos, por lo que es un criterio clave para la segmentación. Las propiedades más interesantes del tono frente a las variaciones luminosas son las siguientes:

- El tono es invariante a factores multiplicativos/cambios de escala cumpliéndose que  $H(R, G, B) = H(aR, aG, aB)$  para todos los valores de  $a$ , tal que  $(aR, aG, aB) \in [0, 255] \times [0, 255] \times [0, 255]$ .
- El tono es invariante a desplazamientos/sumas cumpliéndose que  $H(R, G, B) = H(R + b, G + b, B + b)$  para todos los valores de  $b$ , tal que  $(R + b, G + b, B + b) \in [0, 255] \times [0, 255] \times [0, 255]$ .
- El tono es invariante a cambios de saturación. De hecho, el tono no varía bajo los efectos de las sombras y los golpes de luz, siempre que la luz sea blanca.

No obstante, el tono presenta el problema de la inestabilidad. De hecho, pequeños cambios en las coordenadas RGB pueden causar grandes variaciones de tono [Lalonde95] para colores con baja saturación, o lo que es lo mismo, para colores cercanos a los grises. Lo que sucede al trabajar con grises es que no distingue entre un rojo con baja saturación de un verde o azul con baja saturación.

### 4.2.3. Segmentación basada en HSI

En esta tesis se propone una segmentación por color basada en una umbralización de las componentes de tono y saturación del espacio HSI, donde la componente de tono codifica intrínsecamente el color. No obstante, debido a que las señales se fabrican con colores muy distinguibles utilizando altos niveles de saturación, el modelo de segmentación selecciona píxeles con valores altos de saturación. El modelo de segmentación para cada canal de color se rige mediante la ecuación:

$$O_i(x, y) = \begin{cases} 1, & \text{si } ThA_i \leq H_i(x, y) \leq ThB_i \quad \& \quad ThC_i \leq S_i(x, y) \leq ThD_i \\ 0, & \text{en otro caso} \end{cases} \quad (4.5)$$

donde  $H(x, y)$  y  $S(x, y)$  representan los niveles de tono y saturación de un píxel genérico de coordenadas  $(x, y)$ , y  $O_i(x, y)$  denota el valor de salida del proceso de segmentación

Canal de color	ThA <sub><i>i</i></sub>	ThB <sub><i>i</i></sub>	ThC <sub><i>i</i></sub>	ThD <sub><i>i</i></sub>
Rojo ( <i>i</i> =1)	345	15	30	255
Azul ( <i>i</i> =2)	190	270	45	255
Amarillo ( <i>i</i> =3)	20	60	150	255

Tabla 4.1: Umbrales de los canales cromáticos.

para cada canal de color *i*-ésimo. Para fijar los umbrales estáticos de cada canal ( $ThA_i$ ,  $ThB_i$ ,  $ThC_i$  y  $ThD_i$ ) se han construido los histogramas de las componentes H y S considerando los colores habituales con los que son fabricadas las señales en España: rojo, azul y amarillo. En este trabajo se han tomado manualmente muestras de píxeles de señales correspondientes a estos colores y a partir de las mismas, se han elaborado los histogramas para los tres canales mencionados (ver figura 4.6). Aquí es necesario considerar que la respuesta a la variación de la longitud de onda y a la intensidad de los dispositivos de captura estándar es no-lineal e interdependiente ([Thomson01]). Debido a esta circunstancia, las imágenes utilizadas para la extracción del histograma han sido capturadas bajo diferentes condiciones luminosas, usando diferentes cámaras y además, cambiando la configuración de las mismas. El objetivo es incluir un amplio rango de situaciones con condiciones muy variadas para fijar umbrales adecuados. La Tabla 4.1 muestra los valores fijados para  $ThA_i$ ,  $ThB_i$ ,  $ThC_i$  y  $ThD_i$ , donde por definición del espacio HSI, la componente *H* está acotada en el intervalo  $[0^\circ-360^\circ]$  y la componente *S* en el intervalo  $[0-255]$ .

Además de los canales cromáticos de rojo, azul y amarillo, el blanco es otro de los colores empleados tanto en las señales de fin de prohibición como en el fondo de las señales de advertencia de peligro y prohibición. Para la extracción del canal acromático, el modelo propuesto en esta tesis se apoya en el método descrito en [Liu02], de acuerdo a:

$$f(R, G, B) = \frac{(|R - G| + |G - B| + |B - R|)}{3D}, \quad (4.6)$$

donde *R*, *G* and *B* representan las componentes de color y *D* el grado de extracción de acromaticidad. Si el resultado  $f(R, G, B)$  es inferior a 1, se obtienen colores acromáticos y si es mayor que 1, representa colores cromáticos. Tras realizar diversas pruebas, los mejores resultados de segmentación fueron conseguidos con  $D = 20$ . En cualquier caso, los colores acromáticos no representan sólo el blanco, sino que comprenden toda la gama de grises que van del blanco al negro. Por esta razón para decidir si un píxel es blanco, debemos valorar si las componentes RGB presentan niveles suficientemente altos. En consecuencia, la segmentación acromática del modelo propuesto se basa en una doble condición:

$$O_w(x, y) = 1 \quad si \begin{cases} f(R, G, B) < 1 \\ ((R + G + B)/3) > 150. \end{cases} \quad (4.7)$$

De acuerdo al modelo propuesto, los píxeles de la imagen pueden ser segmentados por cualquiera de los cuatro canales propuestos. En la figura 4.7 se muestran cuatro imágenes

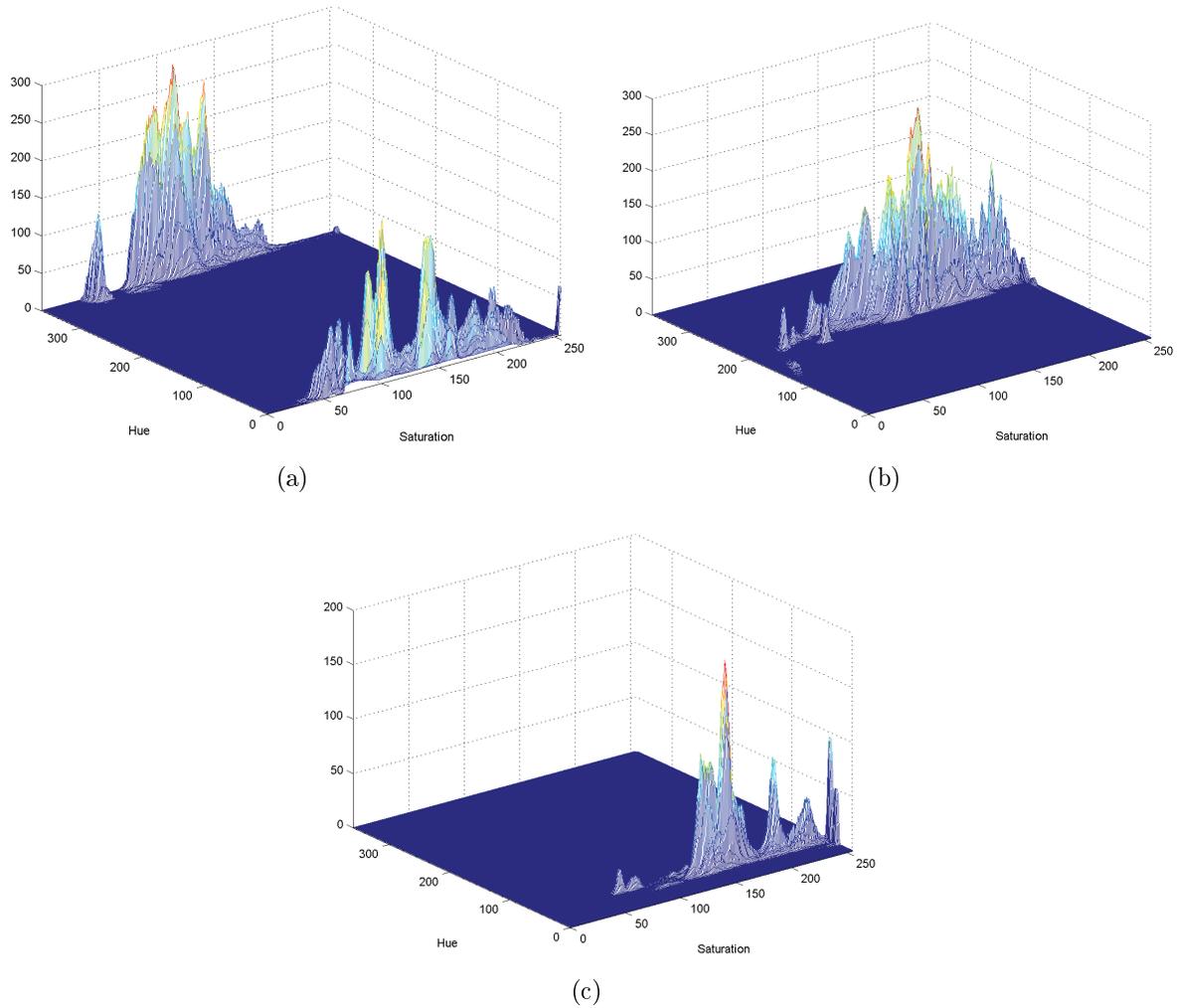


Figura 4.6: Histogramas H-S en visibilidad diurna. (a) Canal rojo. (b) Canal azul. (c) Canal amarillo.

Canal de color	ThA <sub><i>i</i></sub>	ThB <sub><i>i</i></sub>	ThC <sub><i>i</i></sub>	ThD <sub><i>i</i></sub>
Rojo ( <i>i</i> =1)	345	15	200	255
Azul ( <i>i</i> =2)	190	270	150	255
Amarillo ( <i>i</i> =3)	20	60	X	255
Blanco ( <i>i</i> =4)	30	70	30	170

Tabla 4.2: Umbrales de segmentación para visibilidad nocturna.

de ejemplo bajo diferentes condiciones de luminosidad y sus correspondientes máscaras de segmentación dispuestas en columnas.

Teniendo en cuenta que la práctica totalidad de las señales de advertencia de peligro y de prohibición presentan un fondo blanco y una orla exterior roja, a excepción de las señales de ‘entrada prohibida’ y ‘detención obligatoria’ (ver apéndice C), éstas pueden ser analizadas como la suma de dos contribuciones o detecciones: la correspondiente al canal rojo y la correspondiente al canal acromático de la región interior. De esta manera, el sistema consigue redundancia en la detección para aumentar su fiabilidad. Como se puede observar en las figuras 4.7(g), 4.7(h), 4.7(r) y 4.7(s), el canal rojo segmenta las señales a partir de la orla exterior mientras que el canal acromático detecta las regiones del fondo del pictograma. En el caso de señales ubicadas en tramos de obra, el fondo blanco de las señales cambia a color amarillo y por tanto, las contribuciones de la segmentación vendrán dadas en este caso por los canales rojo y amarillo (ver figuras 4.7(f) y 4.7(n)).

Bajo visibilidad nocturna, el modelo de segmentación propuesto debe ser adaptado considerando que las señales acromáticas no son bien detectadas mediante la ecuación 4.6. La razón viene dada porque de noche las señales acromáticas se perciben desde el vehículo con cierto color debido a la incidencia de la luz coloreada de los faros sobre el material reflectante de las señales. Las secuencias nocturnas de la base de datos han sido grabadas con faros delanteros de luz amarilla. Por esta razón, se propone otro canal cromático para las señales blancas en condiciones de visibilidad nocturna. En la figura 4.8 se muestran los histogramas H-S de visibilidad nocturna para los canales rojo, azul y blanco. Como se puede comprobar, las regiones blancas de las señales en condiciones nocturnas son percibidas con un tono similar al amarillo de día (ver figura 4.6(c)) debido al tipo de luz con que se ilumina, aunque con menos saturación. Es importante observar que la componente H se extiende en los histogramas H-S de visibilidad nocturna para los canales cromáticos en un rango similar a los histogramas de visibilidad diurna. No obstante, el color de las señales en visibilidad nocturna bajo iluminación de los vehículos presenta un alto grado de saturación. La diferencia entre la percepción del blanco y el amarillo por la noche estriba, prácticamente, en el nivel de la componente de saturación. En la figura 4.9 se muestra un ejemplo de segmentación bajo visibilidad nocturna en el que se ha trabajado con los umbrales fijados en la tabla 4.2. Como se puede comprobar, las máscaras de segmentación son menos ‘limpias’ que en condiciones diurnas y aparecen numerosos píxeles ruidosos.



Figura 4.7: Ejemplos de segmentación. De arriba hacia abajo, imágenes originales, máscaras de segmentación del canal rojo, máscaras de segmentación del canal azul, máscaras de segmentación del canal amarillo y máscaras de segmentación del canal acromático.

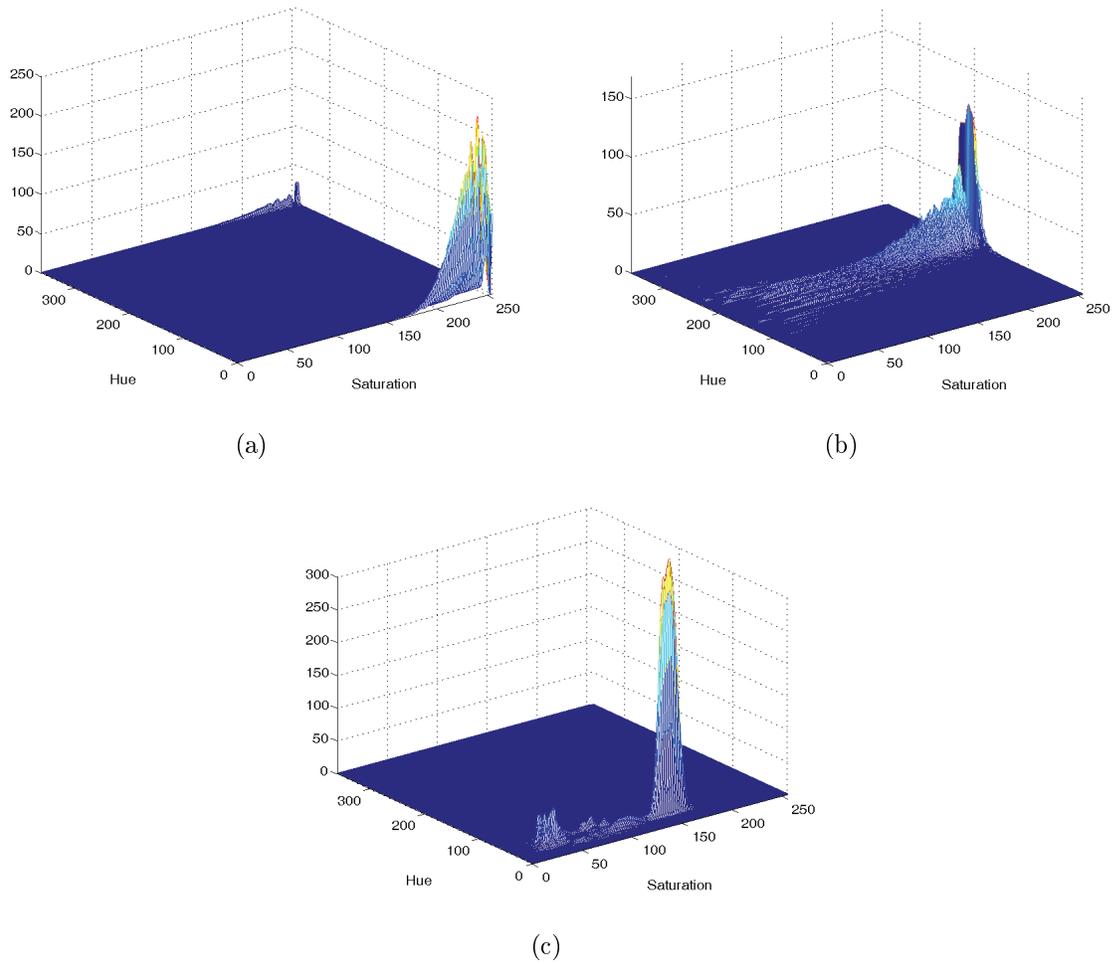


Figura 4.8: Histogramas H-S en visibilidad nocturna. (a). Canal rojo. (b) Canal azul. (c) Canal acromático.

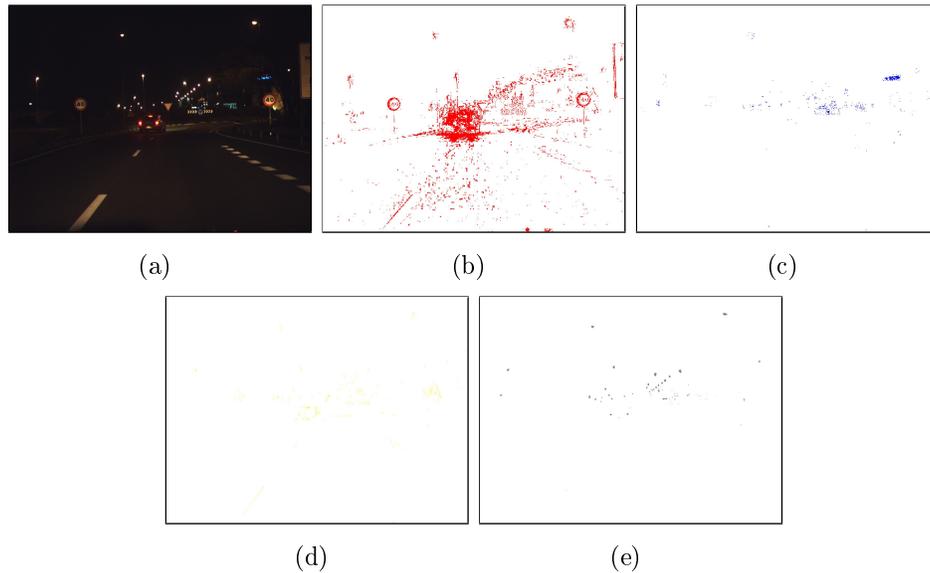


Figura 4.9: Histogramas H-S en visibilidad nocturna. (a) Imagen original. (b) Canal rojo. (c) Canal azul. (d) Canal amarillo. (e) Canal acromático.

Debido a la alta correlación espacial que existe en una imagen, los píxeles segmentados quedan agrupados frecuentemente junto a otros adyacentes en cada máscara de color constituyendo un *blob* o ROI (*Region of interest*). Tras un proceso de extracción de componentes conectadas, los objetos detectados son tratados de manera independiente por el sistema. Como puede comprobarse en las máscaras de la figura 4.7, la segmentación por color implica que además de las señales, se detecten otros objetos de la escena con colores similares a los propios de las señales. Se trata de falsas alarmas que el sistema habrá de descartar en cualquiera de los módulos de detección, clasificación o reconocimiento. No obstante, interesa eliminar las falsas alarmas en las primeras etapas para reducir el tiempo de procesado de la imagen. Específicamente, en la etapa de segmentación se ha implementado un filtro de ROIs basado en el tamaño y relación de aspecto, cuyos umbrales han sido derivados empíricamente. En concreto, se descartan aquellos objetos con un área excesivamente grande o excesivamente pequeña y/o aquellos con un *bounding-box* cuya relación de aspecto es muy diferente a la unidad. En la figura 4.10 se muestran las máscaras resultantes de la Figura 4.7 tras aplicar el proceso de filtrado a las máscaras.

Otros algoritmos de segmentación han sido analizados o propuestos para la segmentación de señales viales por el equipo investigador bajo el que se ha desarrollado esta tesis doctoral. El estudio comparativo de los diferentes métodos de segmentación se recoge en [Gomez10], incluyendo tanto métodos basados en color como métodos basados en detección de bordes. Dentro de los métodos basados en espacios de color, en [Gomez10] se han considerado los siguientes:

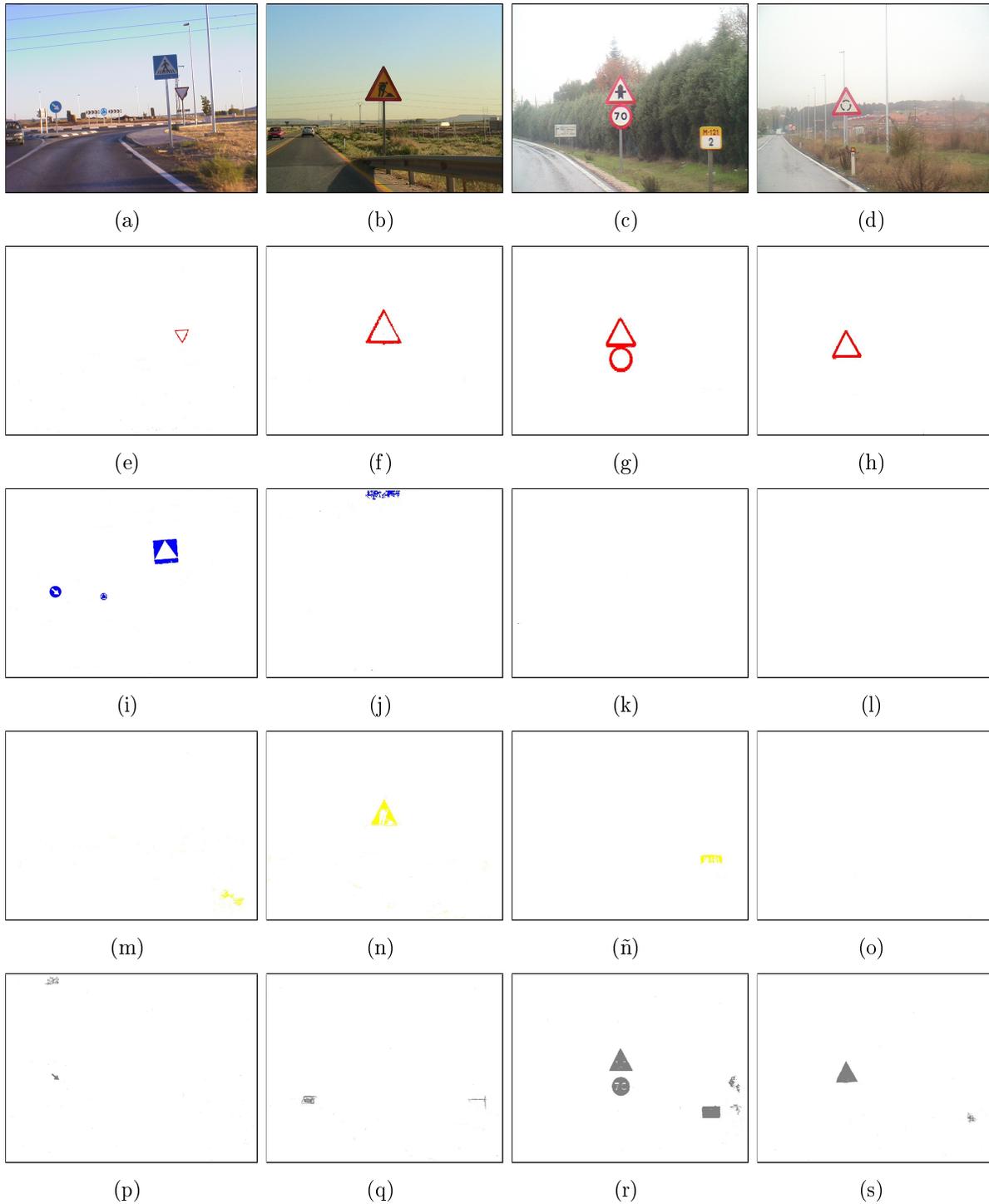


Figura 4.10: Ejemplos de segmentación tras aplicar un filtro de tamaño y relación de aspecto. (a, b, c, d). De arriba hacia abajo, imágenes originales, máscaras de segmentación del canal rojo, máscaras de segmentación del canal azul, máscaras de segmentación del canal amarillo y máscaras de segmentación del canal acromático.

- Umbralizado RGB normalizado. Se ha seguido el modelo RGB normalizado propuesto en [Kamada90] y [Janssen93], donde se reduce el impacto de la iluminación y tan sólo se necesitan dos componentes para realizar la segmentación.
- Umbralización en el espacio Ohta. El espacio de color Ohta se encuentra incluido en la familia de los espacios de color oponentes inspirados en la fisiología del sistema visual humano [Ohta80]. Sus principales propiedades son la simplicidad y la incorrelación de sus componentes.
- Umbralización en tono y saturación mejorados. Se ha utilizado el modelo [Escalera04] basado en el espacio HSI, pero para prevenir los problemas propios de los umbrales rígidos se ha propuesto un umbral suave basado en una transformación.

Respecto a los métodos basados en la detección de bordes, en [Gomez10] se han utilizado los siguientes:

- Eliminación de bordes en escala de gris. Este método es utilizado en [Aoyagi96] y se basa en en dos pasos: suavizado de la imagen y aplicación del filtro de laplaciana.
- Eliminación de bordes Canny. Entre los algoritmos de detección de bordes, el de Canny es comúnmente reconocido como referencia de comparación para muchos investigadores.
- Eliminación de bordes en color. Se ha definido en [Gomez10] un método de extracción de bordes en el espacio RGB.

Además de trabajar con los métodos anteriores, en [Gomez10] se ha propuesto una segmentación basada en SVM para evitar el ajuste de umbrales, mediante un algoritmo que emplea cuantificación de los canales de color y tablas de búsqueda para conseguir mayor velocidad de procesado. Como conclusiones del análisis comparativo realizado por el equipo investigador, cabe mencionar las siguientes:

1. No existe un método de segmentación que se comporte claramente mejor que el resto en todos los contextos.
2. El uso de tablas de búsqueda para mejorar la velocidad es efectivo consiguiendo una calidad similar al método original.
3. Los métodos de detección de bordes se pueden usar como complemento a los métodos de color, pero no de manera aislada. Es importante destacar que aunque los métodos basados en bordes son simples y rápidos, generan numerosas detecciones que sobrecargan los procesos de clasificación y reconocimiento.

### 4.3. Módulo de Clasificación de Forma

El objetivo de esta etapa es la clasificación de los objetos detectados de acuerdo a su geometría, teniendo en cuenta que las señales viales presentan formas muy definidas: circular, triangular, rectangular y octogonal. El hecho de clasificar las señales por su geometría permitirá estructurar mejor el problema de la identificación de la señal, ya que el conjunto de posibles señales para un color y una forma es relativamente reducido con respecto al conjunto total de señales. El descriptor propuesto en esta tesis para la clasificación de formas, denominado Distancia a bordes DTB (*Distance to borders*), se define a partir de la codificación de la distancia desde los bordes externos del objeto hasta el *bounding box* que lo circunscribe. En realidad, estas distancias se evalúan independientemente para cada uno de los cuatro lados del bounding box y por tanto, se obtienen cuatro vectores correspondientes a los lados. La herramienta de clasificación empleada para la clasificación de forma es una SVM lineal, construyéndose cuatro SVM lineales por cada ROI. De esta manera, en el testeo se emitirá una votación por cada uno de los cuatro vectores DTB de cada objeto y al final, para decidir la forma asignada, se aplicará un método de votación mayoritario. Así, si el número total de votos conseguido no supera un umbral establecido para ninguna de las formas, se considera que la forma del objeto no se corresponde con ninguna de las formas definidas para las señales viales y es descartado. En el caso de empate a votos entre dos o más formas, la asignación considerará aquella cuya suma de valores numéricos de las salidas de las SVM sea mayor.

En la Figura 4.11 se muestran los vectores DTB para una ROI de forma triangular, donde  $D_1$ ,  $D_2$ ,  $D_3$  y  $D_4$  representan las distancias desde los bordes hasta los lados izquierdo, derecho, superior e inferior, respectivamente. Con el objetivo de conseguir vectores invariantes a la escala de los objetos, cada vector DTB está constituido por un número fijo de muestras,  $N_s$ , realizándose un muestreo equiespaciado sobre cada uno de los lados del bounding box. En concreto,  $N_s$  ha sido fijado a 20 muestras consiguiendo resultados óptimos con SVM lineales.

Teniendo en cuenta que las señales viales tan sólo se definen para algunas combinaciones de color y forma, el color de segmentación de una ROI determina las SVM que alimenta. De esta manera, los objetos segmentados por rojo alimentan las entradas de las SVM correspondientes a las formas circular y triangular, el canal azul sirve de entrada a las SVM de las formas rectangular y circular y finalmente, los objetos detectados por blanco y amarillo alimentan las SVM de las formas circular y triangular, considerando que la segmentación del fondo de la señal se puede efectuar por blanco o amarillo. La distinción entre las formas circular y octogonal provoca numerosas confusiones que han llevado a incluir la forma octogonal como circular y por tanto, la señal de ‘Stop’ será tratada como una señal circular más.

Las características del método DTB consiguen una alta invarianza frente a la traslación, a la rotación y al escalado, que se justifican del siguiente modo:

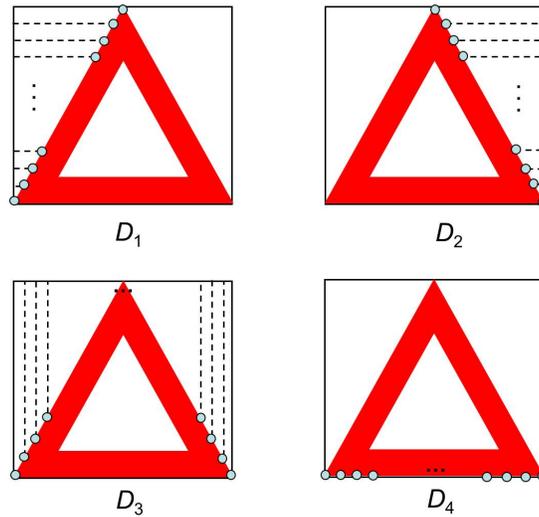


Figura 4.11: Distancia a bordes de una forma triangular.

- La invarianza a la traslación queda resuelta al extraer los objetos de la imagen y obtener los vectores DTB correspondientes al bounding box. En la Figura 4.12 se muestra la alta similitud de los vectores DTB de dos señales situadas en diferentes ubicaciones del plano imagen.
- La invarianza a los cambios de escala (ver Figura 4.13) se consigue mediante un muestreo equiespaciado en cada lado del bounding box. Fijando el número de muestras de los vectores DTB, independientemente de las dimensiones del bounding box, se consigue que todos los vectores tengan la misma longitud.
- La invarianza a la rotación se resuelve en gran medida porque antes de obtener los vectores DTB se realiza una búsqueda de los píxeles más externos y el objeto se orienta según una posición de referencia (ver Figuras 4.14 y 4.15).

En la figura 4.16 se muestran los objetos detectados en 4.10 de acuerdo a la clasificación de su forma geométrica. Como se puede comprobar para estos ejemplos, existen falsas alarmas que no son eliminadas por las dos primeras etapas y alcanzan la entrada del posterior módulo de reconocimiento.

En paralelo al descriptor DTB, el grupo de investigación GRAM de la UAH, bajo el que se ha desarrollado esta tesis, ha propuesto un descriptor alternativo para la clasificación de forma basado en la firma del objeto, definiéndose la firma como una representación unidimensional del contorno de la ROI [Gil08]. La firma del objeto codifica la distancia desde el centro del objeto al contorno como una función del ángulo. Al realizar un muestreo equiespaciado de la firma, se consigue un número fijo de muestras igual a  $360^\circ$  entre el incremento de ángulo entre muestras consecutivas. La firma generada por este procedimiento no varía con la traslación, pero depende de la escala y la rotación. La normalización con respecto a la escala se puede conseguir normalizando la firma obtenida, ya que los

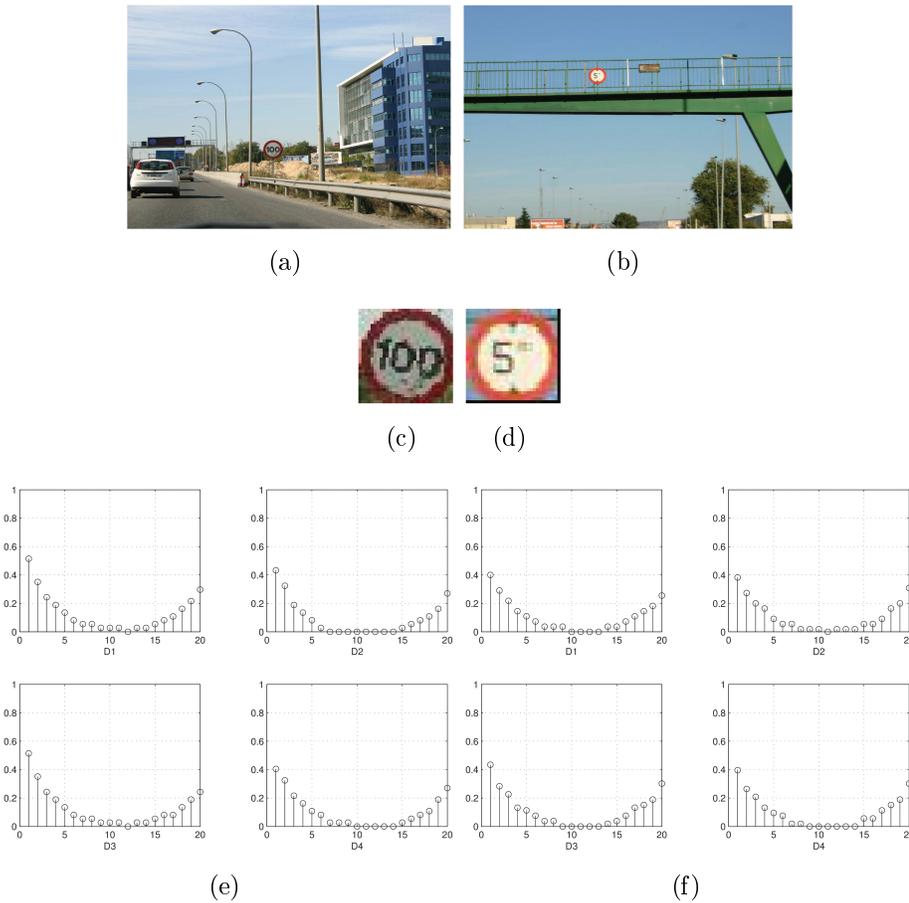


Figura 4.12: Invarianza a la traslación. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d).

cambios de tamaño de un objeto producen cambios en los valores de amplitud de la firma correspondiente. En cualquier caso, la invarianza a los cambios de escala se obtiene normalizando la energía o la amplitud de la señal resultante. Finalmente, las rotaciones de los objetos se traducen en desplazamientos circulares en la firma. Para hacer el algoritmo de clasificación invariante a las rotaciones, en lugar de comparar directamente la propia firma, la comparación se realiza a partir del valor absoluto de la transformada discreta de Fourier (DFT, *Discrete Fourier Transform*), aprovechando la propiedad de invarianza al desplazamiento del módulo de la DFT. En las Figuras 4.17, 4.18 y 4.19 se muestran las firmas de referencia para un círculo, un triángulo y un cuadrado, respectivamente.

## 4.4. Módulo de Reconocimiento

El reconocimiento del pictograma de una señal de tráfico queda incluido en el contexto del campo del reconocimiento de patrones. En esta tarea existen dos puntos cruciales a

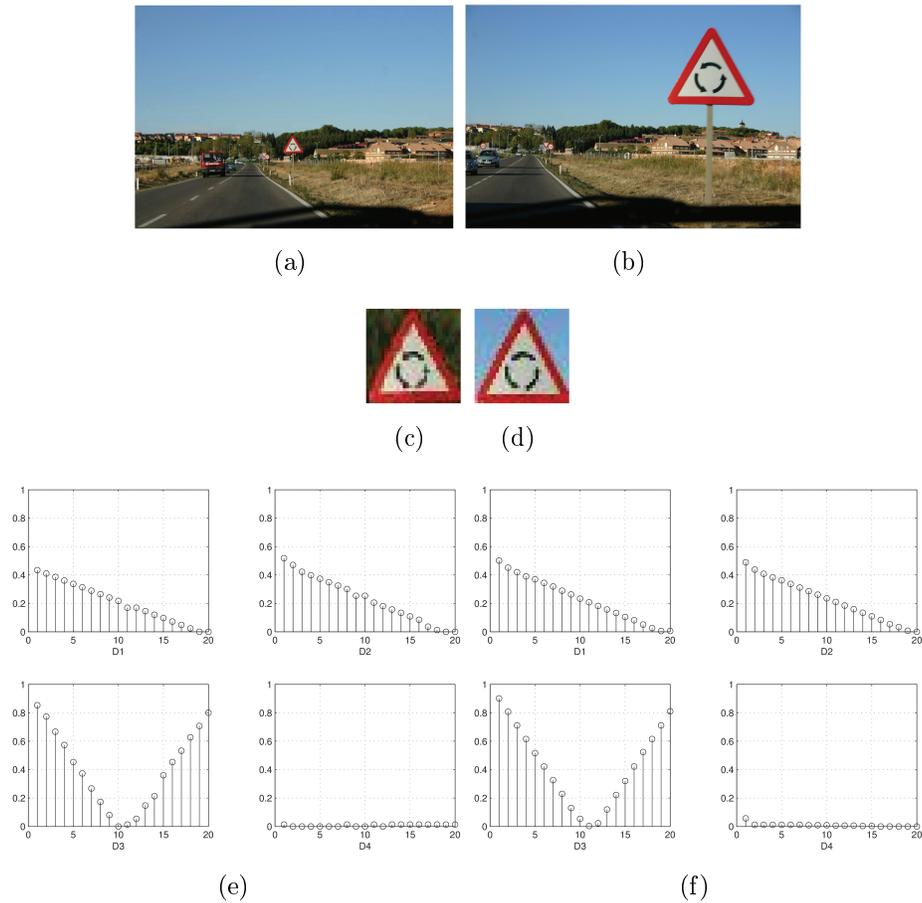


Figura 4.13: Invarianza a la escala. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d).

considerar: 1) la extracción de un descriptor que caracterice el objeto y 2) la implementación de un clasificador para el reconocimiento.

#### 4.4.1. Vector descriptor

Aunque existen numerosas posibilidades de construir un descriptor utilizando diferentes características, la calidad del mismo viene dada por su eficiencia y capacidad discriminativa. La idea básica consiste en encontrar la información relevante y codificarla con el menor número de características, intentando que sea robusta en la mayor medida posible a los problemas derivados de entornos naturales, tales como cambios de luz, el deterioro del material del objeto y las posibles oclusiones parciales. El primer descriptor utilizado en esta tesis doctoral se basa en la concatenación de la luminancia de los píxeles. Como quedará demostrado a través de los resultados, la luminancia del pictograma es suficiente para la identificación de la señal. Debido a que los niveles de luminancia de las regiones rojas y azules de las señales son similares, se propone una variación correctora sobre los

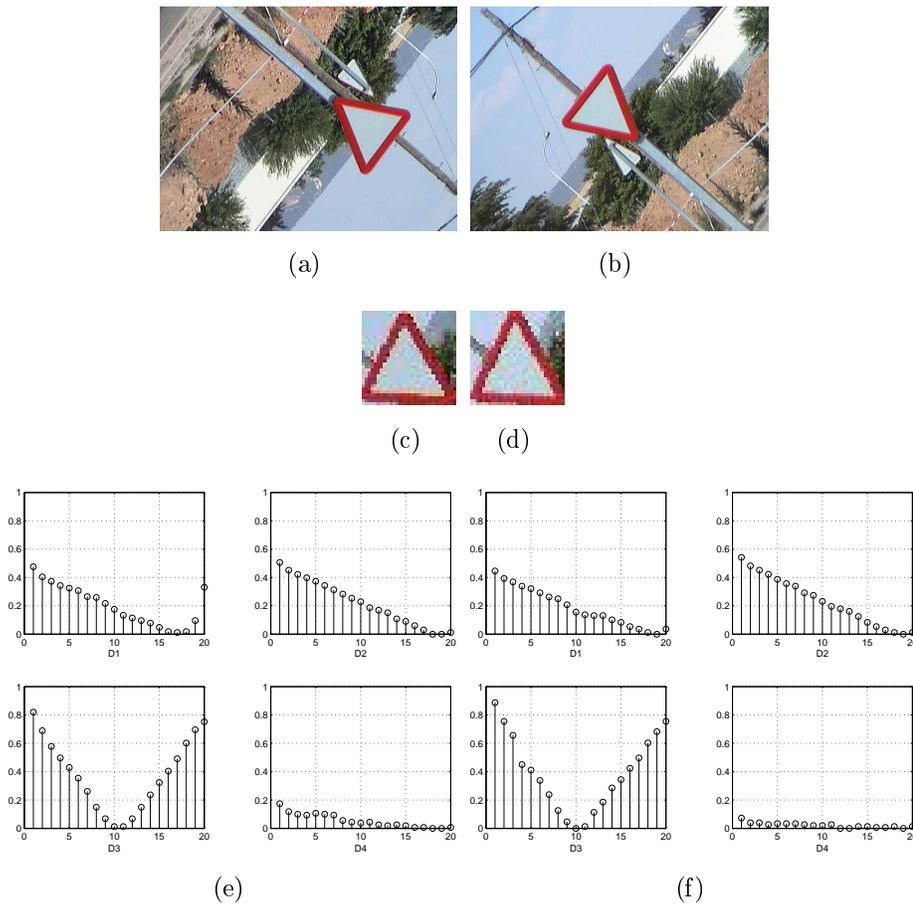


Figura 4.14: Invarianza a la rotación 2D. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d).

coeficientes multiplicativos de los canales RGB, obteniendo la luminancia de acuerdo a la siguiente ecuación:

$$Y = 0.5 \times R + 0.3 \times G + 0.2 \times B. \quad (4.8)$$

Los objetos 2D son representados como un punto de un espacio multidimensional (espacio transformado) en el que cada componente se corresponde con un píxel de la imagen. La recopilación de varios descriptores constituye una nube de puntos en el espacio transformado.

En los blobs circulares y triangulares existen zonas correspondientes al fondo de la escena que no aportan información. Con el objetivo de extraer únicamente los píxeles de interés de la señal, se definen dos máscaras (ver Figura 4.20). Para cada objeto de testeo se considera la información de los píxeles que caen dentro de la zona de interés de la máscara (*PoI-Pixels of Interest*) marcados en blanco. En la figura 4.21 se muestra el proceso de extracción del vector descriptor. Como los blobs procedentes de una imagen

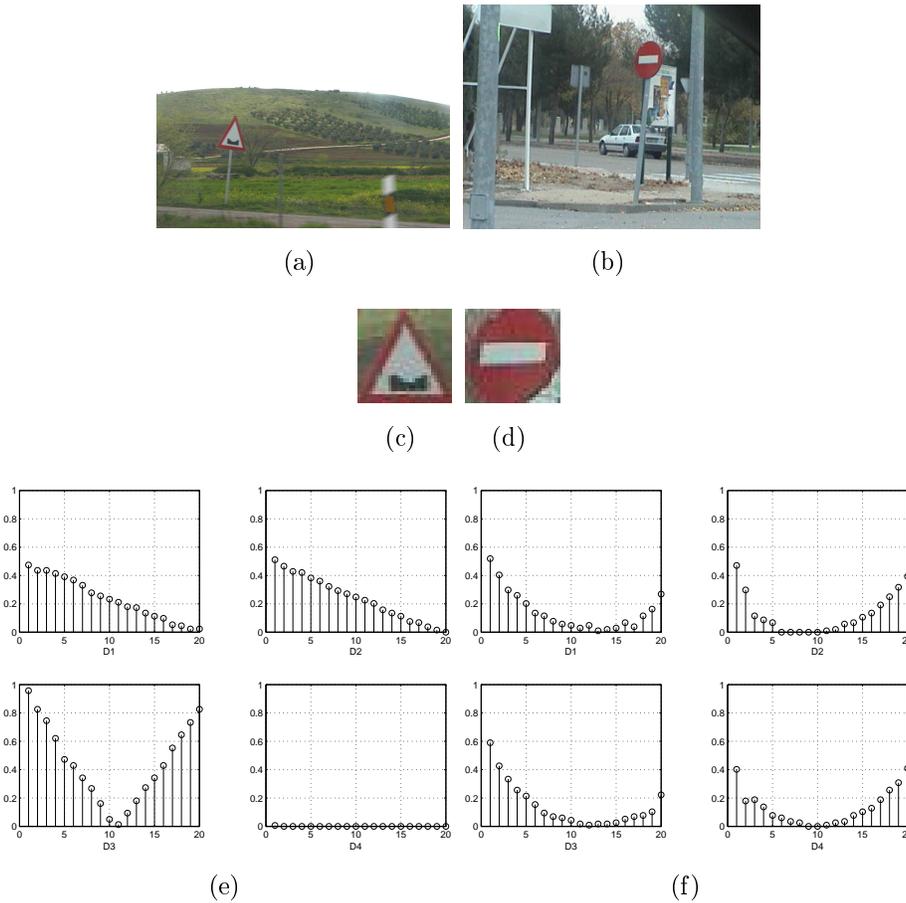


Figura 4.15: Invarianza a la rotación 3D. (a,b) Imágenes originales. (c,d) Blobs de (a,b). (e,f) Vectores DTB de (c,d).

pueden tener tamaños muy diversos, se aplica una normalización de escala a un tamaño de  $31 \times 31$  píxeles. En el caso de las señales rectangulares no existen píxeles de fondo y la longitud del vector descriptor es de 961 componentes, mientras que en el caso de las señales circulares y triangulares el número de componentes queda reducido a 709 y 511, respectivamente. En la figura 4.22 se muestran algunos ejemplos de blobs normalizados en tamaño para su correspondiente reconocimiento.

#### 4.4.2. Clasificador

A partir del descriptor que caracteriza la apariencia de un objeto, es necesario disponer de un clasificador que sea capaz de identificar el pictograma. En esta tesis se propone el uso de las SVM como herramienta de clasificación por su buena capacidad de generalización, planteando por cada combinación de color y forma un problema de multclasificación. La arquitectura utilizada en la etapa de reconocimiento se muestra integrada en la figura 4.23 con las etapas de segmentación y clasificación, donde puede verse que cada ROI

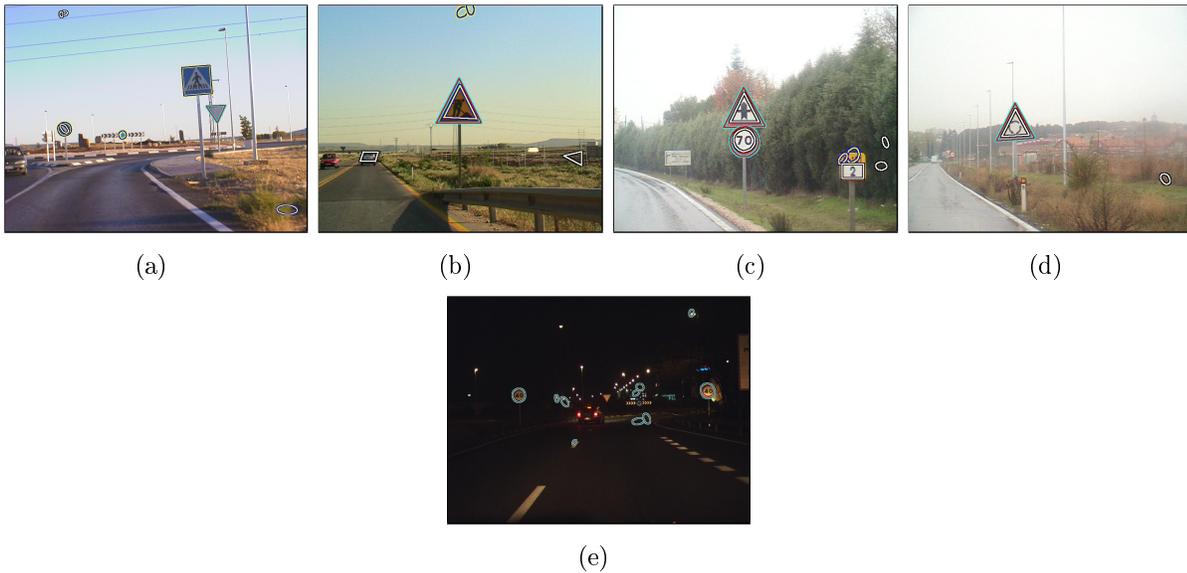


Figura 4.16: Ejemplos de clasificación de los objetos detectados en 4.10.

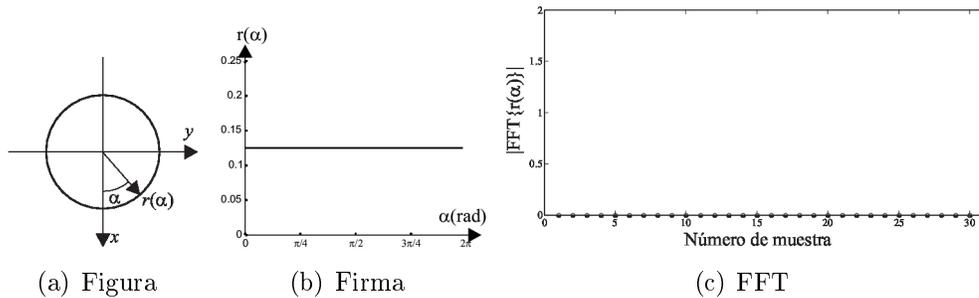


Figura 4.17: Firma y módulo de la FFT para un círculo.

alimenta la entrada de los clasificadores que se corresponden exclusivamente con su color y forma. Por simplicidad en la figura, sólo se ha representado la máscara del canal rojo, entendiendo que el resto de máscaras no aportan blobs candidatos. De este modo, para cada combinación específica de color y forma sólo son posibles algunas clases del conjunto total y así por ejemplo, para un objeto segmentado por color rojo y caracterizado por forma circular son posibles las categorías de las señales de prohibición. La complejidad de la tarea de reconocimiento se ve reducida significativamente respecto al caso en el que se contemplaran todas las combinaciones de color y forma simultáneamente.

El fundamento de una SVM binaria parte de un conjunto de datos de entrenamiento, los cuales vienen dados por  $N$  vectores  $\mathbf{x}_i \in \mathbb{R}^d$  con etiquetas  $y_i \in \{-1, +1\}$ , para buscar un hiperplano que sea capaz de separar las clases positiva (+1) y negativa (-1). Para el caso no lineal, como el que nos ocupa, la función de decisión de las SVM se

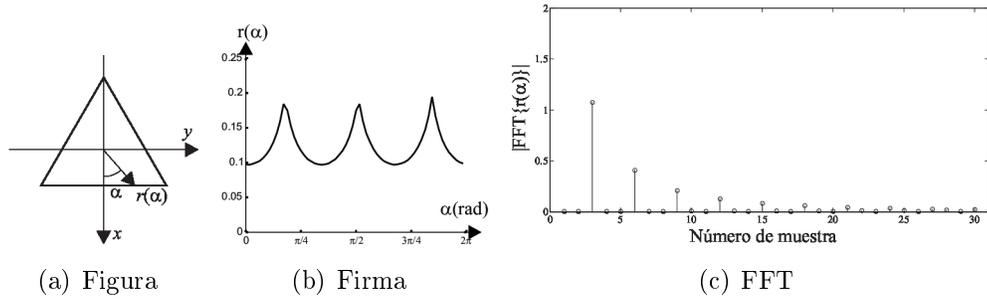


Figura 4.18: Firma y módulo de la FFT para un triángulo equilátero.

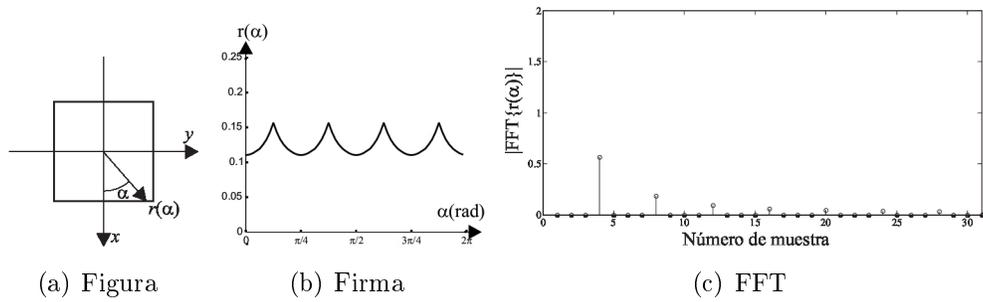


Figura 4.19: Firma y módulo de la FFT para un cuadrado.

calcula mediante un kernel  $K()$  que realiza el mapeado del espacio de entrada al espacio de características mediante la siguiente ecuación:

$$f(\mathbf{x}) = \sum_{s=1}^{N_{sv}} \alpha_i y_i K(\mathbf{x}_s, \mathbf{x}) + b, \tag{4.9}$$

donde  $f(\mathbf{x})$  es el valor de la función de decisión,  $\mathbf{x}$  el vector de test,  $\mathbf{x}_s$  cada uno de los vectores soporte,  $b$  el sesgo,  $\alpha_i$  los multiplicadores de Lagrange y  $N_{sv}$  el número de vectores soporte (ver con más detalle el apéndice B). En el problema a tratar, se ha seleccionado el mejor kernel sobre un conjunto de kernels polinómicos (variando el grado de 1 a 6) y

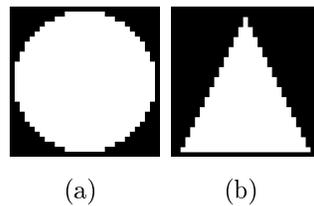


Figura 4.20: Máscaras para eliminación de fondo. (a) Máscara circular. (b) Máscara triangular

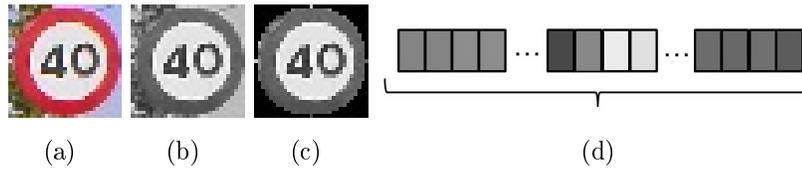


Figura 4.21: Obtención del vector descriptor. (a) Blob en color. (b) Conversión del blob a nivel de gris. (c) Eliminación de los píxeles del fondo. (d) Vector descriptor.



Figura 4.22: Ejemplos de muestras. (a) Señales. (b) Ruidos.

el kernel RBF, siendo éste último el que mejores resultados ha aportado. El kernel RBF se define mediante la ecuación:

$$K(\mathbf{x}_s, \mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}_s - \mathbf{x}\|^2}{\sigma^2}\right), \quad (4.10)$$

y es definido por dos parámetros:  $C$  y  $\gamma$ , donde  $C$  es el parámetro de coste y  $\gamma$  la inversa de  $\sigma^2$  en la ecuación 4.10.

Aunque existen varias arquitecturas posibles para resolver los problemas de multclasificación, tal como se abordará en el siguiente capítulo, la propuesta inicial que se ha elegido por su sencillez es la estrategia ‘*uno contra todos*’ (OVA-*One versus All*) [Hsu02]. En la estrategia OVA se construyen tantas SVM binarias como  $M$  clases existan en el problema. En el entrenamiento del clasificador  $m$ -ésimo se consideran como muestras positivas los patrones de su propia clase con etiquetas  $y_i = +1$  y el resto de patrones, incluyendo los de ruido, se etiquetan como negativos con  $y_i = -1$ . Debido a la dispar apariencia de los falsos positivos (ver ejemplos de muestras en la figura 4.22(b)), el ruido no ha sido considerado como una categoría en esta primera aproximación y por tanto, no ha habido necesidad de construir un clasificador explícito para el mismo.

En esta tesis, los conjuntos de entrenamiento y test para cada combinación de color y forma han sido definidos a partir de la base de datos, de tal manera que para el entrenamiento se han seleccionado aleatoriamente sin reemplazo la mitad de las muestras y con las restantes se ha conformado el conjunto de test. Esta partición se ha realizado cinco veces y los resultados se han obtenido promediando los resultados parciales de las cinco iteraciones. En cada partición, a partir del conjunto de entrenamiento, se han buscado

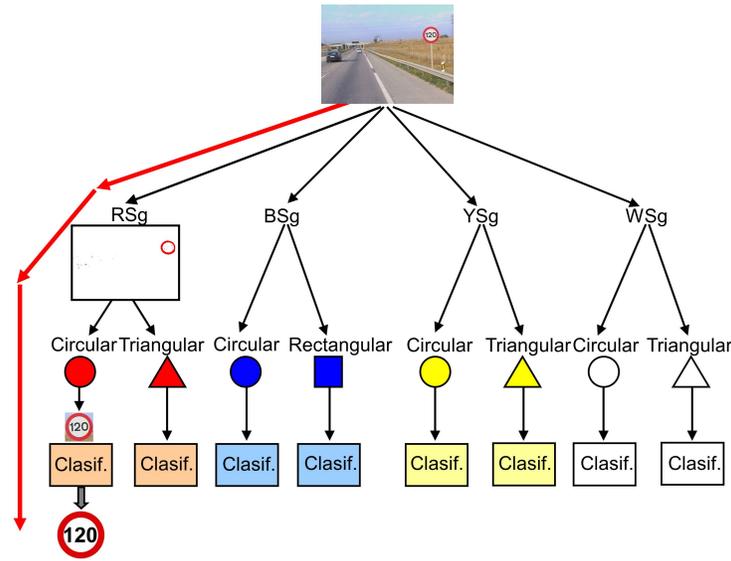


Figura 4.23: Diagrama funcional del sistema completo con un ejemplo de detección y reconocimiento.

los valores óptimos de los parámetros  $C$  y  $\gamma$  con secuencias de crecimiento exponencial en los rangos  $C = \{2^{-5}, 2^{-3}, \dots, 2^{15}\}$  y  $\gamma = \{2^{-15}, 2^{-13}, \dots, 2^3\}$ , llegando a analizar  $21 \times 19$  combinaciones mediante la librería LIBSVM [Chang00]. La búsqueda de los mejores parámetros se realiza considerando una minimización del número de errores en el proceso de entrenamiento utilizando una validación cruzada de 5 iteraciones, lo cual quiere decir que el conjunto de entrenamiento de cada partición se divide a su vez en 5 sub-conjuntos y uno de ellos se utiliza en cada iteración para buscar los mejores parámetros después de entrenar con los cuatro sub-conjuntos restantes de entrenamiento. Finalmente, después de obtener los valores óptimos  $(C^*, \gamma^*)$  se realiza el entrenamiento en cada partición sobre el conjunto completo de entrenamiento. En la tabla 4.3 se muestran la pareja  $(C^*, \gamma^*)$  para cada combinación de color y forma.

El hecho de seleccionar aleatoriamente la mitad de las muestras para el conjunto de entrenamiento y la otra mitad para el conjunto de test, exige que cada clase contenga al menos dos muestras, siendo descartadas aquellas clases con tan sólo una muestra. En la tabla 4.4 se muestra la información para los procesos de entrenamiento y test, incluyendo a partir de la base de datos original el número de clases útiles que aportan más de dos muestras así como el número total de muestras de señales útiles más el número de muestras de ruido. Del conjunto total de muestras de entrenamiento, sólo algunas constituyen los vectores soporte que definen el hiperplano del clasificador. A modo de ejemplo, en la Figura 4.24 se muestran los vectores soporte positivos y negativos de la clase ‘Limitación de velocidad a 40 Km/h’. Hay que considerar que un mismo patrón puede ser un vector soporte común a varias clases. Así, por ejemplo, una muestra de la clase ‘Stop’ puede ser un vector soporte positivo para la máquina de la clase ‘Stop’ y a la vez un vector

<b>Combinación (Color-Forma)</b>	<b>C*</b>	<b><math>\gamma^*</math></b>
Rojo-Circular	32	0.03125
Rojo-Triangular	8	0.03125
Azul-Circular	32	0.03125
Azul-Rectangular	128	0.0078125
Blanco-Circular	8	0.03125
Blanco-Triangular	100	0.02
Amarillo-Circular	8	0.03125
Amarillo-Triangular	100	0.02

Tabla 4.3: Valores óptimos de los parámetros de las SVM.

soporte negativo para muchas otras clases, tal como ocurre en la Figura 4.24 para la clase ‘Limitación de velocidad a 40 Km/h’.

<b>Combinación (Color-Forma)</b>	<b>#Clases</b>	<b>#Clases útiles</b>	<b>#Muestras</b>	<b>#Muestras útiles</b>
Rojo Circular	62+1	58+1	14536+3895	14511+3895
Rojo Triangular	45+1	45+1	9246+3066	9246+3066
Azul Circular	54+1	10+1	2160+610	2116+610
Azul Rectangular	99+1	39+1	2709+890	2649+890
Blanco Circular	114+1	55+1	6695+1574	6636+1574
Blanco Triangular	44+1	43+1	8133+778	8132+778
Amarillo Circular	47+1	10+1	374+1162	337+1162
Amarillo Triangular	26+1	12+1	415+152	401+152

Tabla 4.4: Configuración de los datos para entrenamiento y test.

De acuerdo a las ecuaciones 4.9 y 4.10, la función de test de la clase  $m$ -ésima exige para cada vector de entrada la evaluación de la distancia entre el vector de test  $\mathbf{x}$  y cada uno de los  $N_{sv}$  vectores soporte  $\mathbf{x}_s$  que conforman el clasificador. Así, el valor de la función de decisión de la clase  $m$ -ésima, dentro de una combinación de color y forma, vendrá dado por:

$$f_m(\mathbf{x}) = b + \sum_{s=1}^{N_{sv}} \alpha_s y_s \exp\left(-\frac{\|\mathbf{x}_s - \mathbf{x}\|^2}{\sigma^2}\right). \quad (4.11)$$

De este modo, cada clasificador emite una respuesta numérica  $f_m(\mathbf{x})$  ante una entrada  $\mathbf{x}$  según la ecuación 4.11 y la asignación de la etiqueta de predicción se realiza de acuerdo

Combinación (Color-Forma)	$N_{sv}$		
	Total	Promedio por clase	Desv. Estándar
Rojo Circular	10142	176	127
Rojo Triangular	6615	147	57
Azul Circular	437	44	29
Azul Rectangular	1540	39	20
Blanco Circular	4955	90	46
Blanco Triangular	3224	75	66
Amarillo Circular	504	50	13
Amarillo Triangular	253	21	7

Tabla 4.5: Información sobre los vectores soporte de los clasificadores.

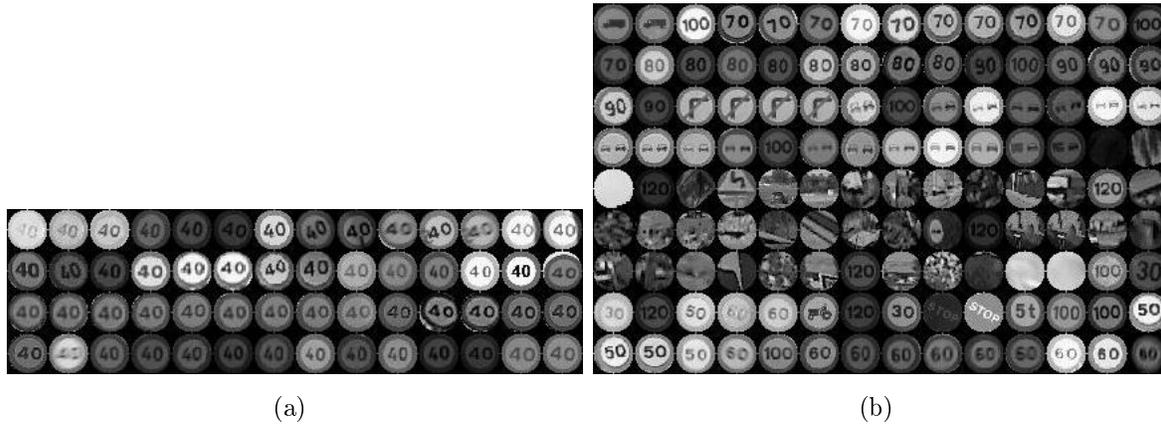


Figura 4.24: Vectores soporte de la clase ‘Limitación de velocidad a 40 Km/h’. (a) Vectores soporte positivos. (b) Vectores soporte negativos.

al mayor valor de salida como:

$$\hat{y} = \operatorname{argmax}_{m=1,\dots,M} [f_m(\mathbf{x})], \quad (4.12)$$

siempre y cuando el mayor valor de salida supere un cierto umbral. En caso contrario, las muestras de entrada son descartadas como falsas alarmas.

Según la ecuación 4.11, la carga computacional de cada clasificador incrementa de manera proporcional al número de vectores soporte que define el hiperplano de decisión entre clases y a la dimensión de los vectores. Para demostrar la influencia que tienen ambos parámetros sobre la carga de evaluación de la SVM, se ha analizado el tiempo de testeo de un vector de test sobre un rango de variación de ambos parámetros y en la gráfica 4.25 se muestra la evolución exponencial del tiempo de testeo de un vector a medida que aumenta el número de vectores soporte y por otra parte, la evolución, prácticamente

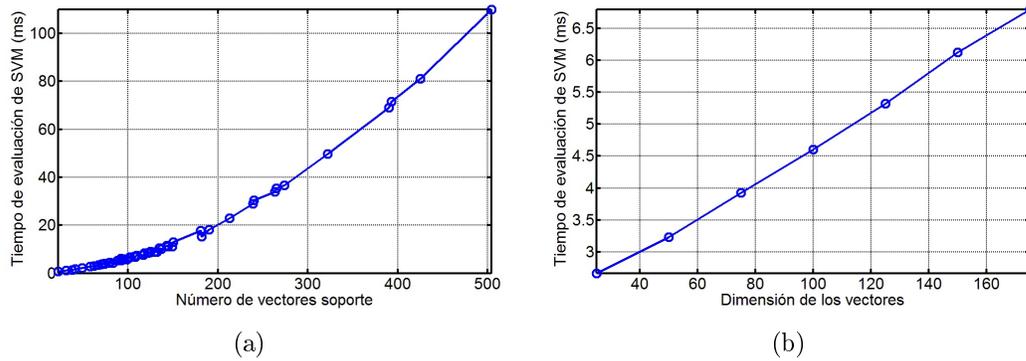


Figura 4.25: Análisis de la influencia del número de vectores soporte y la dimensión en el tiempo de evaluación de las SVM. (a) Tiempo de testeo en función del número de vectores soporte. (b) Tiempo de testeo en función de la dimensión de los vectores.

lineal, a medida que aumenta el número de características de los vectores. En la tabla 4.5 se muestra la suma del número de vectores soporte que definen las SVM por cada combinación de color y forma, así como el promedio del número de vectores soporte y la desviación típica por clase. De la observación de esta tabla, se infiere que el número de vectores soporte depende del número de clases a manejar y del número de muestras de entrenamiento. Además, los altos valores de desviación típica nos dan idea de la disparidad en el número de vectores soporte entre clases.

A efectos prácticos, la escala de salida de diferentes SVM es distinta y no es estrictamente correcto comparar sus salidas directamente [Mayoraz99]. En esta tesis se ha utilizado una implementación mejorada [Lin07] de la probabilidad a posteriori de Platt a partir de las salidas de las SVM [Platt00] para disponer de una medida de confianza en la decisión. A pesar de que no existe una interpretación probabilística clara de la salida de las SVM, las necesidades prácticas han llevado a algunos autores a plantear diversos procedimientos para estimar las probabilidades a posteriori para cada una de las  $M$  clases,  $P(y = m|\mathbf{x} \forall m = 1, 2, \dots, M)$ , basándose en la magnitud de salida como medida de confianza. En [Platt00] se propone un modelo paramétrico para la obtención de la probabilidad binaria a posteriori mediante el uso de una función sigmoide, tal como:

$$P(y = +1|f_m(\mathbf{x})) = \frac{1}{1 + \exp(Af_m(\mathbf{x}) + B)} \quad (4.13)$$

y

$$P(y = -1|f_m(\mathbf{x})) = 1 - P(y = +1|\mathbf{x}), \quad (4.14)$$

donde  $f_m(\mathbf{x})$  es el valor de salida de la SVM  $m$ -ésima para el vector de entrada  $\mathbf{x}$ . Los parámetros  $A$  y  $B$  ajustan la sigmoide maximizando la verosimilitud del conjunto de

entrenamiento de  $N$  muestras como:

$$\operatorname{argm\acute{a}x}_{A,B} \left\{ \sum_{n=1}^N (t_n \log(p_n) + (1 - t_n) \log(1 - p_n)) \right\}, \quad (4.15)$$

donde  $p_n = P(y_n = +1 | f_m(\mathbf{x}_n))$ . El conjunto de entrenamiento para el ajuste de los parámetros del modelo de Platt es el mismo que el utilizado para el entrenamiento de las SVM y, por tanto,  $f_m(\mathbf{x}_n)$  es la salida de la SVM ante la muestra de entrenamiento  $\mathbf{x}_n$ . El entrenamiento de las SVM causa que las salidas  $f_m(\mathbf{x}_n)$  sean una estimación sesgada de la distribución de  $f_m$ . Precisamente, para evitar el sobreajuste del conjunto de calibración, el modelo de Platt usa como etiquetas las muestras de entrenamiento  $t_n^+$  y  $t_n^-$  (a cambio de +1 y -1, respectivamente) para las  $N_+$  muestras positivas y las  $N_-$  muestras negativas, con:

$$t_n = \begin{cases} \frac{N_++1}{N_++2}, & \text{si } y_n = +1 \\ \frac{1}{N_-+2}, & \text{si } y_n = -1. \end{cases} \quad (4.16)$$

En la figura 4.26 se describe gráficamente la aplicación del algoritmo de Platt para la SVM que distingue la clase ‘Incorporación por la derecha’ del resto de señales rojas triangulares, representándose en la figura 4.26(a) los histogramas estimados de la distribución de las muestras positivas y negativas según los valores de salida  $f_m(\mathbf{x})$  de la SVM. La curva sólida representa  $P(f_m(\mathbf{x}) | y = +1)$  y la discontinua  $P(f_m(\mathbf{x}) | y = -1)$ . Utilizando el histograma de muestras positivas, la probabilidad a posteriori  $P(y = +1 | f_m(\mathbf{x}))$  puede estimarse mediante el teorema de Bayes como:

$$P(y = +1 | f_m(\mathbf{x})) = \frac{P(f_m(\mathbf{x}) | y = +1) P(y = +1)}{P(f_m(\mathbf{x}) | y = +1) P(y = +1) + P(f_m(\mathbf{x}) | y = -1) P(y = -1)}, \quad (4.17)$$

donde  $P(y = +1)$  es la probabilidad de las muestras positivas y puede estimarse a partir del conjunto de entrenamiento. Los resultados de aplicar el teorema de Bayes sobre el problema considerado pueden verse en la figura 4.26(b), donde se puede apreciar que para valores muy negativos de  $f_m(\mathbf{x})$  la probabilidad de que la muestra pertenezca a la clase positiva es prácticamente nula, mientras que para valores muy positivos la probabilidad tiende a la unidad. A partir de la evolución de la probabilidad a posteriori del clasificador  $m$ -ésimo, ésta puede aproximarse mediante una función sigmoide de acuerdo a la ecuación 4.13. Como resultado de la aplicación del método de Platt, en la figura 4.26(c) se muestra la función sigmoide obtenida con  $A=-4.33$  y  $B=0.25$  y el valor de salida para una probabilidad a posteriori  $P(y = +1 | f_m(\mathbf{x}))=0.5$  se corresponde con  $f_m(\mathbf{x})=0.05$ . En la figura 4.27 se muestra la familia de sigmoides de ajuste de las diez primeras señales rojas triangulares, donde se puede observar el desplazamiento de las sigmoides alrededor de  $f_m(\mathbf{x}) = 0$ . El valor de salida de la SVM que consigue  $P(y = +1 | f_m(\mathbf{x})) = 0.5$  se puede deducir a partir de la ecuación 4.13, siendo:

$$f_m(\mathbf{x})|_{P(y=+1)=0.5} = \frac{-B}{A}. \quad (4.18)$$

Si se desea usar la información proporcionada por el resto de las SVM binarias en el cálculo de la probabilidad a posteriori, se puede aplicar la función *softmax* sobre las salidas de las SVM (ver sección 6.9 de [Bishop95]):

$$P(y = m|\mathbf{x}) = \frac{\exp(\gamma f_m(\mathbf{x}))}{\sum_{j=1}^M \exp(\gamma f_j(\mathbf{x}))}, \quad (4.19)$$

donde el parámetro  $\gamma$  se estima maximizando la verosimilitud de las muestras de entrenamiento.

$$\operatorname{argmáx}_{\gamma} \left\{ \sum_{n=1}^N \sum_{m=1}^M (t_n^m \log(P(y = m|\mathbf{x}))) \right\}. \quad (4.20)$$

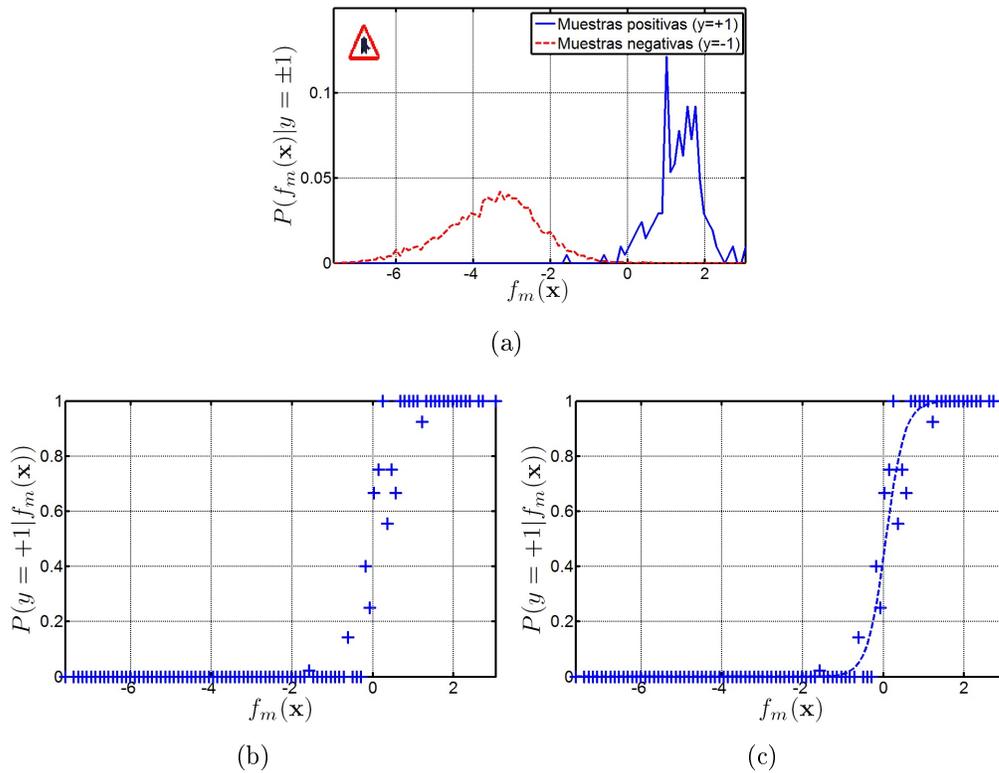


Figura 4.26: Ejemplo del ajuste de la probabilidad de salida para la clase ‘Incorporación por la derecha’ según el método de Platt[Platt00]. (a) Histogramas de la probabilidad a priori. (b) Probabilidad a posteriori de las muestras positivas. (c) Ajuste de la sigmoide.

En la tabla 4.6 se muestran los resultados obtenidos al comparar directamente las salidas de los clasificadores, referidas como OVA-f, con las estimaciones de probabilidad

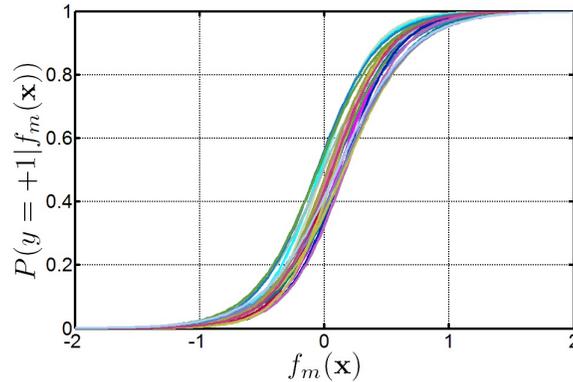


Figura 4.27: Familia de sigmoides de ajuste para las 10 primeras clases del conjunto de señales rojas triangulares.

de Platt denotadas por OVA-P. Para cada combinación, se aporta la TPR habiendo fijado  $FPR=5\%$  y el promedio del tiempo de carga por muestra  $T_t$ , el cual se puede desglosar en la suma del tiempo de obtención del descriptor  $T_{desc}$  y el tiempo de clasificación de las SVM  $T_{svm}$ . Como se puede comprobar, aunque la tasa de acierto es similar en ambos métodos, en la mayor parte de las combinaciones, en concreto en 6 de 8, los resultados son superiores mediante la estimación de probabilidad a posteriori. El método de estimación de probabilidad requiere un tiempo de procesamiento ligeramente superior respecto a la comparación directa de las salidas de las SVM, tanto en el entrenamiento para el ajuste de los parámetros de la sigmoide como en el test para la estimación de probabilidad a partir de las salidas de las SVM. En cualquier caso, el método no busca tanto la mejora de cada clasificador binario, sino la correcta comparación entre las salidas. Hay que hacer notar que si el problema es claramente separable, los parámetros de la sigmoide tenderán a  $A \rightarrow \infty$  y  $B \rightarrow 0$ . De esta forma, se puede entender que cuando todos los parámetros de las sigmoides tienden a estos valores, el método de Platt converge con la comparación directa de las salidas de las SVM, ya que las sigmoides se convierten en funciones escalón. Por esta razón, se justifica que no existe una gran diferencia entre ambos métodos a pesar de que la comparación directa de las salidas de las SVM no sea matemáticamente correcta.

## 4.5. Resultados del sistema

Para testear la calidad conjunta de los módulos de segmentación, clasificación de forma y reconocimiento, los cuales han sido implementados en lenguaje C, el sistema RESET ha sido testeado con cinco secuencias grabadas en el Corredor del Henares, en concreto, en el eje que discurre entre el Campus Universitario de la Universidad de Alcalá y Meco. Cada una de las secuencias cubre aproximadamente una distancia de 6 kilómetros en diferentes condiciones, siendo el tamaño de las imágenes capturadas de  $720 \times 576$  píxeles y habiéndose fijado un tiempo entre capturas consecutivas de 0.2 s. Las secuencias  $S_1$ ,  $S_2$  y

Combinación (Color-Forma)	OVA-f		OVA-P	
	TPR(%)	T <sub>t</sub> (ms)	TPR(%)	T <sub>t</sub> (ms)
Rojo Circular	96.3	612.3	96.4	657.7
Rojo Triangular	96.2	216.3	96.4	273.2
Azul Circular	96.9	20.5	98.4	25.4
Azul Rectangular	97.9	80.4	97.4	87.9
Blanco Circular	97	197.9	97.4	215.2
Blanco Triangular	82.8	98.9	86.2	101.2
Amarillo Circular	91.7	24.2	92.4	24.3
Amarillo Triangular	95.9	9.7	93.1	10.1

Tabla 4.6: Resultados del módulo de reconocimiento utilizando la base de datos.

$S_3$  corresponden a días soleados, la secuencia  $S_4$  a un día lluvioso y la secuencia  $S_5$  ha sido grabada en horario nocturno. Para el conjunto de estos experimentos se ha utilizado la umbralización HSI como algoritmo de segmentación y la distancia a bordes como algoritmo de clasificación de forma. La tabla 4.7 recoge los resultados del test incluyendo la siguiente información: 1) número de imágenes de la secuencia, 2) número de señales de tráfico que aparecen en la secuencia, 3) número de detecciones correctas considerando que cada señal puede ser detectada dos veces en una misma imagen y además en varias imágenes de la secuencia, 4) número de falsas detecciones a la entrada del módulo de reconocimiento, 5) número de falsos positivos a la salida de la etapa de reconocimiento y 6) número de confusiones entre clases. Las confusiones entre clases son atribuibles al reconocimiento de ROIs de baja calidad, lo cual ocurre, principalmente, cuando la iluminación es escasa y cuando el tamaño de la señal es muy reducido en el plano imagen debido a una distancia grande entre la señal y la cámara. En cualquier caso, es importante destacar que todas las señales de las secuencias son correctamente identificadas al menos 2 veces y que las falsas alarmas no se repiten de manera consecutiva en varias imágenes de la secuencia, lo cual será aprovechado por el posterior módulo de seguimiento.

Las figuras 4.28 y 4.29 incluyen algunos ejemplos de los resultados de las rutas  $S_1$  y  $S_2$ , donde en cada detección se muestra la forma asignada al objeto, teniendo en algunos casos una doble detección para las señales detectadas en base a la orla exterior y a la región interna, y la clase asignada al pictograma en el extremo izquierdo de la imagen. En la figura 4.30 se muestran algunos ejemplos de la secuencia con lluvia  $S_4$  donde se puede comprobar que la señal no es identificada en todas las imágenes de la secuencia (en las figuras 4.30(b) y 4.30(c) la señal no es identificada debido a problemas en la etapa de segmentación, ya que la señal acromática se funde con el cielo de la escena) y en otras, se produce confusión entre clases (ver figura 4.30(e)). Finalmente, la figura 4.31 muestra ejemplos en condiciones de iluminación nocturna, donde podemos observar que los resultados son similares a los de horario diurno debido a que en entornos no-urbanos

Secuencia	# Imágenes	# Señales	# Detecciones	# Blobs ruidosos	# FA	# Confusiones
$S_1$	749	21	218	601	0	4
$S_2$	1174	21	237	985	3	4
$S_3$	860	20	227	728	4	4
$S_4$	995	25	85	622	8	2
$S_5$	798	17	127	434	7	7

Tabla 4.7: Resultados del sistema de detección y reconocimiento sobre un conjunto de secuencias.

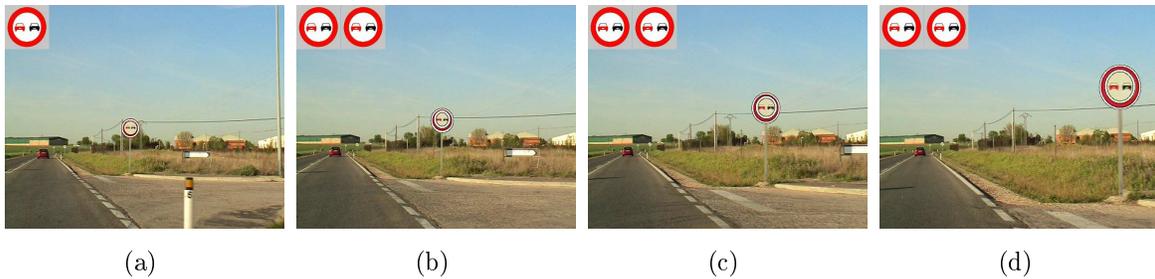


Figura 4.28: Imágenes de resultados de la secuencia  $S_1$ .

las señales se construyen con material retroreflectante que las hace visibles con la fuente de luz del vehículo.

Para testear el comportamiento del sistema ante las oclusiones, se manipuló un conjunto de imágenes con señales, provocando oclusiones de diferentes niveles y con diferentes ángulos de incidencia sobre el área de la señal. Para generar oclusiones se introdujo una máscara sintética circular con una variación aleatoria en la intensidad de los píxeles en los tres canales RGB. Con el objetivo de considerar el impacto del área de oclusión, se fijaron tres tamaños diferentes de máscara con diámetros iguales a la mitad, un tercio y un cuarto de la mayor dimensión del bounding-box (ver figura 4.32). Además, cada máscara se introdujo sobre el área de la señal en ocho orientaciones con incrementos de  $45^\circ$  y se

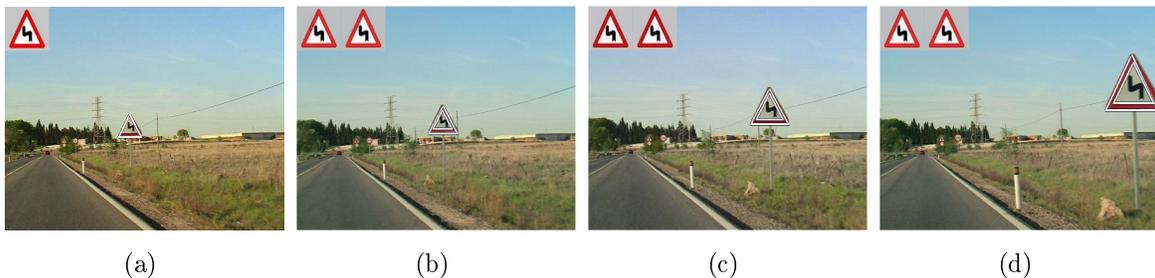


Figura 4.29: Imágenes de resultados de la secuencia  $S_2$ .

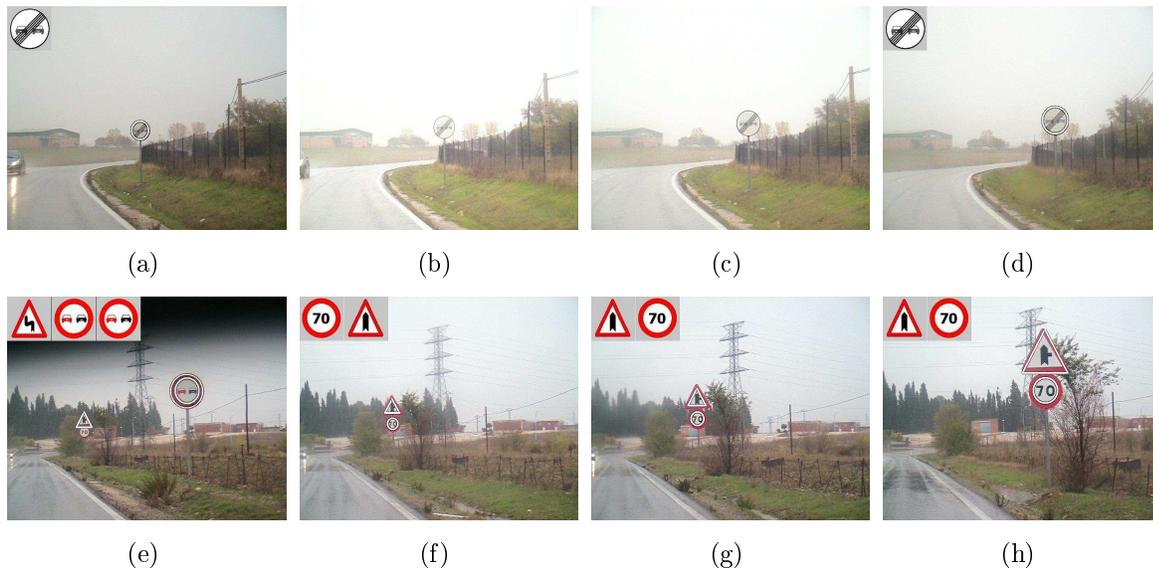


Figura 4.30: Imágenes de resultados de la secuencia  $S_4$  con pérdidas en (b,c) y confusión en (e).

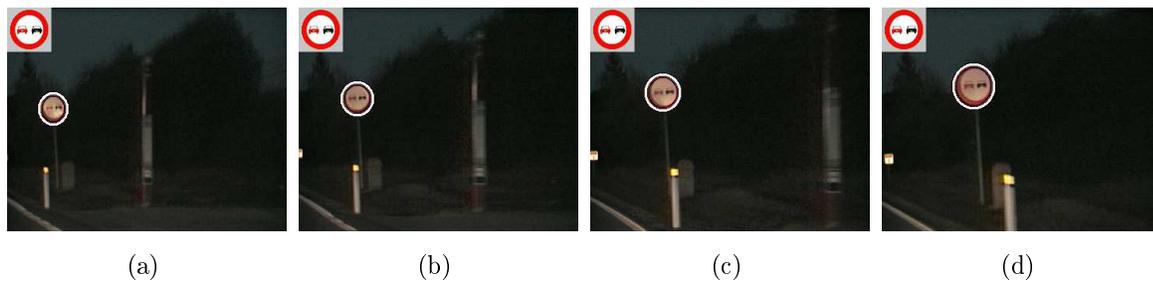


Figura 4.31: Imágenes de resultados de la secuencia nocturna  $S_5$ .

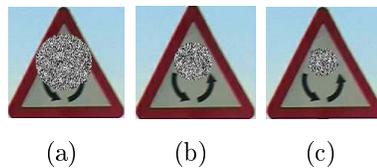


Figura 4.32: Tamaños de la máscara de oclusión con tres diámetros. De izquierda a derecha, diámetros de  $1/2$ ,  $1/3$  and  $1/4$ , respectivamente, de la mayor dimensión del bounding box.

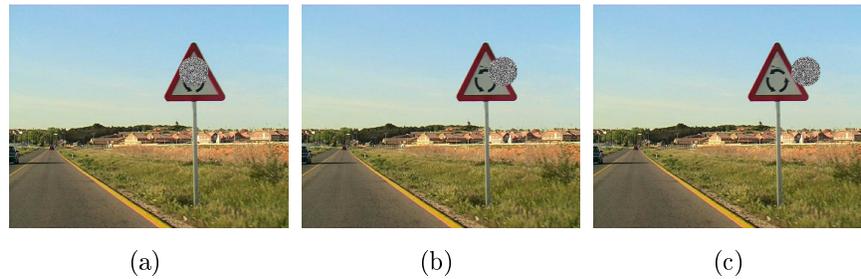


Figura 4.33: Desplazamientos de la máscara para la orientación de  $0^\circ$ .

desplazó en varias posiciones a lo largo de cada orientación, tal como se muestra en la figura 4.33 para un ángulo de  $0^\circ$ . La tasa de acierto en el reconocimiento fue de 93.24 %, 67.85 % y 44.90 % para las máscaras de tamaño pequeño, mediano y grande, respectivamente. Analizando los resultados se puede concluir que, obviamente, los peores resultados se obtienen cuando la máscara ocluye la parte central de la señal perdiendo la visibilidad del pictograma y por supuesto, a medida que aumenta el área de la máscara de oclusión.

## 4.6. Módulo de seguimiento

El seguimiento de las señales se ha realizado a dos niveles: seguimiento intra-imagen para unificar las múltiples detecciones de un objeto en una misma imagen y el seguimiento inter-imagen para realizar el seguimiento de una señal en varias imágenes de una secuencia.

### 4.6.1. Seguimiento intra-imagen

Debido a que la segmentación es un proceso clave en la detección y no hay un método que presente claramente mayor fiabilidad que el resto [Gomez10], la estructura modular del sistema RESET permite configurarlo para poder aplicar simultáneamente varios métodos de segmentación sobre una misma imagen. De esta manera, se consigue redundancia en la detección con el objetivo de minimizar la tasa de pérdida, ya que un mismo objeto puede ser detectado varias veces en una misma imagen. El propósito del seguimiento intra-imagen es agrupar las múltiples detecciones de un mismo objeto y generar una única alarma por cada señal presente en la imagen. Los criterios para establecer correspondencia entre los objetos detectados se enumeran a continuación:

- Color de segmentación. Si se emplean métodos de segmentación por color, los objetos a asociar deben presentar compatibilidad de acuerdo a los colores definidos para las señales. Las correspondencias de color permitidas incluyen las detecciones bajo el mismo canal de color y para diferentes canales, se establece la asociación rojo-blanco debido a la posibilidad de doble segmentación de las señales rojas circulares y rojas triangulares y la asociación rojo-amarillo para las señales en tramos de obras.

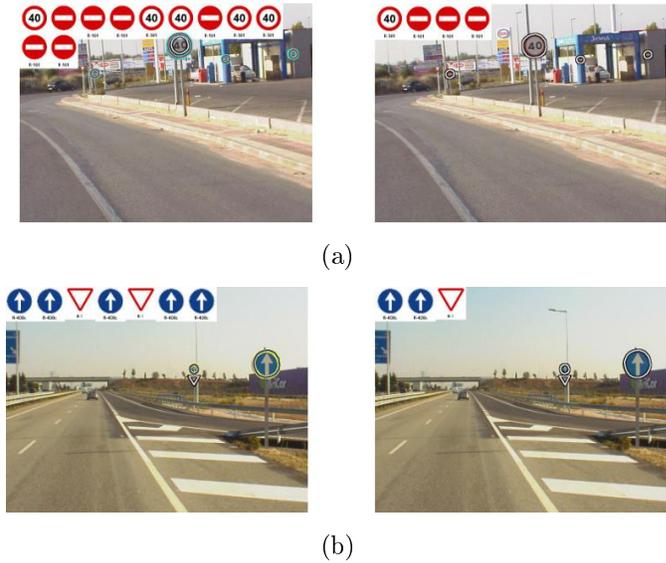


Figura 4.34: Resultados obtenidos mediante el tracking intra-imagen.

- Forma geométrica. Para asociar dos objetos, ambos deben ser detectados con la misma forma geométrica.
- Posición en el plano imagen. Para cada par de objetos detectados  $ROI_i$  y  $ROI_j$  se evalúa el porcentaje de solapamiento que se produce entre sus bounding-box. Para ello se mide la relación entre el área de intersección de sus bounding-box y el área total de la unión de los mismos, estableciéndose una correspondencia entre los objetos si la relación está por encima de un umbral  $\tau$ , de tal modo que la condición a comprobar es:

$$\frac{\text{área}(ROI_i \cap ROI_j)}{\text{área}(ROI_i \cup ROI_j)} > \tau, \quad (4.21)$$

donde el valor de  $\tau$  se ha fijado experimentalmente a 0.7.

En la figura 4.34 se muestra el efecto del tracking intra-imagen con dos ejemplos en los que las clases asignadas a cada detección múltiple coinciden y por tanto, no hay problema en la categorización de la señal. No obstante, se puede dar la circunstancia de que una señal sea identificada con diferentes clases en las diferentes detecciones y entonces, el criterio adoptado consiste en elegir la categoría con mayor valor en la suma de las salidas de las correspondientes SVM.

#### 4.6.2. Seguimiento inter-imagen

La detección de señales en una secuencia a partir de la información de cada imagen tiene dos problemas asociados: 1) es difícil basar la detección y la categorización de un objeto sobre una única imagen y 2) si la señal aparece con problemas de visibilidad, tal

como una oclusión temporal, es probable que ésta no sea detectada en algunas imágenes (ver figura 4.35). El objetivo del tracking inter-imagen es aprovechar la información temporal para realizar detecciones con mayor grado de fiabilidad. La información considerada en el tracking, que incluye el número de imágenes en que se detecta la señal, así como la evolución de su tamaño y su posición en la secuencia, permite reducir la cantidad de falsas alarmas.

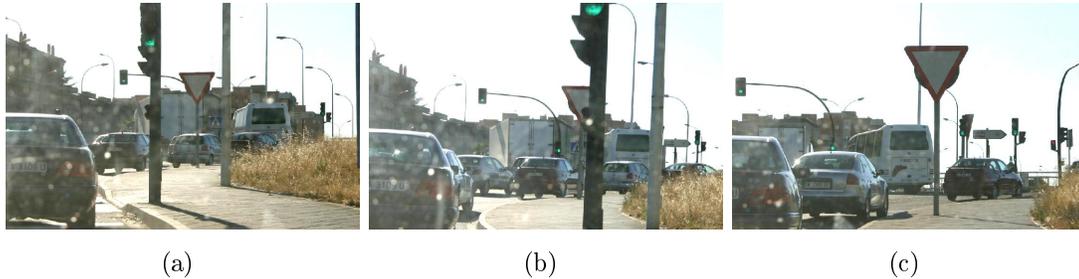


Figura 4.35: Ejemplo de oclusión temporal parcial en una video-secuencia.

El seguimiento de señales no es un problema trivial al existir movimiento relativo entre la cámara, los objetos estáticos (las señales) y el entorno. Además, el sistema debe tener capacidad para realizar el seguimiento de múltiples objetos simultáneamente, ya que una misma escena puede incluir varias señales. Desde el punto de vista de la cámara, el movimiento de aproximación del vehículo a la señal puede ser considerado como un sistema dinámico. Como modelo del sistema, en esta tesis se ha utilizado un filtro Alpha-Beta-Gamma ( $\alpha-\beta-\gamma$ ) [Kalata84] como caso particular del filtro de Kalman [Kalman60]. En el modelo se ha asumido que el movimiento de la cámara se realiza en la dirección del eje óptico y que el ángulo de la cámara respecto al firme de la calzada es nulo, donde la segunda premisa es cierta en la mayor parte de las configuraciones.

Para cada nueva señal detectada, el sistema realiza un proceso de inicialización en el que estima la distancia que el vehículo debe recorrer hasta sobrepasar la señal. Esta distancia bajo la que se rastrea la señal será referida como distancia de tracking  $D_t$ . Para determinar su valor se propone por simplicidad un modelo de cámara *pin-hole* y a partir de la semejanza de triángulos de la figura 4.36, se obtiene  $D_t$  mediante la siguiente ecuación:

$$D_t = \frac{A \cdot f}{d}, \quad (4.22)$$

donde  $A$  es el diámetro real de la señal,  $f$  es la distancia focal de la cámara y  $d$  es el diámetro de la señal en el plano imagen, el cual se calcula como el producto del tamaño de la señal en píxeles y el tamaño de la celda de la cámara CCD. De acuerdo a la normativa española [BOE00], existen tres tamaños normalizados de señales: 1200 mm, 900 mm y 600 mm. Teniendo en cuenta que el tamaño de celda de la cámara utilizada es  $4.65 \mu\text{m} \times 4.65 \mu\text{m}$ , la distancia focal es igual a 8.5 mm y que el filtro de tamaño de la etapa

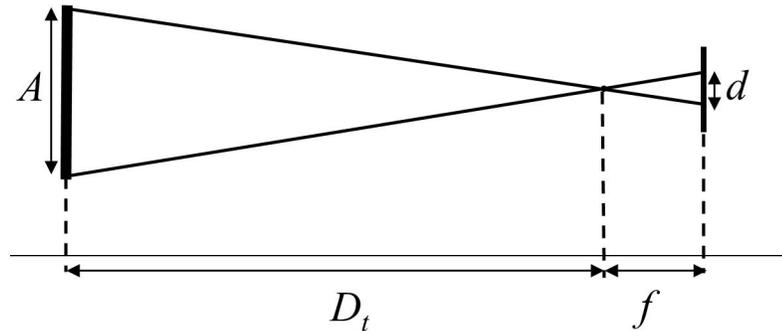


Figura 4.36: Modelo de cámara *pin-hole* para el cálculo de la máxima distancia de tracking.

600 mm	900 mm	1200 mm
35.37 m	53.06 m	70.75 m

Tabla 4.8: Distancia máxima de seguimiento para tamaños normalizados de señales viales en España.

de segmentación fija las dimensiones mínimas del blob a  $31 \times 31$  píxeles, se puede calcular la distancia máxima de seguimiento en cada caso, tal como se muestra en la tabla 4.8.

El algoritmo diseñado para el seguimiento inter-imagen maneja una lista con las señales detectadas y en cada nueva imagen se actualiza la posición de las mismas. Una vez que una señal ha desaparecido del campo de visión de la cámara, la señal abandona la lista generando un aviso o alarma. El funcionamiento del algoritmo de tracking sobre cada imagen a procesar se basa en los siguientes pasos:

1. Extracción de las señales en la imagen a partir de los procesos de segmentación, clasificación, reconocimiento y tracking intra-imagen.
2. Búsqueda de correspondencia de cada señal detectada con las señales de la lista de seguimiento.
3. Las señales de la lista de seguimiento que encuentran correspondencia con alguna de las señales detectadas en la imagen, actualizan la información con la nueva posición y dimensión en el plano imagen. Por el contrario, aquellas señales detectadas que no encuentran correspondencia con ninguna de la lista, inicializan en la lista una nueva señal.
4. Finalmente, se comprueba para cada una de las señales de la lista si se ha sobrepasado su distancia de tracking. En caso afirmativo, se extraen de la lista y se consideran verdaderas positivas aquellas señales que han sido detectadas en, al menos, 2 imágenes. La comprobación de si el sistema se encuentra actualmente dentro

de la distancia de seguimiento de una señal puede realizarse de dos maneras: 1) calculando la distancia a partir de las coordenadas GPS, y 2) a partir de la equivalencia en distancia de los pulsos generados por el odómetro.

La correspondencia entre una nueva detección y uno de los objetos de la lista de seguimiento en el tracking inter-imagen queda establecida siempre que se cumplan los siguientes criterios:

- Color de segmentación. Si se emplean métodos de segmentación por color, el objeto detectado en la imagen y el objeto de la lista deben presentar compatibilidad de acuerdo a los colores de las señales. Los criterios de color son los mismos que los del tracking intra-imagen.
- Forma geométrica. Las formas de los objetos entre los que se busca una posible correspondencia deben ser coincidentes.
- Dimensiones del bounding box. Si tenemos en cuenta el movimiento de aproximación de la cámara a la señal, las dimensiones del bounding-box de la detección en la imagen actual deben ser superiores a las de la señal en la lista de seguimiento. En el caso excepcional de que el vehículo se detenga, las dimensiones se mantendrán aproximadamente iguales.
- Posición en el plano imagen. La localización del centroide de un objeto detectado en la imagen actual viene dada por  $\mathbf{p} = [x, y]^T$  y la predicción de posición de cada objeto de la lista para la imagen actual es definida por  $\tilde{\mathbf{p}}_i = [\tilde{x}_i, \tilde{y}_i]^T$ . Para cada objeto  $i$ -ésimo de la lista de seguimiento se define una ventana de búsqueda en la imagen actual de tamaño  $W \times H$  y se establece una correspondencia entre este objeto y la nueva detección de la imagen si el centroide de esta última se encuentra dentro de la ventana, de tal manera que se cumpla:  $(\tilde{x}_i - W/2) < x < (\tilde{x}_i + W/2)$  y  $(\tilde{y}_i - H/2) < y < (\tilde{y}_i + H/2)$ . A la hora de estimar la nueva posición de los objetos de seguimiento en la imagen actual se utiliza un filtro de predicción, el cual actúa en dos etapas: 1) predicción, en la que el filtro estima la posición de los objetos detectados previamente sobre la imagen actual, y 2) corrección, en la que el filtro corrige la predicción en base a las posiciones actualizadas.

Aunque es una situación poco frecuente, puede darse el caso de que un objeto pueda establecer correspondencia simultáneamente con varios objetos de la lista de seguimiento al cumplir todos los criterios enumerados. En este caso, se ha elegido como criterio elegir aquel objeto más próximo a la posición de predicción.

Cabe destacar que en las diferentes detecciones de una señal a lo largo de la secuencia, el sistema de reconocimiento puede asignar diferentes clases debido a errores de confusión. No obstante, la alarma generada por el módulo de seguimiento es única para cada señal y también la clase asignada. La secuencia de las  $N$  imágenes en que una señal es detectada

se puede definir como  $S = [I_1, I_2, \dots, I_N]$  y el conjunto de las  $M$  clases asignadas a la señal sería  $\mathcal{C} = [\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_M]$ , donde  $M \leq N$  ( $M = N$  si la clase asignada es diferente en cada imagen detectada y  $M = 1$  si la clase asignada es la misma en todas las detecciones). A cada una de las clases asignadas  $\mathcal{C}_m$  se le otorga un peso  $W_{\mathcal{C}_m}$ , el cual es función de dos parámetros: 1) el número de veces que se ha asignado la clase  $\mathcal{C}_m$  a la señal en la secuencia, y 2) un factor que otorga mayor ponderación a las identificaciones de las últimas detecciones. El segundo aspecto tiene en cuenta el efecto del movimiento de aproximación del vehículo a la señal, a través del cual el tamaño del bounding-box aumenta progresivamente en el plano imagen. Por tanto, la definición de la señal es mayor en las últimas detecciones que en las primeras y la fiabilidad del proceso de reconocimiento aumenta. Así, el peso asociado a cada clase  $\mathcal{C}_m$  en una secuencia  $S$  viene dado por:

$$W_{\mathcal{C}_m} = \frac{ND_{\mathcal{C}_m}}{1 + \sum_{n=1}^N g_n(N - n)} \quad (4.23)$$

con

$$g_n = \begin{cases} 1, & \text{si } \mathcal{C}_m = \mathcal{C}_n \\ 0, & \text{si } \mathcal{C}_m \neq \mathcal{C}_n \end{cases} \quad (4.24)$$

donde  $ND_{\mathcal{C}_m}$  simboliza el número de veces que se asigna la clase  $\mathcal{C}_m$  a la señal,  $\mathcal{C}_n$  es la clase asignada en la detección de la imagen  $I_n$  y  $(N - n)$  representa la diferencia entre los índices de secuencia de la última imagen  $I_N$  en que la señal es detectada y la imagen  $n$ -ésima  $I_n$ . A partir de los  $M$  pesos, la clase asignada  $\mathcal{C}_{m^*}$  viene dada por el mayor peso:

$$m^* = \underset{m=1, \dots, M}{\operatorname{argmax}} [W_{\mathcal{C}_m}]. \quad (4.25)$$

### Algoritmo de tracking

El filtro  $\alpha - \beta - \gamma$ , basado en el filtro de Kalman, es usado en esta tesis doctoral para estimar los parámetros de movimiento en los estados de predicción y corrección mediante un vector de estado que incluye la posición, la velocidad y la aceleración. Suponiendo que la posición del centroide de un objeto para la imagen actual  $I_t$  es dada por  $\mathbf{p}(t) = [x(t), y(t)]^T$ , su vector de estado se define mediante  $\mathbf{s}(t) = [\mathbf{p}(t), \dot{\mathbf{p}}(t), \ddot{\mathbf{p}}(t)]^T$ , donde  $\dot{\mathbf{p}}(t)$  representa la velocidad y  $\ddot{\mathbf{p}}(t)$  la aceleración. Realizando una expansión en serie de Taylor, la ecuación del vector de predicción viene dada por:

$$\mathbf{s}(t^-) = \mathbf{A} \cdot \mathbf{s}(t - \Delta t) + \mathbf{w}(t) \quad (4.26)$$

o en forma matricial como:

$$\begin{bmatrix} \mathbf{p}(t^-) \\ \dot{\mathbf{p}}(t^-) \\ \ddot{\mathbf{p}}(t^-) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_2 & \Delta t \mathbf{I}_2 & \frac{\Delta t^2}{2} \mathbf{I}_2 \\ \mathbf{O}_2 & \mathbf{I}_2 & \Delta t \mathbf{I}_2 \\ \mathbf{O}_2 & \mathbf{O}_2 & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \mathbf{p}(t - \Delta t) \\ \dot{\mathbf{p}}(t - \Delta t) \\ \ddot{\mathbf{p}}(t - \Delta t) \end{bmatrix} + \mathbf{w}(t) \quad (4.27)$$

donde  $\mathbf{I}_2$  y  $\mathbf{O}_2$  representan, respectivamente, una matriz identidad y una matriz nula de orden  $2 \times 2$ . La evolución del vector de estado desde la imagen en el instante  $(t - \Delta t)$  a la imagen en el instante  $t^-$  viene dada por una matriz de orden  $3 \times 3$  conocida como matriz de transición  $\mathbf{A}$ . El vector  $\mathbf{w}(t)$  representa la incertidumbre de la predicción, asumiendo una variable aleatoria de media nula y una matriz de covarianza  $\mathbf{Q}(t) = E[\mathbf{w}(t)\mathbf{w}^T(t)]$ .

En el estado de predicción, la posición de la señal es estimada extrapolando el estado del filtro de Kalman de la imagen previa a la imagen actual. Entonces, se calcula la correspondencia entre la posición del objeto predicho y la posición detectada por el sistema de reconocimiento. La estimación de la posición de la señal, la velocidad y la aceleración del movimiento relativo se obtienen evaluando la diferencia de la trayectoria correspondiente:

$$\mathbf{z}(t) = \mathbf{H}\mathbf{s}(t) + \mathbf{v}(t), \quad (4.28)$$

donde  $\mathbf{z}(t)$  representa la observación externa (es decir, la posición detectada), y  $\mathbf{v}(t)$  la incertidumbre de tal observación. Aquí se asume que  $\mathbf{v}(t)$  es de media nula con una matriz de covarianza  $\mathbf{R}(t) = E[\mathbf{v}(t)\mathbf{v}^T(t)]$ . En el estado de corrección, la medida de la posición del objeto es utilizada para adaptar el estado del modelo del filtro de Kalman a partir de:

$$\mathbf{s}(t) = \mathbf{s}(t^-) + \mathbf{K}(t)(\mathbf{z}(t) - \mathbf{H}\mathbf{s}(t^-)). \quad (4.29)$$

El factor de ponderación  $\mathbf{K}(t)$  viene dado a partir de las siguientes ecuaciones:

$$\mathbf{K}(t) = \mathbf{E}(t^-)\mathbf{H}^T(t)[\mathbf{H}(t)\mathbf{E}(t^-)\mathbf{H}^T(t) + \mathbf{R}(t)]^{-1} \quad (4.30)$$

$$\mathbf{E}(t^-) = \mathbf{A}\mathbf{E}(t - \Delta t)\mathbf{A}^T + \mathbf{Q}(t) \quad (4.31)$$

$$\mathbf{E}(t^+) = [\mathbf{I} - \mathbf{K}(t)\mathbf{H}(t)]\mathbf{E}(t^-). \quad (4.32)$$

Para el filtro  $\alpha - \beta - \gamma$ , la matriz de pesos viene dada mediante:

$$K = \begin{bmatrix} \alpha \\ \beta/T \\ \gamma/T^2 \end{bmatrix}, \quad (4.33)$$

donde la relación óptima entre los parámetros  $\alpha$ ,  $\beta$  y  $\gamma$  que minimiza el error cuadrático medio de las estimaciones de posición, velocidad y aceleración [Kalata84] es dada como:

$$\beta = 2(2 - \alpha) - 4\sqrt{1 - \alpha} \quad (4.34)$$

y

$$\gamma = \frac{\beta^2}{\alpha}, \quad (4.35)$$

donde el parámetro  $\alpha$ , que ha sido ajustado experimentalmente, toma valores comprendidos entre 0 y 1.

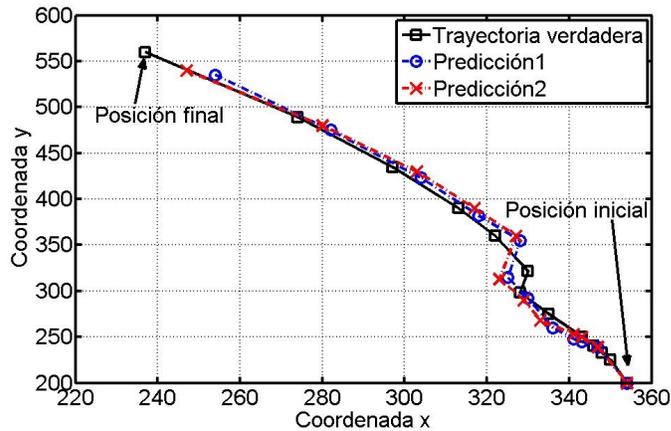


Figura 4.37: Predicción de posición con un modelo de aceleración constante (predicción 1) y con un modelo de velocidad constante (predicción 2).

La figura 4.37 muestra la predicción de la posición de una señal utilizando dos modelos basados, respectivamente, en suposiciones de velocidad y aceleración constantes. Experimentalmente, los valores óptimos para  $\alpha$ ,  $\beta$  and  $\gamma$  son 0.8, 0.61 y 0.47, respectivamente.

Como ejemplo de la información de salida del módulo de tracking, en la tabla 4.9 se muestra la información proporcionada por el sistema en los primeros dos kilómetros de la secuencia  $S_4$  de la sección 4.5. La información incluye para cada señal el número de veces que ha sido detectada, su posición GPS en la última imagen detectada y la distancia real en metros bajo la que ha sido rastreada en la secuencia. Todas las señales han sido detectadas y rastreadas correctamente en este tramo y además, el módulo de seguimiento es capaz de rechazar tres falsas alarmas de detecciones individuales que no presentan coherencia temporal, tal como se puede observar en la tabla 4.10. Es importante considerar que el número de veces que una señal aparece en la secuencia no es constante y depende de la velocidad de marcha del vehículo. En los resultados de la secuencia de test, existe una oclusión debido al efecto del limpiaparabrisas del automóvil, tal como se muestra en la figura 4.38, y se puede observar que el seguimiento de la señal se realiza correctamente sin que se produzca desenganche del filtro. Para mostrar los resultados del tracking secuencial se ha representado en cada imagen el bounding-box de los objetos detectados tras aplicar el seguimiento intra-imagen y tan sólo en la última detección de la secuencia, el sistema muestra el resultado de la identificación del seguimiento inter-imagen.

Para probar la fiabilidad del sistema completo se ha utilizado una secuencia de 7799 imágenes grabadas en un entorno mixto urbano y de carretera que cubre un tramo de, aproximadamente, 12 kilómetros. Esta secuencia no tiene relación con las imágenes utilizadas en la base de datos. En la tabla 4.11 se muestran los resultados obtenidos con el tracking utilizando tres algoritmos diferentes en la etapa de segmentación (umbralización del RGB normalizado (RGBNT), umbralización del HSI (HST) y umbralización

Clase asignada	# Detecciones	Latitud	Longitud	Distancia de tracking
	14	40.5175N	3.3489W	40.6213
	11	40.5172N	3.3491W	39.1705
	9	40.5168N	3.3495W	43.0108
	5	40.5165N	3.3497W	22.8495
	14	40.5157N	3.3504W	37.8198
	5	40.6652N	3.3509W	31.6728

Tabla 4.9: Señales detectadas a la salida del módulo de tracking para una secuencia de ejemplo.

Clase asignada	# Detecciones	Latitud	Longitud
	1	40.5163N	3.3498W
	1	40.5159N	3.3502W
	1	40.5154N	3.3507W

Tabla 4.10: Falsas alarmas descartadas por el módulo de tracking para una secuencia de ejemplo.

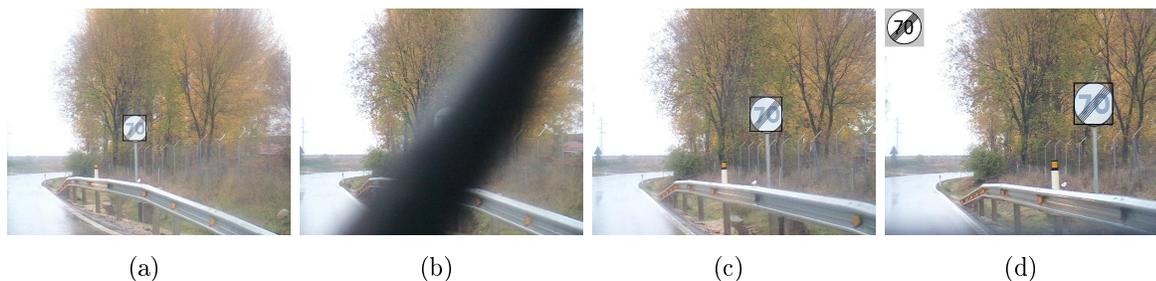


Figura 4.38: Ejemplo de tracking con oclusión temporal en una secuencia.

Resultados	RGBN	HS	OST
Total	46	45	47
Correctas	46	44	47
Falsas	0	1	0
Perdidas	1	3	0

Tabla 4.11: Resultados del sistema completo usando la información de tracking.

del espacio Ohta (OST)), mientras que para la clasificación de forma se ha empleado la firma y para el reconocimiento se ha empleado el descriptor basado en píxeles de interés con una arquitectura OVA-SVM. Además, el método utilizado para la comprobación de la distancia de seguimiento es el basado en las coordenadas GPS. La primera fila de la tabla 4.11 muestra el número de señales totales detectadas que han entrado al módulo de tracking, la segunda fila indica las señales que fueron correctamente identificadas, la tercera el número de falsas alarmas y la cuarta el número de señales perdidas respecto al método que proporciona el mayor número de señales correctas.

Después del análisis del tracking en varias secuencias, se ha observado el ‘desenganche’ del filtro de predicción en los tramos de curvas cerradas. La posición de las señales en el plano imagen está estrechamente vinculada a la trayectoria del sistema de captura instalado en el vehículo y por tanto, en los tramos de máxima curvatura la apariencia de las señales varía bruscamente en posición y tamaño entre imágenes consecutivas (ver figura 4.39).

## 4.7. Análisis de la carga computacional

La calidad de un sistema TSDRS se mide en base a dos criterios principales: 1) la fiabilidad, que viene dada por un alto valor de la tasa de acierto TPR y una baja tasa de falsa alarma FPR y 2) la velocidad, buscando optimizar el tiempo de procesado. En el campo del procesado de imagen, la carga computacional se puede definir como el número de operaciones aritméticas y de gestión de datos por unidad de tiempo que son necesarias



Figura 4.39: Ejemplo de ‘desenganche’ del tracking debido a la curvatura del recorrido.

para procesar cada imagen. Los factores más relevantes que influyen en el tiempo de cómputo son:

1. Propiedades de la imagen. Las propiedades de las imágenes son fácilmente configurables mediante el sistema de adquisición. La carga computacional en el módulo de segmentación depende del tamaño de la imagen, especialmente cuando se emplean algoritmos que trabajan píxel a píxel. Existe un compromiso entre velocidad y fiabilidad en cuanto que el tamaño de la imagen se puede reducir hasta un cierto límite, ya que para tamaños muy pequeños de la imagen los objetos de interés son difícilmente detectables e identificables. Además, en el caso de un TSDRS que realice tareas de tracking es de vital importancia que el sistema detecte las señales tan pronto como aparezcan en la escena. Otro criterio a considerar es si el sistema trabaja con niveles de gris o con imágenes en color. El procesado con niveles de gris demanda menor carga computacional pero, en contrapartida, descarta la información de color.
2. Número de algoritmos de segmentación. Aunque existen numerosos trabajos sobre segmentación de señales, no existe hasta la actualidad ningún método que sea robusto a todas las dificultades mencionadas. Por esta razón, RESET permite trabajar con varios algoritmos de segmentación en paralelo para conseguir redundancia de detección, a pesar de aumentar la complejidad computacional.
3. Complejidad del módulo de reconocimiento. La carga computacional de un clasificador basado en SVM depende en gran medida del número de vectores soporte. Según crece el número de clases y el número de muestras de entrenamiento, el número de vectores soporte crece exponencialmente.

En el intento de optimizar el sistema e identificar los posibles ‘cuellos de botella’ es conveniente analizar el perfil de carga. En la tabla 4.12 se muestra el número de ciclos de CPU demandados por los módulos de segmentación, clasificación y reconocimiento sobre un promedio de 100 imágenes con 2 algoritmos de segmentación para cada imagen. En esta información, se ha omitido la carga de otras tareas adicionales que debe realizar el sistema, tales como las operaciones de lectura y escritura de la imagen, que son necesarias.

Proceso	Ciclos de CPU	Porcentaje de carga
Segmentación	1068	1.52 %
Clasificación	3118	4.44 %
Reconocimiento	49363	70.38 %
Tracking	16579	23.64 %

Tabla 4.12: Carga computacional de los módulos del sistema.

De los resultados de la tabla, se puede concluir que el proceso de reconocimiento requiere una carga 46, 15 y 3 veces mayor que la correspondiente a los procesos de segmentación, clasificación y seguimiento, respectivamente. Este hecho viene dado por dos factores: 1) el número de vectores soporte de las SVM implementadas es elevado al trabajar con un alto número de clases y debido a las variaciones de iluminación existentes entre las muestras de entrenamiento, y 2) el número de ROIs que entran al módulo de reconocimiento es alto debido a que, a pesar de que las falsas alarmas pueden filtrarse en las etapas de segmentación y clasificación de forma, muchas de ellas no son descartadas hasta alcanzar la etapa de reconocimiento.

En la figura 4.40 se representa la salida del módulo de clasificación para una misma imagen, utilizando dos métodos de segmentación diferentes. Cabe destacar la capacidad del módulo de reconocimiento para rechazar el gran número de falsas detecciones, tal como puede verse en la figura 4.40(c).

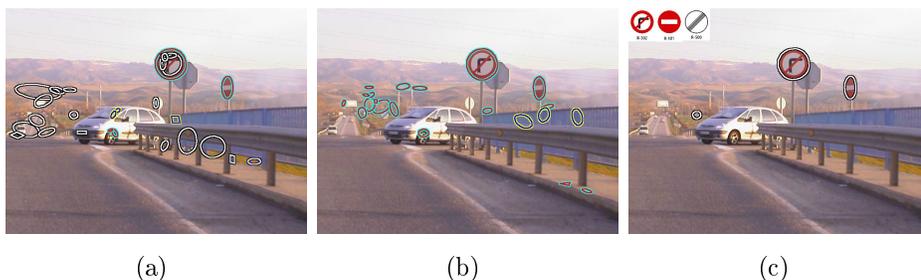


Figura 4.40: (a) y (b) Detecciones realizadas con diferentes métodos de segmentación. (c) Resultado a la salida del módulo de reconocimiento.

#### 4.7.1. Resumen

En este capítulo se ha descrito la implementación de un sistema completo para detección y reconocimiento de señales (RESET) basado en una estructura modular: segmentación, clasificación de forma, reconocimiento del pictograma y seguimiento. A continuación se describen las características más relevantes del sistema RESET mediante ejemplos de video-secuencias grabadas con diferentes cámaras, tanto en entornos urbanos como interurbanos. Los resultados que se muestran son los obtenidos a la salida de la etapa de

reconocimiento, previamente al proceso de seguimiento, y en cada imagen se representan las plantillas de las señales identificadas. Cabe recordar que para minimizar las pérdidas (verdaderos negativos), se trabaja con multidetección aplicando en paralelo varios algoritmos de segmentación sobre la misma imagen. De un modo experimental, las características más destacables del sistema RESET son las siguientes:

- Las técnicas homográficas utilizadas para normalizar la orientación de cada ROI permiten que el sistema sea robusto a las deformaciones proyectivas. En las figuras 4.41(a) y 4.41(b) se muestran dos imágenes de ejemplo.
- La robustez del sistema se ha testado con secuencias grabadas bajo diferentes condiciones ambientales. En las mismas, se han considerado cambios de iluminación que incluyen: a) día soleado, b) día lluvioso, c) luz de bajo contraste, y d) noche. En las figuras 4.41(c) y 4.41(d) se muestran los resultados para luz diurna y nocturna.
- Los resultados muestran que el sistema RESET posee cierto grado de tolerancia frente a las oclusiones parciales. En las figuras 4.41(e) y 4.41(f) se recogen dos imágenes de una secuencia en las que la señal de *Ceda el paso* es correctamente detectada, a pesar de que el árbol ocluye parcialmente su visibilidad.
- Las señales verticales se colocan, en algunas ocasiones, agrupadas en forma de arrays sobre un mismo poste. Esta circunstancia se da con mayor frecuencia en entornos urbanos y podría provocar que diferentes señales se segmentasen como un único objeto. No obstante, los algoritmos de detección implementados para la clasificación de formas del RESET permiten aislar de forma independiente los diferentes elementos del array, tal como se muestra en las figuras 4.41(g) y 4.41(h).
- Al contrario que en la gran mayoría de trabajos publicados en la literatura, el sistema RESET permite trabajar con una base de datos completa y discriminar con gran fiabilidad las aproximadamente 300 señales viales definidas en España. Aunque el sistema ha sido entrenado y testado para trabajar con la señalización española, éste puede ser fácilmente configurable para ser utilizado con la señalización de otros países. Para ello, se requiere un proceso previo de entrenamiento con el nuevo conjunto de clases y en algunos casos, es necesario añadir nuevos canales cromáticos para la segmentación de otros colores a los empleados en la señalización española. En las figuras 4.41(i) y 4.41(i) se muestran dos ejemplos de los resultados obtenidos al trabajar sobre una base de datos de EE.UU.

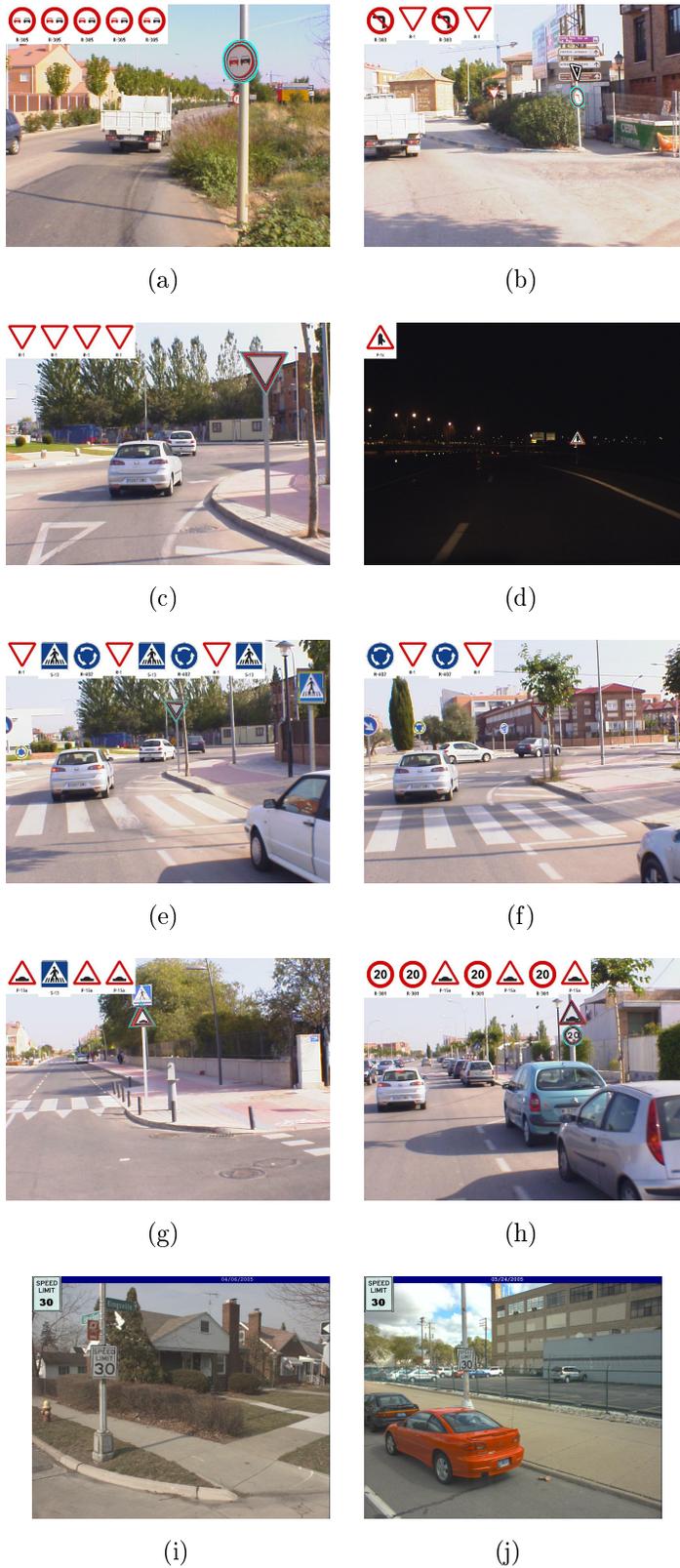


Figura 4.41: Características del sistema RESET. De arriba a abajo, deformaciones proyectivas, cambios de iluminación, oclusiones parciales, arrays y señales viales estadounidenses.

# Capítulo 5

## Arquitecturas de reconocimiento basadas en SVM

El objetivo de este capítulo es ahondar en el estudio comparativo de diferentes técnicas para el reconocimiento de señalización vertical, llegando a un compromiso que conjugue fiabilidad y eficacia. El diagrama de bloques propuesto para este módulo se estructura en tres etapas, tal como puede verse en la figura 5.1, correspondientes al preprocesado, a la extracción del descriptor y a la etapa de clasificación, la cual se basa en las SVM como herramienta debido a su buena capacidad de generalización. En este capítulo se investigará el comportamiento de diferentes técnicas a la hora de abordar cada una de las tres etapas con el objetivo de optimizar el sistema de reconocimiento.

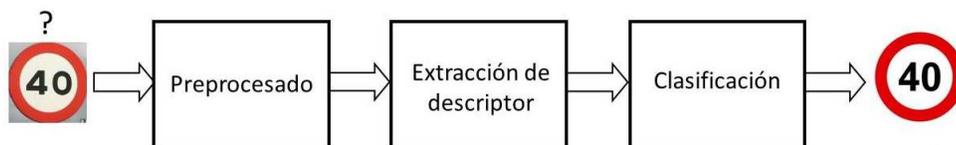


Figura 5.1: Diagrama de bloques del sistema de reconocimiento.

### 5.1. Técnicas de preprocesado

Uno de los mayores problemas en el módulo de reconocimiento es la influencia que tiene la iluminación del entorno en el que se encuentra localizada la señal. El coste de clasificación con SVM es directamente proporcional al número de vectores soporte y éste se encuentra relacionado con la disparidad de las muestras. Esto puede comprobarse en la figura 4.24, donde se observa que muchos de los vectores soporte positivos son similares, salvo en sus niveles de iluminación y esta circunstancia puede dificultar la tarea de reconocimiento. Por estas razones, en esta tesis doctoral se ha propuesto el uso de cuatro

técnicas de preprocesado con el objetivo de intentar homogeneizar el nivel de iluminación de las muestras:

1. Normalización del nivel de gris. Consiste en la normalización del nivel medio de luminancia, de tal modo que los histogramas queden centrados en torno a un nivel de gris de valor constante,  $K$ . En la figura 5.2(c) se muestran los resultados de la normalización sobre un conjunto de muestras de ejemplo ordenadas por nivel creciente del promedio de luminancia (ver figura 5.2(b)), después de aplicarse una normalización con  $K = 120$ .
2. *Contrast-stretching*. Se basa en utilizar funciones lineales por tramos a través de una transformación. Las causas de las imágenes con poco contraste pueden ser atribuibles a una pobre iluminación de la escena, a la pérdida de rango dinámico en el sensor de la cámara, a un mal ajuste de la configuración durante la grabación de la video-secuencia o al mal estado de conservación de la señal. La figura 5.3 muestra el perfil de transformación contrast-stretching, donde los puntos  $(r_1, s_1)$  y  $(r_2, s_2)$  controlan la forma de la función. Si  $r_1 = s_1$  y  $r_2 = s_2$ , la transformación es una función lineal que no produce ningún cambio en los niveles de gris. En esta tesis se ha fijado  $(r_1, s_1) = (r_{min}, 0)$  y  $(r_2, s_2) = (r_{max}, L - 1)$ , donde aquí  $L$  representa el número de niveles de gris o equivalentemente la profundidad de la imagen en escala de gris, que para un mapa de bits es igual a 256. Los valores de los parámetros  $r_{min}$  y  $r_{max}$  son fijados a través de dos umbrales,  $T_{cs1}$  y  $T_{cs2}$ , que especifican el porcentaje de píxeles que debe presentar un nivel de gris menor que  $r_{min}$  y  $r_{max}$ , respectivamente. La Figura 5.2(d) muestra el resultado de aplicar esta transformación sobre el conjunto de la figura 5.2(b) con  $T_{cs1} = 0.02$  y  $T_{cs2} = 0.98$ .
3. Ecualización del histograma. Esta transformación pretende obtener para cada imagen un histograma con una distribución uniforme sobre un conjunto discreto de niveles de gris,  $N_{eq}$ . El histograma se distribuye en todo el rango disponible separando en lo posible las ocupaciones de cada nivel. El resultado de la ecualización maximiza el contraste de una imagen sin perder información de tipo estructural, esto es, conservando su entropía. En la figura 5.2(e) se muestra el resultado de la ecualización sobre el conjunto original de la figura 5.2(b).
4. Umbralización. Debido a que la mayor parte de las señales presentan un mensaje que destaca del fondo, el proceso de umbralización trata de extraer este contenido. Al aplicar un umbral sobre la muestra en escala de gris, teóricamente se separan los píxeles correspondientes al mensaje de aquellos que pertenecen al fondo. La mayoría de las técnicas de umbralización se basan en estadísticas sobre el histograma unidimensional de la imagen y/o su matriz de co-ocurrencia. Para localizar los umbrales se pueden usar procedimientos paramétricos y no paramétricos. En particular, en este trabajo se ha optado por el método de Otsu (ver [Otsu79]), el cual elige el umbral óptimo maximizando la varianza entre las regiones a diferenciar dentro de una

imagen. Cuando se trabaja con imágenes del mundo real, la mayoría de los métodos de umbralización no proporcionan muy buenos resultados debido a la presencia de ruido, histogramas planos o una iluminación inadecuada. En la figura 5.2(f) se muestra el resultado de la umbralización por el método de Otsu sobre el conjunto de la figura 5.2(b), donde se puede ver que en algunos casos la umbralización provoca la pérdida absoluta del pictograma de las señales y da idea de que no es una buena elección de preprocesado.

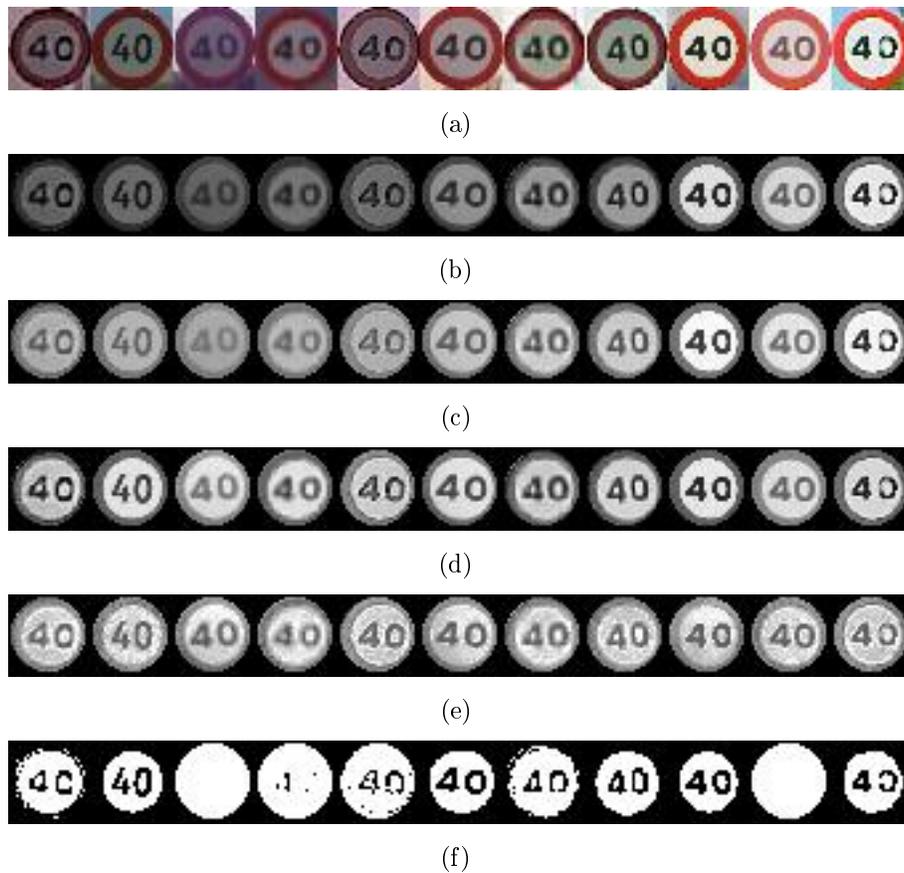


Figura 5.2: Ejemplos de pre-procesado sobre un conjunto de muestras. (a) Conjunto original de muestras en color. (b) Conjunto original en escala de gris tras aplicar la máscara de eliminación de fondo. (c) Muestras tras aplicar normalización del nivel de gris con  $K = 120$ . (d) Muestras tras aplicar la técnica de contrast-stretching con  $T_{cs1}=0.02$  y  $T_{cs2}=0.98$ . (e) Muestras con ecualización de histograma empleando  $N_{cq} = 256$ . (f) Muestras tras aplicar la umbralización de Otsu.

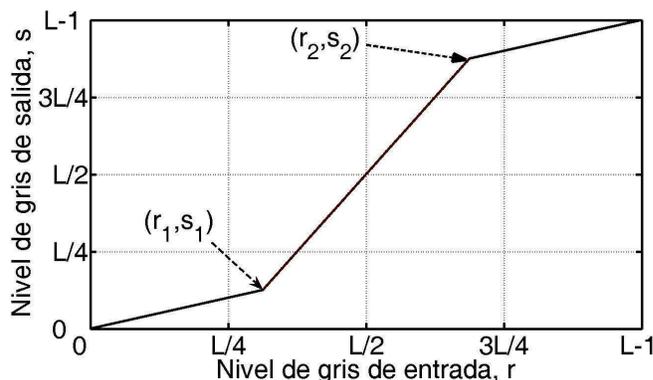


Figura 5.3: Perfil de transformación Contrast-Stretching.

## 5.2. Extracción del descriptor

La elección óptima de las características más apropiadas para el reconocimiento de señalización vertical es un problema abierto, del cual depende la calidad del sistema, y presenta gran importancia en sistemas TSDRS al considerar la variabilidad propia de los entornos exteriores. El objetivo del descriptor es codificar las características que mejor representan numéricamente la imagen con la finalidad de alimentar la posterior etapa de clasificación. La calidad del descriptor se mide por su nivel de discriminabilidad a la hora de diferenciar las muestras entre clases y por su dimensión, de la cual dependerá también la carga computacional del sistema.

Múltiples descriptores han sido empleados en la literatura intentando capturar diferentes propiedades de la imagen, tales como las intensidades de los píxeles, el color, la textura y los bordes. Los diferentes tipos de descriptores se pueden incluir en una de las siguientes aproximaciones:

- Descriptores basados en el dominio espacial. Las características del descriptor proceden del muestreo de los píxeles en la representación directa del objeto en el plano imagen.
- Descriptores basados en distribuciones. Estas técnicas utilizan histogramas o representan diferentes características de apariencia. Un descriptor muy simple es el formado por la distribución de las intensidades de los píxeles representada mediante un histograma.
- Técnicas espacio-frecuenciales. Algunas técnicas describen el contenido frecuencial de una imagen. La transformada de Fourier descompone el contenido de la imagen en funciones base, pero en esta representación las relaciones espaciales entre puntos no son consideradas, lo que las hace difíciles de adaptar a una aproximación local. Los filtros de Gabor son un grupo de wavelets que resuelven estos problemas, donde cada wavelet captura energía a una frecuencia específica y una dirección específica.

pero un número elevado de filtros de Gabor son necesarios para capturar pequeños cambios en frecuencia y orientación. Los filtros de Gabor y las wavelets son empleados en el contexto de clasificación de imágenes a partir de texturas.

- Técnicas basadas en derivadas de la imagen. Las propiedades de las derivadas locales ('local jet') fueron investigadas por [Koenderik87]. Freeman y Adelson [Freeman91] implementaron filtros direccionables ('steerable filters') que sintetizan un filtro en una orientación particular a partir de la combinación lineal de un conjunto de funciones base. Una estimación estable de las derivadas es obtenida mediante la convolución con las derivadas gaussianas.

En cualquier caso, no existe un trabajo que analice qué descriptores son los más apropiados para el reconocimiento de señales y en esta tesis doctoral se evalúan cinco descriptores a partir de imágenes en escala de gris:

1. Histograma de gradientes orientados (HOG, *Histograms of Oriented Gradient*). El descriptor HOG fue introducido por [Dalal05] para la detección de peatones. Las características HOG poseen una gran invarianza a los cambios de iluminación y están basados en la idea de que la apariencia local de un objeto puede ser descrita mediante la distribución de las direcciones de sus bordes, aún sin información espacial. El cálculo de las características HOG conlleva los siguientes pasos:
  - a) Cálculo del gradiente. Consiste en filtrar la imagen con los kernels  $[-1,0,1]$  y  $[-1,0,1]^T$  y calcular a partir de la imagen filtrada, la magnitud y orientación del gradiente en cada píxel de la imagen.
  - b) Discretización de la orientación. Consiste en dividir la imagen en celdas no solapadas de igual dimensión y dentro de cada una de ellas, se calcula un histograma de orientaciones. El voto de cada píxel es ponderado por la magnitud de su gradiente. Las celdas pueden presentar simetría rectangular o radial y las orientaciones del histograma pueden extenderse de 0 a  $\pi$  rad o de 0 a  $2\pi$  rad, dependiendo si se considera el gradiente sin o con signo.
  - c) Agrupamiento de bloques. Las celdas son agrupadas en bloques solapados con el objetivo de normalizar el gradiente de manera local.
  - d) Normalización del bloque. Para conseguir una mejor invarianza frente a los cambios de iluminación, se realiza una normalización en cada bloque acumulando una medida de la energía local. Así, dentro de cada bloque se aplica una normalización sobre las celdas.

El vector final se forma concatenando los histogramas de cada celda después de realizar los cuatro pasos mencionados. En la figura 5.4 se muestra el gradiente de una muestra completa y el histograma de una celda en el intervalo  $[0,2\pi]$  rad, donde se pueden observar las dos orientaciones dominantes correspondientes a los bordes

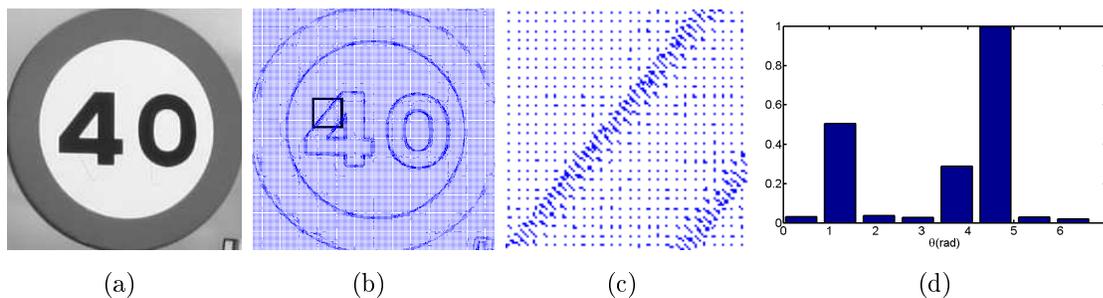


Figura 5.4: Ejemplo de cálculo del gradiente. (a) Muestra original en escala de gris. (b) Mapa de orientación del gradiente de la imagen original. (c) Mapa de orientación del gradiente en la celda seleccionada. (d) Histograma de orientaciones.

del dígito ‘4’. Considerando que las muestras se corresponden con blobs normalizados en tamaño con las mismas dimensiones vertical y horizontal, en esta tesis se proponen celdas cuadradas y por tanto, el descriptor HOG queda definido mediante dos parámetros de configuración: el número de celdas o ventanas en cada dimensión vertical y horizontal, denotado por  $N_r$  y  $N_c$ , respectivamente, que en esta tesis coinciden entre sí ( $N_r=N_c=N_w$ ) y el número de intervalos en que se discretizan las orientaciones del gradiente en cada celda,  $N_b$ .

2. *SPIN*. Lazebnik [Lazebnik03] propuso un descriptor invariante llamado ‘imagen spin’ en el contexto de clasificación de texturas. El descriptor spin está basado en regiones concéntricas y es completamente invariante a los cambios de rotación. El dominio de la imagen spin es un histograma bidimensional que codifica la distribución de la intensidad y la distancia al punto central (ver figura 5.5). La proyección de la imagen spin correspondiente a una distancia fija  $d$  es el histograma de los valores de intensidad de los píxeles localizados a distancia  $d$  del centro. En esta tesis se ha implementado el descriptor como un histograma suave en el sentido de que cada pixel contribuye a más de un intervalo. En concreto, la contribución de un pixel localizado en  $r$  al intervalo indexado por  $(d, i)$  viene dado por:

$$\exp\left(\frac{(|r - r_0| - d)^2}{2\alpha^2} - \frac{|I(x) - i|^2}{2\beta^2}\right), \quad (5.1)$$

donde  $r_0$  es la localización del centro del pixel y  $\alpha$  y  $\beta$  son los parámetros que modelan la anchura del suavizado del histograma. Fijados  $\alpha=0.6$  y  $\beta=0.4$ , en el descriptor SPIN manejamos dos parámetros de ajuste: el número de intervalos de distancia al punto central de la muestra,  $N_d$ , y el número de intervalos de intensidad,  $N_f$ .

3. *RIFT* (*Rotation-Invariant Feature Transform*). La muestra es dividida en anillos concéntricos de igual anchura y el histograma de gradiente orientado es compu-

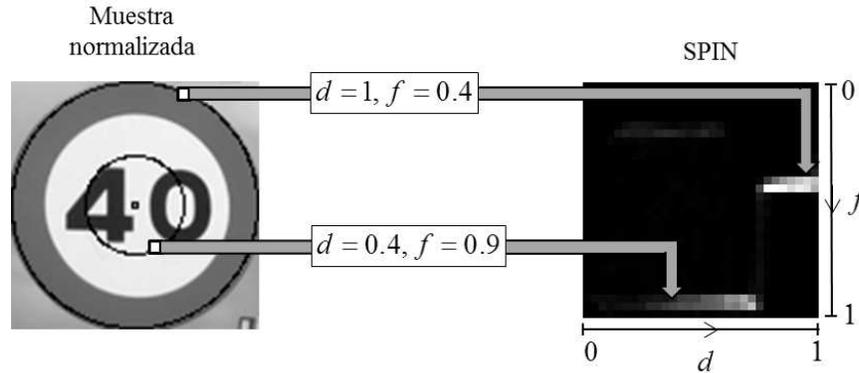


Figura 5.5: Construcción del descriptor SPIN. Dos puntos son mapeados de la ROI normalizada (izquierda) a las localizaciones del mapa del descriptor (derecho).

tado dentro de cada anillo (ver figura 5.6). Para mantener la invarianza frente a la rotación, la orientación es medida en cada punto en relación a la dirección del punto central. En este descriptor existen dos parámetros de configuración: el número de intervalos de distancia normalizados,  $N_d$ , y el número de intervalos en que se discretizan las orientaciones del gradiente en cada anillo,  $N_b$ .

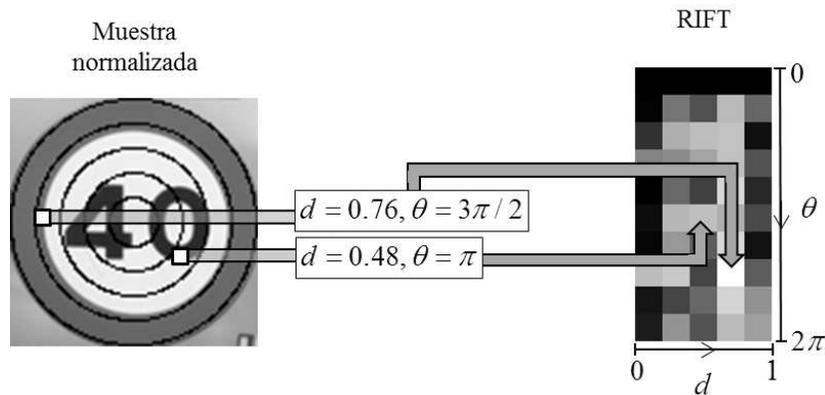


Figura 5.6: Construcción del descriptor RIFT. Dos puntos son mapeados de la ROI normalizada (izquierda) a las localizaciones del mapa del descriptor (derecha).

4. *LBP (Local Binary Pattern)*. El descriptor LBP [Ojala02] se inspira en la idea de que los objetos pueden ser analizados como una composición de micropatrones. Esta técnica, ideada originalmente para la descripción de texturas, presenta alta invarianza frente a cambios de intensidad de la imagen y gran eficiencia computacional. El operador LBP asigna una etiqueta a cada píxel de la imagen, umbralizando las intensidades del vecindario respecto al valor del píxel central y considerando el resultado como un número binario. De esta manera, el histograma de etiquetas puede ser usado como un descriptor de textura, tal como puede verse en la figura 5.7.

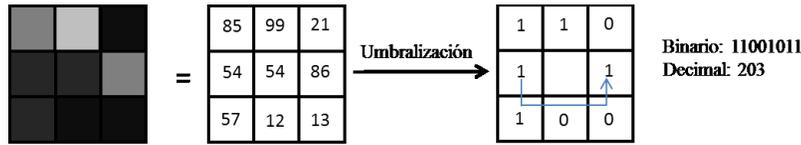


Figura 5.7: Operador LBP.

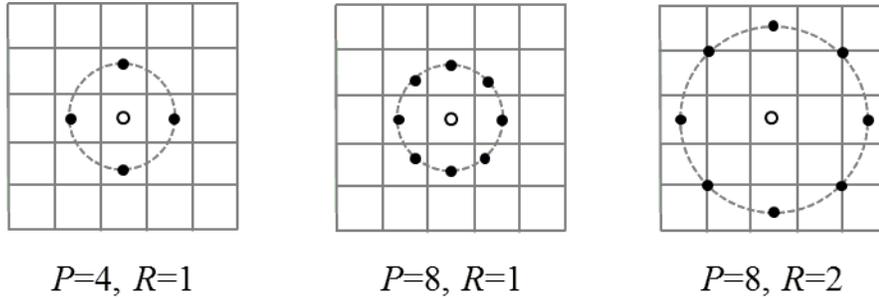


Figura 5.8: LBP con diferentes escalas variando el radio y los puntos de muestreo.

Para tratar con texturas a diferentes escalas, el operador puede ser extendido a diferentes tamaños ampliando el vecindario local para cualquier radio y número de puntos de muestreo. La notación utilizada es  $LBP_{P,R}$ , donde  $R$  es el radio del círculo a muestrear y  $P$  es el número de puntos muestreados. En la figura 5.2 se muestran tres ejemplos para diferentes parejas de valores de ambos parámetros. Denotando la característica del anillo centrado alrededor del píxel  $(x, y)$  como  $B(x, y) = \langle b_{P-1}, \dots, b_1, b_0 \rangle$ , donde  $b_i \in \{0, 1\}$  representa la etiqueta de umbralización de cada píxel  $P_i$ , es común transformar  $B(x, y)$  en un código decimal a través de:

$$LBP_{P,R}(x, y) = \sum_{i=0}^{P-1} b_i 2^i. \tag{5.2}$$

Para trabajar a nivel de descripción local mejor que a nivel global, se puede dividir la imagen en  $N_w \times N_w$  celdas donde cada histograma se construye de forma independiente contabilizando la frecuencia de los valores de los códigos LBP. Finalmente, todos los histogramas se combinan en el descriptor (ver figura 5.9). Los parámetros de configuración del operador LBP serán los correspondientes al número de celdas o ventanas en cada una de las dos dimensiones del plano imagen ( $N_w$ ), el número de intervalos del histograma ( $N_b$ ), y por supuesto, el radio y píxeles vecinos ( $P, R$ ) que definen el entorno de vecindad en el muestreo. En esta tesis, como paso previo a la extracción mediante el operador LBP, se ha aplicado sobre cada muestra de test un suavizado de la imagen para evitar grandes discontinuidades en las intensidades de los píxeles adyacentes.

Una variante del operador LBP consiste en combinar e integrar la información procedente de  $N$  operadores LBP, donde cada operador trabaja con unos valores diferentes

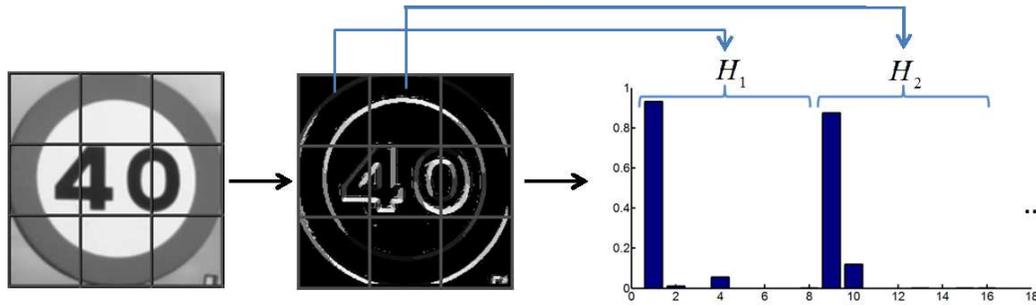


Figura 5.9: Proceso de extracción del descriptor LBP con una partición de la muestra en una rejilla de  $3 \times 3$  celdas.

de  $P$  y  $R$ . A pesar de que los códigos LBP a diferentes radios no son independientes desde el punto de vista estadístico, el análisis multi-resolución refuerza la capacidad de discriminación [Guo10]. No obstante, esta estrategia no ha sido contemplada en esta tesis doctoral debido a que el análisis multi-resolución demanda mayor tiempo de cómputo a la hora de calcular las características en diferentes escalas y realizar un proceso de integración de las mismas.

### 5.3. Análisis comparativo de técnicas de preprocesado y descriptores

El estudio de las dos primeras etapas del sistema de reconocimiento, correspondientes al preprocesado y a la obtención del vector descriptor, tiene como objetivo evaluar el comportamiento de las diferentes técnicas propuestas. Para este fin se ha cuantificado la fiabilidad y el rendimiento del módulo de reconocimiento a la salida de la etapa de clasificación, utilizando en todos los casos las SVM como herramienta de clasificación y la arquitectura OVA basada en estimación de probabilidad, tal como se realizó en la sección 4.4.2.

Sin perder generalidad, el estudio comparativo se ha abordado para el conjunto de señales de prohibición al ser éste el que incluye más muestras en la base de datos creada en el desarrollo de la tesis y a partir de los resultados, se generalizará al resto de señales. La partición de muestras en los conjuntos de entrenamiento y test se ha definido de acuerdo a la sección 4.4.2, donde las muestras se han particionado 5 veces en un conjunto de entrenamiento y otro de test. El conjunto de test se conforma a partir de la selección aleatoria sin reemplazo de la mitad de las muestras y con las restantes se conforma el conjunto de test. Así, los resultados se obtienen como el promedio de los resultados parciales de cada partición. Los algoritmos propuestos fueron codificados en Matlab y ejecutados en una Estación de Trabajo Sun Java W1100z con un procesador de 2.6 GHz bajo el sistema operativo Ubuntu GNU/Linux.

	Parámetros
Normalización	$K = \{1, 150, 250\}$
Contrast-Stretching	$T_{cs1} = \{0.02, 0.08, 0.16\}$
Ecuación	$N_{eq} = \{1, 150, 250\}$

Tabla 5.1: Valores de variación de los parámetros para las técnicas de preprocesado.

	Parámetros			
HOG	$N_w = \{3, 4\}$	$N_b = \{8, 12\}$	–	–
SPIN	$N_f = \{8, 14\}$	$N_d = \{8, 14\}$	–	–
RIFT	$N_\theta = \{8, 14\}$	$N_d = \{8, 14\}$	–	–
LBP	$P = \{4, 8\}$	$R = \{1, 2\}$	$N_w = \{3, 4\}$	$N_b = \{8, 12\}$

Tabla 5.2: Valores de variación de los parámetros de los descriptores.

El primer experimento consiste en valorar la fiabilidad del sistema con cada posible pareja preprocesado/descriptor. Para poder establecer un criterio comparativo que nos de idea de cuáles son las mejores técnicas, se han obtenido los resultados variando los parámetros de los algoritmos de preprocesado y de los descriptores en un rango significativo. Los valores de variación de cada parámetro se indican en las tablas 5.1 y 5.2, respectivamente, para las técnicas de preprocesado y los descriptores sin incluir la técnica de umbralización al haber sido descartada por sus malos resultados. Con la finalidad de que la comparativa entre los cuatro descriptores se produzca en igualdad de condiciones, se ha establecido un límite en el máximo número de características, el cual ha sido fijado en 200. El reto es conseguir un descriptor que alcance resultados de fiabilidad similares al descriptor basado en el barrido directo de píxeles descrito en la sección 4.4.1, pero con menor carga de procesado. La fiabilidad de las diferentes técnicas se ha determinado a partir de la tasa de acierto de positivos correspondiente a una FPR específica. La elección del valor de FPR busca mostrar el desempeño en puntos representativos en los que las diferencias entre curvas sean apreciables, evitando que se produzcan posibles cruces entre las curvas ROC a partir de la FPR seleccionada. Estas premisas han llevado a elegir FPR=5%, lo que significa que una curva será considerada mejor que otra si su TPR es superior para cualquier FPR mayor del 5%. En la tabla 5.3 se indica la mejor combinación de valores de parámetros para los valores especificados en las tablas 5.1 y 5.2, teniendo en cuenta que el número de características resultantes para los descriptores HOG y LBP es de 192 y 196 en el caso de SPIN y RIFT, frente a las 709 componentes del descriptor basado en el barrido de píxeles.

En la figura 5.10 se muestran las correspondientes curvas ROC de cada combinación preprocesado-descriptor para los valores de parámetros de la tabla 5.3. Tal como se observa en la figura 5.10(a), la calidad del descriptor formado por la concatenación directa

(a) Barrido de píxeles: 709 atributos		
Normalización( $K = 150$ )		
Contrast-Stretching( $T_{cs1}=0.02, T_{cs2}=0.98$ )		
Ecuilización( $N_{eq} = 250$ )		
(b) HOG( $N_w = 4, N_b = 12$ ): 192 atributos		
Normalización( $K = 50$ )		
Contrast-Stretching( $T_{cs1}=0.16, T_{cs2}=0.84$ )		
Ecuilización( $N_{eq} = 150$ )		
(c) SPIN( $N_f = 14, N_d = 14$ ): 196 atributos		
Normalización( $K = 50$ )		
Contrast-Stretching( $T_{cs1}=0.08, T_{cs2}=0.92$ )		
Ecuilización( $N_{eq} = 50$ )		
(d) RIFT( $N_\theta = 14, N_d = 14$ ): 196 atributos		
Normalización( $K = 150$ )		
Contrast-Stretching( $T_{cs1}=0.16, T_{cs2}=0.84$ )		
Ecuilización( $N_{eq}=50$ )		
(e) LBP( $P = 8, R = 1, N_w = 4, N_b = 12$ ): 192 atributos		
Normalización( $K = 50$ )		
Contrast-Stretching( $T_{cs1}=0.08, T_{cs2}=0.92$ )		
Ecuilización( $N_{eq} = 50$ )		

Tabla 5.3: Valores óptimos de los parámetros de configuración.

de la luminancia de los píxeles, con una longitud de 709 componentes, se ve reforzada mediante el preprocesado con cualquiera de las técnicas propuestas, resultando la ecualización como la mejor opción al pasar de TPR=96.4 % sin preprocesado a TPR=97.5 % con ecualización. No obstante, es destacable que, prácticamente, se llega a la misma tasa de acierto con el descriptor HOG sin preprocesado (ver figura 5.10(b)), que es construido con tan sólo 192 características y aporta claramente los mejores resultados de entre todos los descriptores (TPR=97.3 % para HOG sin preprocesado frente a TPR=89.8 % con Contrast-Stretching+RIFT, TPR=88 % con Normalización+SPIN y TPR=94.4 % con Normalización+LBP). En cualquier caso, del análisis de las curvas ROC se puede concluir que no existe una técnica de preprocesado que funcione claramente mejor para todos los descriptores simultáneamente. De hecho, normalización y contrast-stretching han sido las técnicas que mejores resultados han aportado en la mayor parte de los descriptores, mientras que la ecualización es la técnica de preprocesado peor parada. No obstante, una excepción puede encontrarse en el barrido de píxeles, donde la ecualización es la técnica de preprocesado con mejor resultado. Curiosamente, el mejor desempeño en el descriptor HOG se obtiene cuando no se realiza previamente ningún preprocesado sobre las muestras. Esta circunstancia puede justificarse debido a que la técnica HOG está basada en el gradiente de la imagen y la manipulación del preprocesado refuerza bordes débiles e introduce otros nuevos que pueden complicar el posterior reconocimiento.

Por otra parte, los resultados confirman que los descriptores RIFT y SPIN no son competentes frente a HOG, siendo la técnica SPIN la que arroja peores resultados. Los descriptores SPIN y RIFT siguen una estrategia similar, en el sentido de que ambas subdividen la ROI en anillos concéntricos y computan un histograma de apariencia (valores de píxel y orientaciones de gradiente) en cada región. La pérdida de información local, al incluir en un mismo histograma la información global de todas las orientaciones de cada región, justifican los pobres resultados conseguidos con ambos descriptores. El descriptor SPIN presenta características que lo hacen apropiado para clasificación de texturas [Lazebnik03], pero éstas no pueden ser extrapoladas a las señales, ya que no poseen una estructura de patrones repetidos. Finalmente, el emergente descriptor LBP parece ser un competidor de la técnica HOG, ya que como podemos comprobar en la gráfica 5.10(e) llega a alcanzar una TPR=94.4 % al utilizar un preprocesado basado en la normalización.

En este contexto, también se ha analizado el tiempo de cómputo por cada combinación preprocesado-descriptor y los perfiles de tiempo se han representado en la figura 5.11, desglosando el tiempo total de procesamiento ( $T_t$ ) en tiempo de preprocesado ( $T_{prep}$ ), tiempo de obtención del descriptor ( $T_{desc}$ ) y tiempo de clasificación de las SVM ( $T_{svm}$ ). Los experimentos fueron lanzados sobre las 5 particiones de datos y el tiempo fue promediado para minimizar las variaciones de los tiempos de ejecución ('jitter'). Como se puede comprobar, la mayor carga del sistema de reconocimiento recae en la etapa de clasificación de las SVM, mientras que la carga del preprocesado es insignificante, aún teniendo en cuenta que las técnicas de contrast-stretching y ecualización son las que más tiempo demandan, y la carga de la extracción del descriptor supone tan sólo un pequeño porcentaje del tiempo

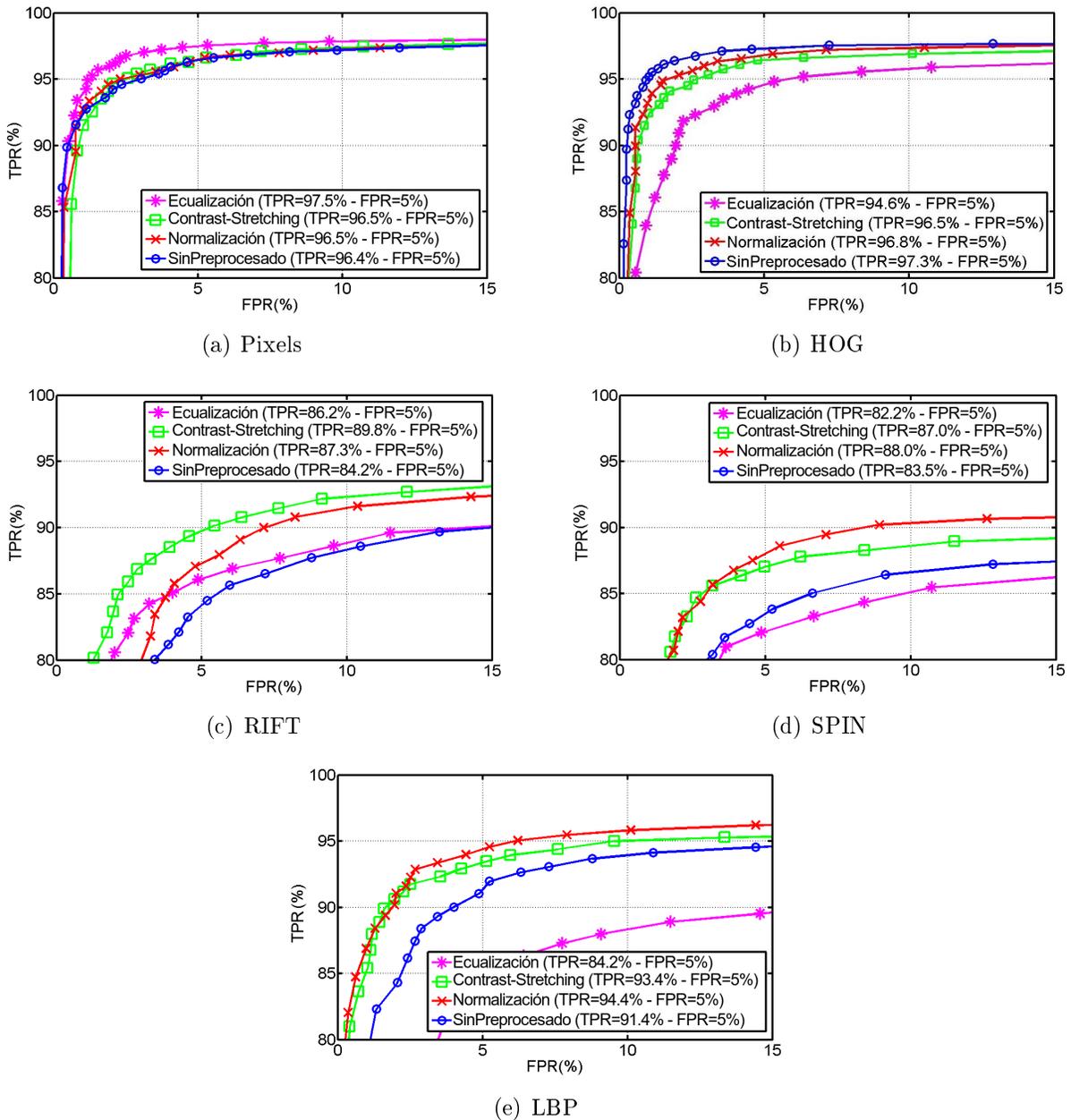


Figura 5.10: Curvas ROC para análisis de técnicas de preprocesado con diferentes descriptores utilizando señales de prohibición. (a) Barrido de la intensidad de los píxeles. (b) HOG. (c) RIFT. (d) SPIN. (e) LBP.

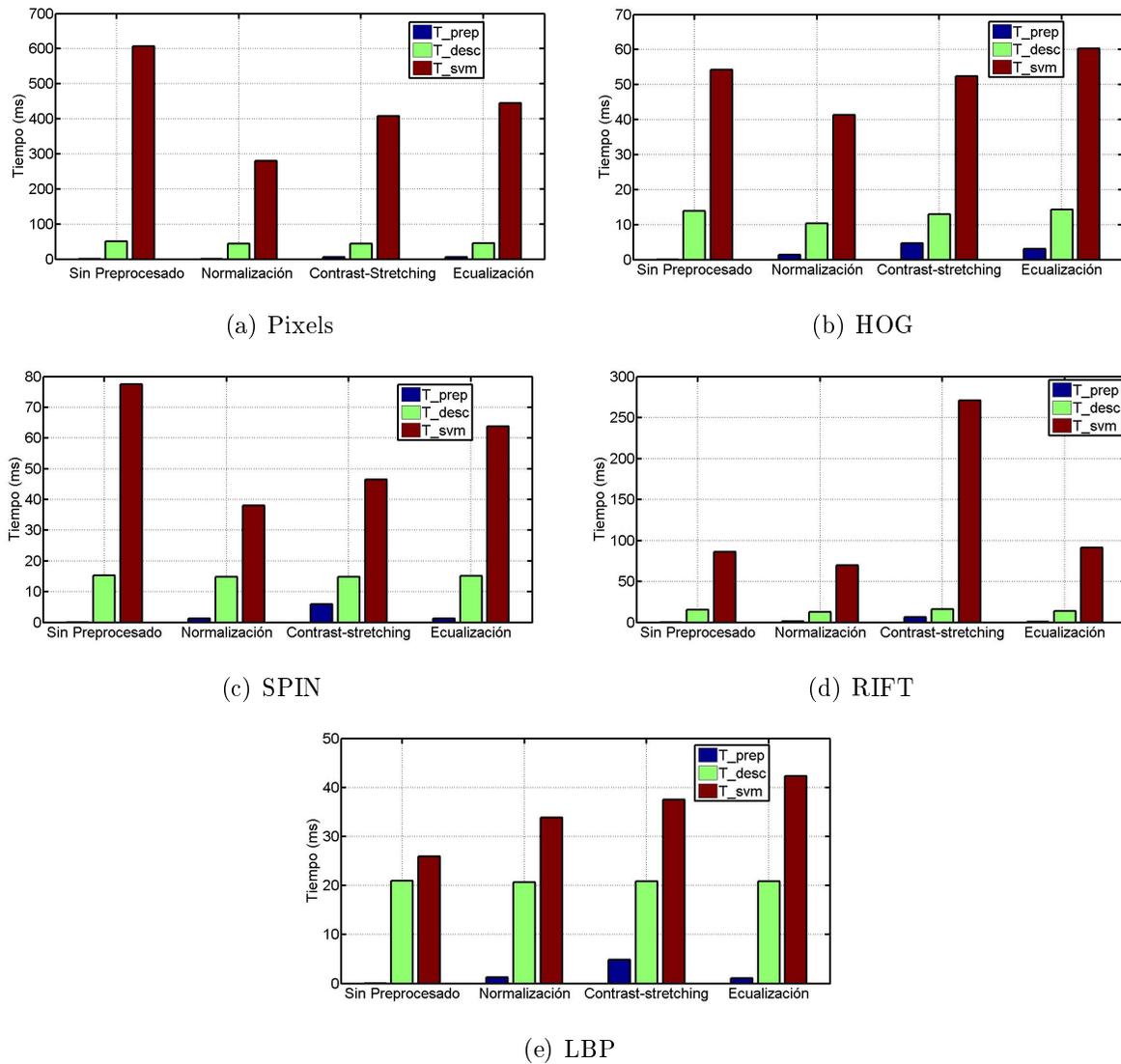


Figura 5.11: Tiempo de cómputo para análisis de técnicas de preprocesado con diferentes descriptores utilizando señales de prohibición. (a) Barrido de la intensidad de los píxeles. (b) HOG. (c) SPIN. (d) RIFT. (e) LBP.

total. En la figura 5.11(a) se puede observar que el tiempo de clasificación en el barrido de píxeles se reduce considerablemente con cualquiera de los descriptores propuestos como consecuencia de la disminución del número de vectores soporte de las SVM generadas.

Para un análisis comparativo más directo, en la tabla 5.4 se recopila la información correspondiente a cada combinación preprocesado-descriptor, incluyendo la TPR para FPR=5 %, el número de vectores soporte que definen la etapa de clasificación y el tiempo total de procesado por muestra, marcándose en negrita la técnica de preprocesado con la que se consigue la mejor tasa de acierto. Tal como se puede comprobar, la reducción de tiempo que se consigue con el descriptor HOG es considerable: 68.1 ms frente a 495.36 ms en el caso de barrido de píxeles con ecualización, lo cual supone una reducción del 86.25 %. Esta gran mejora de tiempo viene dada a través de una doble reducción correspondiente al número de atributos (192 frente a 709 en el método original) y al número de vectores soporte (6763 frente a 9163). En el caso de los descriptores RIFT y SPIN, el tiempo es también menor que en el caso del barrido original de píxeles, pero ahora debido tan sólo al menor número de atributos del descriptor. De hecho, es importante observar en la tabla 5.4 que el número de vectores soporte es significativamente mayor para los descriptores SPIN y RIFT que en el caso del método original, lo cual viene a corroborar problemas en la homogeneidad de los datos y en la generalización del problema de clasificación. Por su parte, la técnica LBP mantiene tasas de detección cercanas al descriptor HOG con un buen rendimiento, ya que los tiempos de procesado con Normalización son comparables a los conseguidos con HOG. En la tabla 5.4 se puede observar que no existe en todos los casos una relación de proporcionalidad entre el número de vectores soporte y el tiempo de procesado, lo cual responde a la forma de implementación de la librería LIBSVM debido a que utiliza una matriz dispersa para el almacenamiento de los vectores soporte. El almacenamiento y procesado de las estructuras de datos basadas en matrices dispersas consiguen una buena optimización cuando un alto porcentaje de los elementos son nulos. Precisamente, al porcentaje de elementos nulos de la matriz es denominado densidad y como ejemplo, los valores de densidad de la matriz de los vectores soporte sin utilización de ninguna técnica de preprocesado son: 1.64 % en HOG, 65.33 % en SPIN, 21.23 % en RIFT y 57.32 % en LBP.

Como conclusión de este análisis experimental, se puede afirmar que las técnicas HOG sin preprocesado y LBP con normalización se perfilan como las mejores candidatas para el sistema de reconocimiento de esta tesis doctoral en base a sus buenas prestaciones.

### 5.3.1. Estudio paramétrico de los descriptores elegidos

Los parámetros de ajuste de cada descriptor influyen de manera significativa en el desempeño del sistema. Con el propósito de que los resultados de la sección anterior no se vean sesgados por la elección de los valores de los parámetros, en esta sección se realizará un análisis paramétrico de los dos mejores descriptores: HOG y LBP.

(a) Barrido de píxeles: 709 atributos

Técnica Preprocesado	$TPR(\%)$	$N_{sv}$	$T_t(\text{ms})$
–	96.4	10142	657.7
Normalización	96.5	7648	326.3
Contrast-Stretching	96.5	7428	459.6
Ecuilización	<b>97.5</b>	<b>9163</b>	<b>495.3</b>

(b) HOG: 192 atributos

Técnica Preprocesado	$TPR(\%)$	$N_{sv}$	$T_t(\text{ms})$
–	<b>97.3</b>	<b>6763</b>	<b>68.1</b>
Normalización	96.8	5396	52.9
Contrast-Stretching	96.5	6102	70.0
Ecuilización	94.6	7049	77.5

(c) SPIN: 196 atributos

Técnica Preprocesado	$TPR(\%)$	$N_{sv}$	$T_t(\text{ms})$
–	83.5	18375	92.7
Normalización	<b>88.0</b>	<b>14083</b>	<b>54.2</b>
Contrast-Stretching	87.0	12162	67.1
Ecuilización	82.2	13005	80.1

(d) RIFT: 196 atributos

Técnica Preprocesado	$TPR(\%)$	$N_{sv}$	$T_t(\text{ms})$
–	84.2	11431	101.9
Normalización	87.3	8992	83.6
Contrast-Stretching	<b>89.8</b>	<b>10944</b>	<b>293.1</b>
Ecuilización	86.2	10942	106.0

(e) LBP: 192 atributos

Técnica Preprocesado	$TPR(\%)$	$N_{sv}$	$T_t(\text{ms})$
-	91.4	6380	46.8
Normalización	<b>94.4</b>	<b>8347</b>	<b>55.7</b>
Contrast-Stretching	93.4	8698	63.1
Ecuilización	84.2	9333	64.1

Tabla 5.4: Análisis comparativo de resultados para diferentes técnicas de preprocesado y extracción de descriptores.

	$N_w = 3$		$N_w = 4$		$N_w = 5$		$N_w = 6$		$N_w = 7$	
	$TPR$	$T_t$								
$N_b = 6$	87.7	25.6	94.6	32.8	97.6	45.0	98.8	69.0	99.0	173.3
$N_b = 7$	91.4	25.9	95.9	35.7	97.9	51.8	98.9	122.2	99.0	210.3
$N_b = 8$	87.9	30.6	94.51	47.0	97.6	72.3	98.6	145.5	98.9	452.2
$N_b = 9$	91.7	32.1	96.5	47.34	98.1	77	99.0	154.8	99.1	537.8
$N_b = 10$	92.4	33.9	96.7	63.4	98.3	84.8	98.9	158.3	99.0	678.6
$N_b = 11$	93.9	39.4	97.1	61.5	98.4	114.0	98.9	250.5	98.9	892.8
$N_b = 12$	93.3	45.0	97.3	68.1	98.5	129.1	98.9	252.9	98.8	1096.1

Tabla 5.5: Resultados comparativos de HOG ( $TPR(\%)$  y  $T_t(\text{ms})$ ) en función de los parámetros  $N_b$  y  $N_w$ .

En concreto, el descriptor HOG viene determinado por dos parámetros de configuración: el número de celdas en vertical y horizontal, que en esta tesis se supondrán iguales entre sí y será denotado por  $N_w$ , y el número de intervalos en que se discretizan las orientaciones del gradiente en cada celda ( $N_b$ ). En la tabla 5.5 se muestra la tasa de acierto de las muestras positivas  $TPR(\%)$  para una  $FPR=5\%$  y el tiempo total de cómputo por muestra  $T_t(\text{ms})$  para un rango de valores de  $N_w$  y  $N_b$  que proporcione tasas de acierto significativas (en concreto,  $3 \leq N_w \leq 7$  y  $6 \leq N_b \leq 12$ ). Como puede comprobarse, la tasa de acierto del sistema crece de forma general conforme aumentan los valores de  $N_w$  y  $N_b$ , aunque cabe destacar dos aspectos: 1) la tasa de acierto apenas mejora para  $N_b \geq 10$  siempre que  $N_b \geq 5$  e incluso en algún caso sufre una ligera degradación, y 2) la tasa de acierto también aumenta según dividimos los blobs en un mayor número de celdas, aunque el sistema se sobrecarga para  $N_w = 7$ , aumentando considerablemente el tiempo de procesado respecto a las situaciones con  $N_w < 7$ . Es importante destacar que el descriptor HOG con normalización consigue una  $TPR=97.6\%$  ( $N_w = 5$  y  $N_b = 6$ ) con un tiempo de carga por muestra  $T_t=45.0$  ms frente a los mejores resultados que se habían obtenido con barrido de píxeles (mediante ecualización previa) con  $TPR=97.5\%$  y  $T_t=495.3$  ms.

Por su parte, el operador LBP define los píxeles de muestreo a partir del radio y del número de puntos a muestrear mediante los parámetros  $P$  y  $R$ . En la tabla 5.6 se muestran los resultados obtenidos para la combinación Normalización+LBP con  $R = \{1, 2, 3, 4\}$  y  $P = \{4, 8, 16\}$ , habiendo fijado  $N_b = 8$  y  $N_w = 5$ . El hecho de mantener fijados  $N_b$  y  $N_w$  hace que el número de atributos del descriptor sea igual para todos los casos y la diferencia de carga computacional vendrá dada por el número de vectores soporte. Como se puede apreciar en la tabla 5.6, no existen diferencias significativas en la tasa de acierto si se varía  $P$ , siempre que  $R \leq 3$ , lo cual quiere decir que las relaciones entre píxeles a partir de un entorno de vecindad de radio igual a 4 hacen decaer la discriminabilidad. Asimismo, se observa que los mejores resultados se obtienen con  $P = 8$ , mientras que para  $P = 4$  la información es más reducida y para  $P = 16$  se pierde información al intentar

	$P = 4$		$P = 8$		$P = 16$	
	$TPR$	$N_{sv}$	$TPR$	$N_{sv}$	$TPR$	$N_{sv}$
$R = 1$	92.6	12264	94.4	10012	93.2	8665
$R = 2$	93.4	12304	94.1	10429	92.5	9447
$R = 3$	93.8	11871	94.2	10089	93.5	9605
$R = 4$	86.2	16874	88.5	14395	90.6	12479

Tabla 5.6: Resultados comparativos de Normalización+LBP ( $TPR(\%)$  y  $N_{sv}$ ) en función de los parámetros  $P$  y  $R$ .

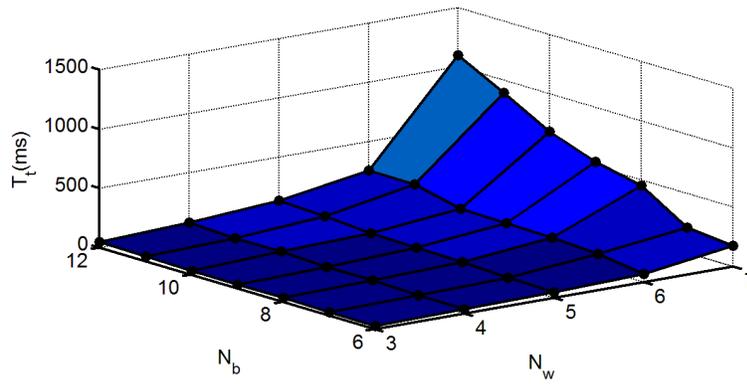
	$N_w = 3$		$N_w = 4$		$N_w = 5$		$N_w = 6$		$N_w = 7$	
	$TPR$	$T_t$								
$N_b = 8$	87.0	30.7	92.2	42.9	94.4	75.6	96.2	127.1	93.5	278.2
$N_b = 12$	90.6	36.7	94.4	55.7	95.2	105.1	96.5	178.3	94.0	349.7
$N_b = 16$	90.9	42.1	94.8	69.4	95.8	131.1	96.5	268.1	94.6	443.4
$N_b = 20$	92.6	49.7	95.1	87.2	96.2	157.9	96.9	335.2	94.7	618.6

Tabla 5.7: Resultados comparativos de Normalización+LBP ( $TPR(\%)$  y  $T_t(\text{ms})$ ) en función de los parámetros  $N_b$  y  $N_w$ .

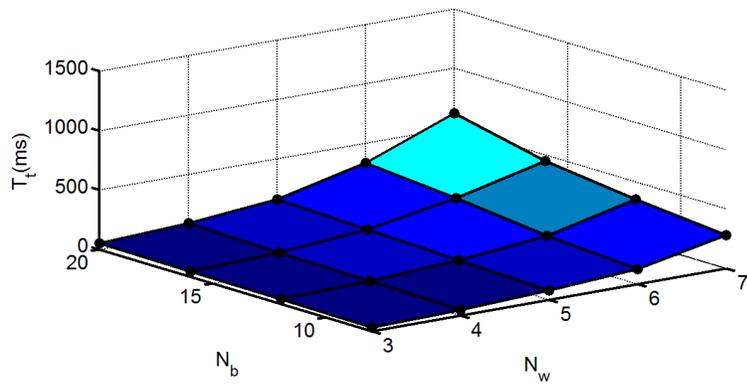
discretizar el número de posibles códigos ( $2^{16}=65536$ ) en tan sólo 8 intervalos. De acuerdo a estos criterios, la combinación elegida es ( $R = 1, P = 8$ ).

Una vez optimizados los parámetros ( $P, R$ ) del descriptor LBP, se ha realizado un análisis similar al del descriptor HOG variando el número de celdas ( $N_w$ ) y el número de intervalos en que se discretiza el histograma de cada celda ( $N_b$ ). En la tabla 5.7 se muestra la tasa de acierto  $TPR(\%)$  y el tiempo total de cómputo por muestra  $T_t(\text{ms})$  para valores de  $N_w$  y  $N_b$  que proporcionen tasas de acierto significativas (en concreto,  $3 \leq N_w \leq 7$  y  $N_b = \{8, 12, 16, 20\}$ ). En este caso, las variaciones de la tasa de acierto con respecto a  $N_b$  son más lentas que en el caso del HOG y por ello, se muestran los resultados para valores discretos de  $N_b$ , en vez de para un rango continuo. Como se desprende de los resultados, también el sistema se sobrecarga con el operador LBP para  $N_w \geq 7$ , aunque el ‘pico’ no es tan acusado como en el caso del descriptor HOG. En la figura 5.12 se representa gráficamente el tiempo demandado por el sistema de reconocimiento en función de  $N_w$  y  $N_b$ , tanto para el descriptor HOG como para el descriptor LBP, observándose el ‘pico’ de tiempo sobrevenido para  $N_w \geq 7$ .

Finalmente, en la figura 5.13 se representa gráficamente la evolución de la tasa de acierto de las tablas 5.5 y 5.7 en función del tiempo de procesado, habiendo considerado el rango temporal común de los resultados de ambos experimentos para  $3 \leq N_w \leq 6$ . Se observa claramente la superioridad de las prestaciones del descriptor HOG frente a las de la combinación Normalización+LBP.



(a)



(b)

Figura 5.12: Tiempo de procesado por muestra en función de los parámetros  $N_b$  y  $N_w$ . (a) HOG. (b) LBP.

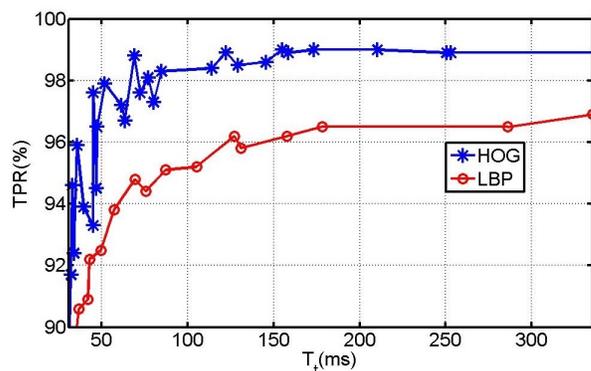


Figura 5.13: Comparativa entre los descriptores HOG y LBP.

## 5.4. Arquitecturas de multclasificación basadas en SVM

Las SVM, descritas en el apéndice B, son un clasificador binario en su formulación original. No obstante, en la práctica se presentan con frecuencia problemas multiclase en los que se debe determinar la clase correcta entre varias posibilidades. Aunque algunos clasificadores binarios pueden extenderse fácilmente a escenarios de multclasificación, no es el caso del paradigma de las SVM. De acuerdo a [Duin05], el problema de multclasificación es todavía un campo abierto de investigación y en la literatura se pueden encontrar diversos trabajos que abordan el problema multiclase de forma conjunta, de modo que se modifica la formulación de las SVM con el objetivo de resolver un único problema de optimización que considere todas las clases a la vez ([Vapnik95], [Crammer01] y [Perez05]). Las diferencias entre los distintos métodos son sutiles y consisten fundamentalmente en la forma en que se penalizan las muestras mal clasificadas. En cualquier caso, la aplicación práctica de estos métodos queda restringida a problemas relativamente sencillos debido a la alta complejidad computacional que presentan. Su formulación se complica sustancialmente respecto a la de la SVM binaria, como consecuencia de considerar simultáneamente todas las clases y muestras. Esto se traduce en una mayor complejidad de los métodos de descomposición usados y resulta inviable cuando el número de clases y de muestras es elevado.

Por los motivos expuestos, es común simplificar el problema de multclasificación con SVM a partir de clasificadores binarios de naturaleza más simple, a los cuales es usual referirse como dicotomías o clasificadores base. La estrategia de resolución de los problemas de multclasificación mediante clasificadores binarios implica la descomposición del problema en varias dicotomías y la consiguiente combinación de sus salidas para realizar la oportuna predicción. En un escenario de multclasificación supervisada con  $M$  clases, dispondremos de un conjunto de entrenamiento  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$ , donde  $\mathbf{x}_i \in \mathbb{R}^d$ ,  $i = 1, \dots, N$ , e  $y_i \in \{1, \dots, M\}$  es la clase del vector  $\mathbf{x}_i$ . Si consideramos  $H$  como un conjunto de clasificadores que representa una descomposición del problema en  $N_h$  problemas binarios más

simples, cada clasificador base fijará su propia función de decisión. La mayoría de las aproximaciones que intentan resolver la multclasificación se pueden incluir en una de las tres siguientes estrategias: 1) ‘uno contra todos’ (OVA, *One-vs-All*), 2) ‘uno contra uno’ (OVO, *One-vs-One*) y 3) códigos correctores de error (ECOC, *Error Correcting Output Codes*).

### 5.4.1. Método ‘Uno contra todos’

En la estrategia ‘uno contra todos’ (OVA, *One-vs-All*) cada clasificador considera los patrones de su propia clase como positivos (+1) y los pertenecientes al resto de clases como negativos (-1). De esta manera, se cumple que el número de clasificadores  $N_h$  coincide con el número de clases  $M$ . En el proceso de testeo, cada instancia es asignada a la clase con mayor valor de salida. En esta tesis doctoral, se emplean dos estrategias dentro de la arquitectura OVA para abordar la identificación de las muestras ruidosas. En la primera, a la que nos referiremos como OVA-1 y que ha sido la utilizada hasta ahora en esta tesis doctoral, se incluyen las muestras de ruido como negativas en el entrenamiento de todos los clasificadores, pero no se construye un clasificador explícito para el ruido. En cualquier caso, aquellas muestras de test para las que todas las probabilidades de salida sean inferiores a un umbral dado serán descartadas como ruido. La segunda estrategia (OVA-2) consiste en considerar el ruido como una clase característica y construir un clasificador propio para el mismo. En el caso OVA-2 las muestras de test siempre son asignadas a alguna de las clases. Según lo expuesto, el número de clasificadores será  $N_h = M - 1$  para OVA-1 y  $N_h = M$  para OVA-2. La figura 5.14 muestra un ejemplo OVA con 5 clases, mostrándose las fronteras de decisión de cada SVM.

Los dos principales problemas asociados a esta arquitectura son los siguientes:

- Un elevado número de clases puede dar lugar a problemas de clasificación muy desbalanceados en los que la clase positiva en cada clasificador binario está mucho menos representada que la negativa. Esta circunstancia puede provocar un desplazamiento de la frontera de decisión, introduciendo un sesgo artificial.
- Cada clasificador binario se entrena con todas las muestras disponibles, por lo que la complejidad de cada uno de los clasificadores puede resultar muy elevada. Esto conlleva un elevado número de vectores soporte para determinar la frontera de decisión y un alto coste computacional en las etapas de entrenamiento y test.

Un método para obtener la probabilidad a posteriori de cada clasificador binario en la arquitectura OVA fue descrito anteriormente en la sección 4.4.2.

### 5.4.2. Método ‘Uno contra uno’

En la estrategia ‘uno contra uno’ (OVO, *One-vs-One*) se construye un número de clasificadores  $N_h = \frac{M(M-1)}{2}$ . En esta estrategia se tiene un clasificador para cada pareja de

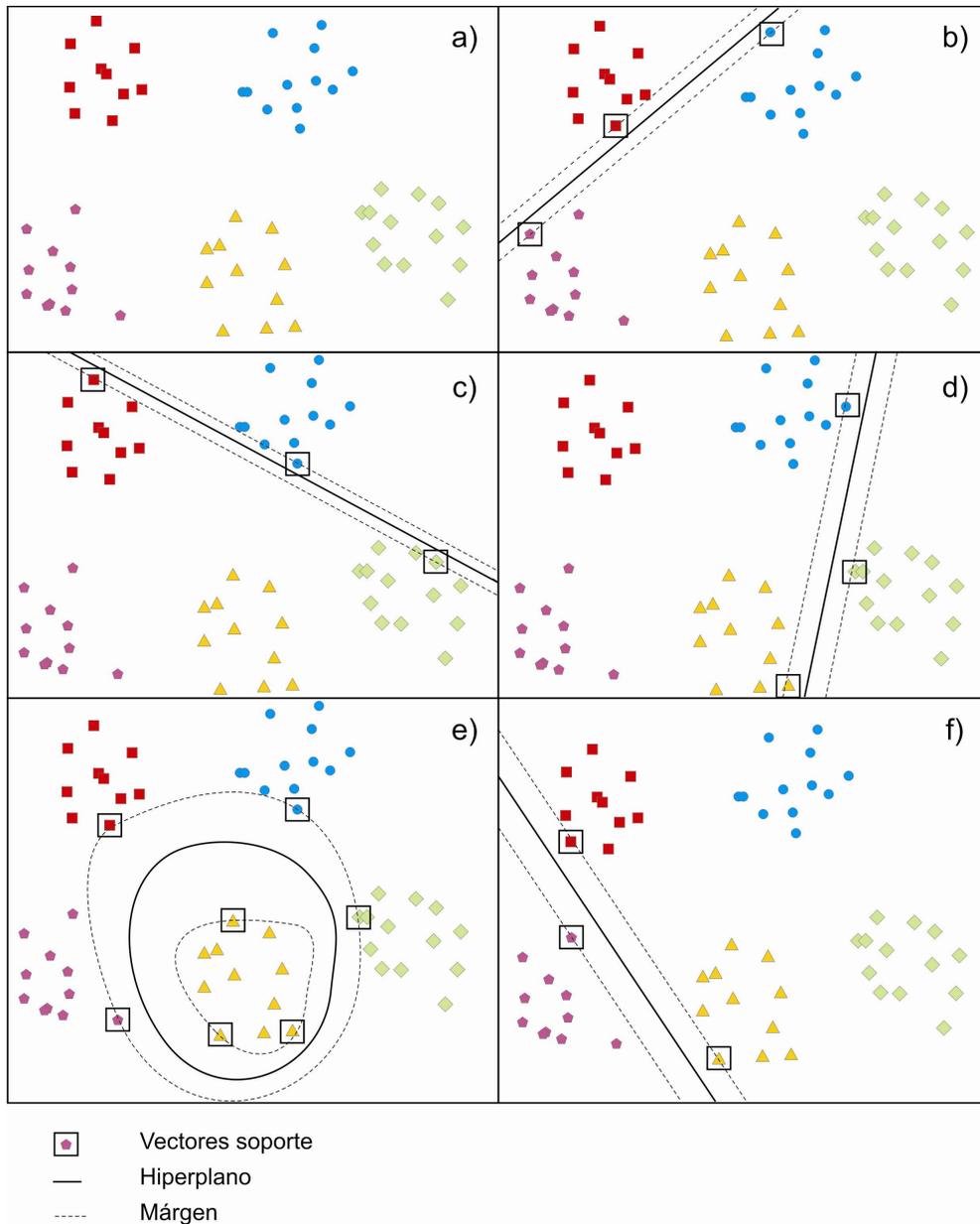


Figura 5.14: Ejemplo de clasificación OVA con  $M = 5$  clases. a) Representación de las 5 clases en un espacio bidimensional. b) Clase 1 contra el resto:  $y_1 = +1; y_{2,3,4,5} = -1$ . c) Clase 2 contra el resto:  $y_2 = +1; y_{1,3,4,5} = -1$ . d) Clase 3 contra el resto:  $y_3 = +1; y_{1,2,4,5} = -1$ . e) Clase 4 contra el resto:  $y_4 = +1; y_{1,2,3,5} = -1$ . f) Clase 5 contra el resto:  $y_5 = +1; y_{1,2,3,4} = -1$ .

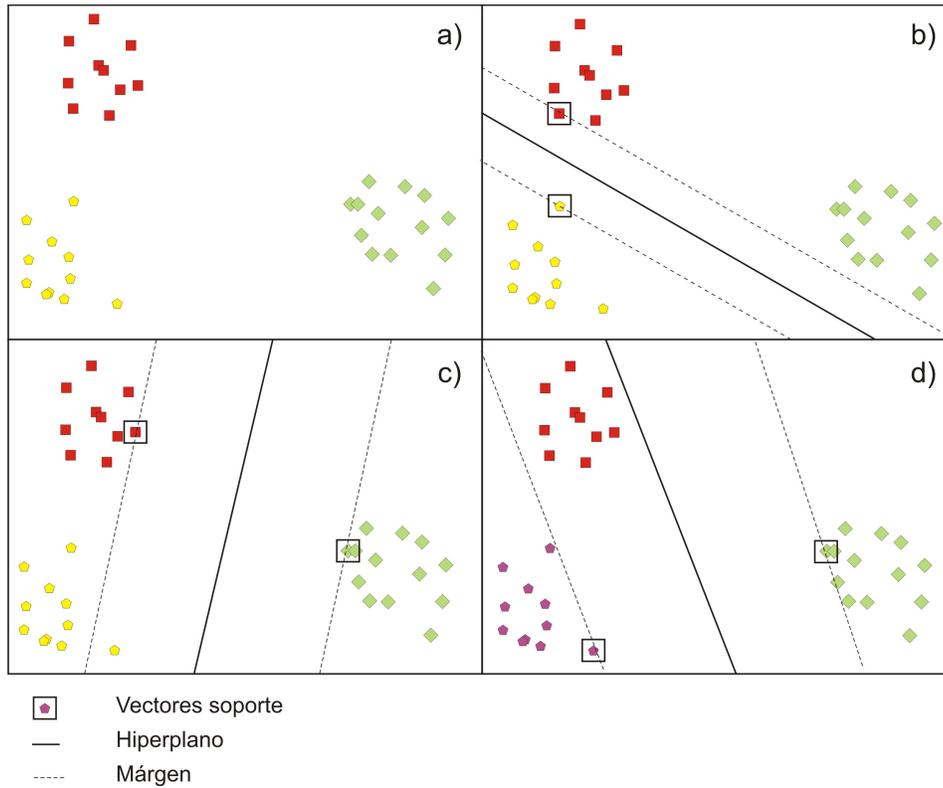


Figura 5.15: Ejemplo de clasificación 1-vs-1 con  $M = 3$  clases. a) Representación de las 3 clases en un espacio bidimensional. b) Clase 1 contra clase 2:  $y_1 = +1; y_2 = -1$ . c) Clase 1 contra clase 3:  $y_1 = +1; y_3 = -1$ . d) Clase 2 contra clase 3:  $y_2 = +1; y_3 = -1$ .

clases  $(\mathcal{C}_i, \mathcal{C}_j)$ , el cual utiliza durante el entrenamiento las muestras de la clase  $\mathcal{C}_i$  como positivas y las de la clase  $\mathcal{C}_j$  como negativas, ignorando las muestras del resto de clases. La figura 5.15 muestra los hiperplanos de decisión de cada SVM en un sistema OVO para un ejemplo con tres clases. A la hora de combinar las salidas de los clasificadores, existen dos principales técnicas: votación y grafos dirigidos (DDAG, *Decision directed acyclic graphs*). La técnica de votación sugiere que cada clasificador base emite un voto para una clase y la clase ganadora es aquella que acumula un mayor número de votos. El problema de esta arquitectura es que el número de clasificadores base crece con  $M^2$  y para un número elevado de clases, puede ser excesivamente lenta. En el caso del DDAG, donde cada nodo es un clasificador binario de las clases  $\mathcal{C}_i$  y  $\mathcal{C}_j$ , el grafo a construir posee un número de nodos igual a  $\frac{M(M-1)}{2}$  y un número de terminales igual a  $M$ . Dada una muestra de test  $\mathbf{x}$ , se comienza a evaluar la función de decisión binaria desde el nudo raíz y dependiendo del valor de salida, se mueve a la izquierda o a la derecha sucesivamente, hasta alcanzar un camino que lleva a alcanzar el nudo terminal de una clase. Este nudo terminal alcanzado se corresponde con la clase predicha.

En [Hsu02] se argumenta como principal ventaja del método OVO el hecho de que los problemas de clasificación son más equilibrados y fácilmente separables que en la estrategia OVA, de tal modo que en problemas con una cierta complejidad (muchas clases con una elevada superposición entre ellas y muchas muestras), la arquitectura OVO puede proporcionar mejores resultados. No obstante, en el esquema de clasificación OVO existen dos inconvenientes a tener en cuenta: 1) cada clasificador considera únicamente la información de dos clases, ya que es entrenado sobre parte del conjunto total de entrenamiento,  $\mathcal{T}r$ , en concreto sobre  $\mathcal{T}r_i \cup \mathcal{T}r_j$ , siendo  $\mathcal{C}_i$  y  $\mathcal{C}_j$  dos clases del conjunto  $\mathcal{T}r$ , y 2) el tamaño del módulo de clasificación asociado a los nodos crece de forma exponencial conforme aumenta el número de clases  $M$ .

La estimación de probabilidad en la arquitectura OVO exige calcular la probabilidad de Platt para la muestra de entrada  $\mathbf{x}$  en cada SVM binaria  $(i, j) \forall i, j = 1, 2, \dots, M$ :

$$r_{ij} = P(y = i | y = i \text{ ó } y = j, \mathbf{x}) = \frac{1}{1 + \exp(A_{ij}f_{ij}(\mathbf{x}) + B_{ij})} \quad (5.3)$$

y

$$r_{ji} = P(y = j | y = i \text{ ó } y = j, \mathbf{x}) = 1 - r_{ij}, \quad (5.4)$$

siendo  $f_{ij}(\mathbf{x})$  la salida de la SVM binaria  $(i, j)$ .

El siguiente paso consiste en encontrar las probabilidades multiclase  $P(y = i | \mathbf{x}) \forall i = 1, 2, \dots, M$  compatibles con las probabilidades binarias  $r_{ij}$ . Partiendo de la aproximación sugerida en [Bradley52]:

$$r_{ij} \approx \frac{P(y = i | \mathbf{x})}{P(y = i | \mathbf{x}) + P(y = j | \mathbf{x})}, \quad (5.5)$$

en [Refregier91] se propone resolver un sistema lineal formado por  $M - 1$  ecuaciones del tipo:

$$r_{ij}P(y = j | \mathbf{x}) = r_{ji}P(y = i | \mathbf{x}) \quad (5.6)$$

junto con otra que fuerce que la suma de probabilidades sea igual a la unidad. No obstante, la calidad de estimación de las probabilidades a posteriori depende de la elección del conjunto  $M - 1$  de probabilidades binarias. Por ello, en [Price94] se propone un método alternativo en el que en el cálculo de la probabilidad  $P(y = i | \mathbf{x})$  intervienen únicamente las probabilidades binarias en las que participa la clase:

$$P(y = i | \mathbf{x}) = \frac{1}{\sum_{j:j \neq i} \frac{1}{r_{ij}} - (M - 2)}. \quad (5.7)$$

### 5.4.3. Códigos correctores de Error

Dietterich y Bakiri [Dietterich95] presentaron un escenario en el que el problema de multclasificación es resuelto de acuerdo a un conjunto de códigos binarios correctores de error (ECOC, *Error Correcting Output Codes*). La técnica ECOC se fundamenta en dos pasos: una etapa de codificación, en la que se asigna una palabra código a cada una de

(a)

	$h_1$	$h_2$	$h_3$	$h_4$
$\mathcal{C}_1$	+1	-1	-1	-1
$\mathcal{C}_2$	-1	+1	-1	-1
$\mathcal{C}_3$	-1	-1	+1	-1
$\mathcal{C}_4$	-1	-1	-1	+1

(b)

	$h_1$	$h_2$	$h_3$	$h_4$	$h_5$	$h_6$
$\mathcal{C}_1$	+1	+1	+1	0	0	0
$\mathcal{C}_2$	-1	0	0	+1	+1	0
$\mathcal{C}_3$	0	-1	0	-1	0	+1
$\mathcal{C}_4$	0	0	-1	0	-1	-1

Tabla 5.8: Matrices para clasificación de 4 clases con arquitectura OVA(a) y OVO(b).

las clases, y otra de decodificación, en la que se busca para cada vector de entrada  $\mathbf{x}$  la clase con código más similar al de éste.

El objetivo de la etapa de codificación es crear una palabra-código para cada una de las  $M$  clases del problema. Ordenando los códigos como filas de una matriz, se define la matriz de codificación  $\mathbf{A} \in \{-1, 1\}^{M \times N_h}$  en el caso binario y  $\mathbf{A} \in \{-1, 0, 1\}^{M \times N_h}$  en el caso ternario. De esta manera, se obtienen  $M$  códigos, donde cada uno posee una longitud  $N_h$ . Desde el punto de vista de la teoría de aprendizaje, la matriz  $\mathbf{A}$  es interpretada como un conjunto de  $N_h$  problemas de clasificación binarios; uno por cada columna. Así, cada columna define una partición de clases y cada clase es codificada como ‘+1’ ó ‘-1’ de acuerdo a su pertenencia o no a la dicotomía. Además, en el caso ternario, el símbolo ‘0’ indica que una clase particular no es significativa para una dicotomía específica. En la tabla 5.8 se muestran las matrices de codificación ECOC para las estrategias OVA y OVO con  $N_h = 4$ .

En el proceso de test se calcula la distancia  $d(\mathbf{A}(m, \cdot), \mathbf{o}(\mathbf{x}))$  entre el código  $\mathbf{o}(\mathbf{x})$  asignado al vector  $\mathbf{x}$  en la decodificación y cada una de las palabras código  $\mathbf{A}(m, \cdot)$  correspondiente a las clases definidas en la matriz  $\mathbf{A}$ . La etiqueta de predicción para  $\mathbf{x}$  viene dada por la clase con menor distancia de acuerdo a la siguiente ecuación:

$$\hat{y} = \underset{m}{\operatorname{argmin}}[d(\mathbf{A}(m, \cdot), \mathbf{o}(\mathbf{x}))]. \quad (5.8)$$

### Codificación ECOC

La codificación de los métodos ECOC se puede dividir, de acuerdo al número de símbolos empleados, en binaria y ternaria. Dentro de las codificaciones binarias, se encuentra la arquitectura OVA ya mencionada y el paradigma de aleatoriedad densa [Allwein02]. Esta

última técnica genera un alto número de códigos aleatorios, donde se fija la probabilidad de los símbolos  $\{+1, -1\}$  en los códigos (usualmente,  $P(1) = P(-1) = 0.5$ ). El análisis en la implementación de la estrategia de aleatoriedad densa [Allwein02] ha sugerido una longitud óptima de  $N_h = 10\log_2(M)$  bits por código. Para el conjunto de códigos generados, la matriz óptima es aquella que maximiza la distancia de Hamming entre sus filas y columnas. En la tabla 5.9 se muestra una matriz de aleatoriedad densa para cuatro clases, construida sobre 5 clasificadores binarios.

Respecto a los códigos ternarios ECOC, las dos técnicas clásicas se fundamentan en la estrategia OVO y la aleatoriedad dispersa [Allwein02]. La codificación basada en códigos aleatorios dispersos toma sus valores del conjunto  $\{+1, 0, -1\}$ . Cada elemento de la matriz  $\mathbf{A}$  puede ser '0' con probabilidad  $1/2$  y '+1' o '-1' con probabilidad  $1/4$  cada uno. La longitud óptima del código es fijada en este caso a  $N_h = 15\log_2(M)$  y la matriz se elige entre un conjunto de matrices como aquella cuya menor distancia de Hamming entre sus códigos es mayor. Un ejemplo de matriz de aleatoriedad dispersa para cuatro clases puede encontrarse en la tabla 5.9.

Las estrategias de codificación aleatoria, tanto para códigos densos como dispersos, satisfacen dos propiedades:

- Separación de filas. Cada palabra  $\mathbf{A}(m, \cdot)$  con  $m = 1, \dots, M$  debería ser fácilmente separable del resto de filas de la matriz  $\mathbf{A}$ . Esta propiedad es conseguida si la distancia de Hamming entre una fila y el resto, incluidos sus complementarios, es grande.
- Separación de columnas. Cada vector columna  $(\mathbf{A}(\cdot, n))^T$  con  $n = 1, \dots, N_h$  debería ser incorrelado con el resto de columnas de la matriz. Esta propiedad es conseguida si la distancia de Hamming entre una columna y el resto, incluidos sus complementarios, es grande.

Uno de los inconvenientes de las estrategias de codificación descritas es que no tienen en cuenta el problema de clasificación a tratar y para poder generalizar correctamente, el criterio que se utiliza es que los códigos entre filas y entre columnas estén bien separados en términos de la distancia Hamming. Debido al gran número de bits implicados en estas estrategias clásicas de codificación, se han diseñado nuevas técnicas que explotan el conocimiento de los datos del problema. De esta manera, se trata de conseguir buenos resultados de clasificación con códigos de pequeña longitud. Entre los más destacados, cabe citar la técnica DECOC [Pujol06], la cual sigue una codificación ternaria.

El método DECOC (DECOC, *Discriminant Error Correcting Output Codes*) aborda la clasificación ECOC mediante árboles binarios, de tal modo que en cada paso los nodos buscan la mejor partición de clases. El proceso es repetido hasta que se obtienen nodos terminales con una única clase. De esta manera, la codificación de la matriz  $\mathbf{A}$  se aborda en dos pasos:

(a)

	$h_1$	$h_2$	$h_3$	$h_4$	$h_5$
$\mathcal{C}_1$	-1	-1	+1	+1	+1
$\mathcal{C}_2$	+1	-1	+1	-1	-1
$\mathcal{C}_3$	-1	+1	-1	+1	-1
$\mathcal{C}_4$	+1	-1	+1	-1	+1

(b)

	$h_1$	$h_2$	$h_3$	$h_4$	$h_5$
$\mathcal{C}_1$	+1	-1	+1	+1	0
$\mathcal{C}_2$	+1	0	+1	-1	+1
$\mathcal{C}_3$	-1	-1	0	0	+1
$\mathcal{C}_4$	0	+1	-1	-1	-1

Tabla 5.9: Matrices para clasificación de 4 clases con arquitectura de aleatoriedad densa(a) y aleatoriedad dispersa(b).

1. Creación del árbol binario de manera recursiva, encontrando en cada paso la partición binaria más discriminante asociada al conjunto de clases del nudo padre  $n$ ,  $\{\wp_n^1, \wp_n^2\}$ .
2. Asignación del código relativo a la partición  $\{\wp_n^1, \wp_n^2\}$  a la columna  $n$  de la matriz  $\mathbf{A}$ .

La partición de cada nodo debe satisfacer la condición de ser altamente separable en términos de discriminatoriedad y para ello se emplea como criterio la información mutua media entre los datos y las etiquetas creadas para tal partición  $d$ .

En el algoritmo,  $d$  es una variable aleatoria discreta, de tal modo que dada una partición binaria  $\{\wp_n^1, \wp_n^2\}$  del conjunto  $S_n$ ,  $\{\wp_n^1, \wp_n^2\} = BP(S_n)$ ,  $d$  es definido en los siguientes términos:

$$d = d(\mathbf{x}, BP(S_n)) = \begin{cases} +1 & \text{si } \mathbf{x} \in \mathcal{C}_m | \mathcal{C}_m \in \wp_n^1 \\ -1 & \text{si } \mathbf{x} \in \mathcal{C}_m | \mathcal{C}_m \in \wp_n^2. \end{cases} \quad (5.9)$$

El árbol debe ser visto como un medio de encontrar las palabras código. El segundo paso es el proceso de rellenar la matriz ECOC, donde las columnas  $\mathbf{A}(\cdot, n)^T$  se corresponden con los códigos obtenidos en cada nodo. Por tanto, dada una cierta clase  $\mathcal{C}_m$  y el conjunto  $\{\wp_n^1 \cup \wp_n^2\}$  asociado al nodo  $n$ , donde  $\wp_n^1$  y  $\wp_n^2$  son los conjuntos de clases para cada uno de los nodos hijos del nodo  $n$ , el valor de cada componente  $A_{m,n}$  de la matriz  $\mathbf{A}$  viene dado por:

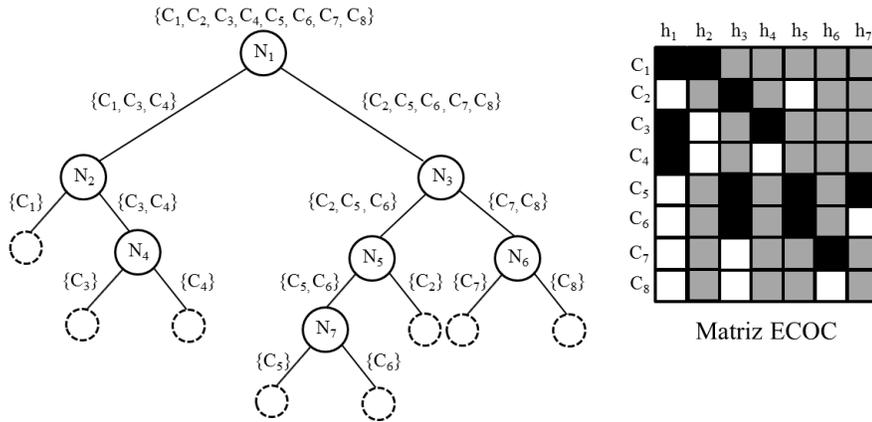


Figura 5.16: Ejemplo de conversión de árbol binario a matriz DECOC.

$$A_{m,n} = \begin{cases} 0 & \text{si } C_m \notin \wp_n \\ +1 & \text{si } C_m \in \wp_n^1 \\ -1 & \text{si } C_m \in \wp_n^2. \end{cases} \quad (5.10)$$

Cabe notar que el número de columnas coincide con el número de nodos internos y de esta manera, se asegura que la longitud de los códigos es  $N_h = M - 1$ . La figura 5.16 muestra un ejemplo de codificación de árbol binario para ocho clases. En el lado derecho de la figura se muestra la matriz de codificación ECOC, donde los cuadrados blancos representan el símbolo '+1', los negros el símbolo '-1' y los grises el símbolo '0'. Como puede observarse a modo de ejemplo, la columna 5 corresponde a la partición  $\wp_5^1 = \{C_5, C_6\}$  y  $\wp_5^2 = \{C_2\}$ . Por otra parte, desde el punto de vista de las filas de la matriz, la palabra asociada a la clase  $C_6$  es  $\mathbf{A}(6, \cdot) = \{+1, 0, -1, 0, -1, 0, +1\}$ . Como un ejemplo real, en la figura 5.17 se muestra la matriz de codificación DECOC para el conjunto de las señales azules circulares de la base de datos, teniendo en cuenta la clase de ruido que en la figura se ha representado mediante una falsa alarma.

La información mutua es un buen criterio para la partición de clases, ya que a través de la misma se calcula la cantidad de información que una variable aleatoria contiene sobre otra. En la teoría de clasificación se ha demostrado que esta medida es óptima en términos de separación de clases. La búsqueda de la mejor partición de clases entre todas las posibles puede resultar muy costosa para un número de clases elevado. Por ello, en [Pujol06] se propone el método de búsqueda flotante [Pudil94], el cual nació como un algoritmo de búsqueda secuencial para aligerar el coste computacional prohibitivo de los métodos de búsqueda en la selección de características. Los métodos de búsqueda flotante pueden describirse como algoritmos dinámicos de pasos hacia delante y pasos hacia atrás con los que se consiguen conjuntos resultantes de mejores características que los previamente evaluados. En este sentido, este método evita efectos anidados, resultando óptimo en el sentido que las mejores clases son añadidas al conjunto y las peores son descartadas.

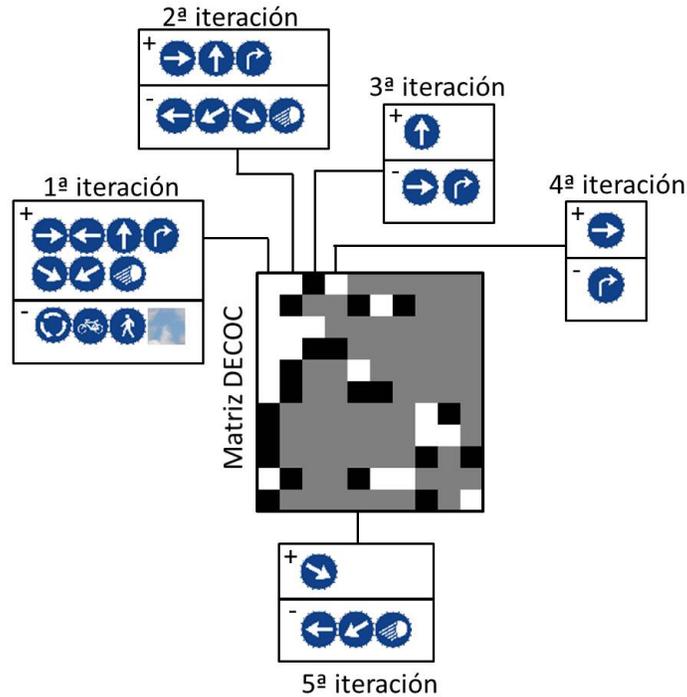


Figura 5.17: Ejemplo de matriz de codificación DECOOC para el conjunto de las señales azules circulares.

### Decodificación ECOC

Una vez definida la matriz de codificación, el proceso de decodificación obtiene el código  $\mathbf{o}(\mathbf{x})$  del vector de test  $\mathbf{x}$  y busca la palabra con código de mayor similitud. Las técnicas de decodificación binaria más empleadas son la decodificación de Hamming y la euclídea, tal como se detallan a continuación:

- Decodificación Hamming (HD). La distancia de Hamming entre el código de la clase  $\mathcal{C}_m$ , denotado por  $\mathbf{A}(m, \cdot)$ , y la realización de una muestra de test  $\mathbf{o}(\mathbf{x})$  se calcula como  $d(\mathbf{A}(m, \cdot), \mathbf{o}(\mathbf{x})) = \sum_{n=1}^{N_h} |A_{m,n} - o_n(\mathbf{x})|/2$ . Esta decodificación está basada en los principios de corrección de error en el sentido de que la tarea de aprendizaje puede ser modelada como un problema de comunicación sobre un canal con secuencias de dos o tres símbolos.
- Decodificación euclídea (ED). La distancia euclídea se define como  $d(\mathbf{A}(m, \cdot), \mathbf{o}(\mathbf{x})) = \sqrt{\sum_{n=1}^{N_h} (A_{m,n} - o_n(\mathbf{x}))^2}$ .

No obstante, en el caso de la decodificación ternaria es importante considerar el efecto del símbolo '0', a través del cual se pueden ignorar algunas clases para una cierta dicotomía. Las posiciones con símbolo '0' no deben ser significativas en comparación con las posiciones codificadas con  $\{+1, -1\}$ . Así, se define el valor base de la decodificación

como el valor introducido en la comparación de una posición codificada con  $\{+1, -1\}$  en el código  $\mathbf{o}(\mathbf{x})$  respecto a una posición que contiene el símbolo ‘0’ en la matriz de codificación. Por otra parte, las medidas de decodificación ternaria no son comparables entre palabras-código, ya que cada código trabaja en un rango distinto de valores. Por ejemplo, si consideramos dos códigos  $\mathbf{A}(m_1, \cdot)$  y  $\mathbf{A}(m_2, \cdot)$  no podemos considerar el mismo error con un fallo para una palabra  $\mathbf{A}(m_1, \cdot)$  que tenga una única posición codificada con  $\{+1, -1\}$  que para una  $\mathbf{A}(m_2, \cdot)$  con cinco posiciones codificadas. Por tanto, la diferencia en el número de posiciones codificadas es un aspecto a considerar en la decodificación ternaria. Para cuantificar este efecto, se define el rango dinámico como el rango de valores que puede generar la decodificación de cada palabra-código.

Entre las técnicas de decodificación ternarias clásicas, podemos encontrar la decodificación euclídea atenuada [Pujol06], la decodificación basada en pérdidas [Allwein02], la decodificación probabilística [Passerini04], la decodificación basada en una distribución de probabilidad  $\beta$  [Escalera10] y la decodificación basada en pérdidas ponderadas [Escalera10], a saber:

- Decodificación euclídea atenuada (AED). Esta técnica es una adaptación de la distancia euclídea, incluyendo el símbolo ‘0’. Para ello, se redefine la distancia euclídea como  $d(\mathbf{A}(m, \cdot), \mathbf{o}(\mathbf{x})) = \sqrt{\sum_{n=1}^{N_h} |A_{m,n}| (A_{m,n} - o_n(\mathbf{x}))^2}$ , donde el factor  $|A_{m,n}|$  rechaza los errores acumulados por el símbolo ‘0’ en la palabra-código de la clase  $m$ -ésima. En esta técnica se considera que la información relevante está sólo representada por las posiciones codificadas. Extendiendo esta idea discreta de la importancia del símbolo ‘0’ al caso probabilístico, se llega a la estrategia de decodificación basada en pérdidas.
- Decodificación basada en pérdidas (LB). El método de decodificación basado en pérdidas requiere que las salidas de los clasificadores binarios sea un valor que satisfaga dos especificaciones: 1) el valor de salida debe ser positivo si la muestra de test es clasificada como positiva y un valor negativo en caso contrario, y 2) la magnitud del valor debe servir como estimación de confianza en la predicción. Formalmente, dado un modelo de función de pérdidas, la medida de decodificación es la pérdida total en una muestra, con  $f_n$  como la salida predicha por el clasificador correspondiente a la columna  $n$  de la matriz  $\mathbf{A}$ . Para cada código  $m$  de la matriz  $\mathbf{A}$ , se calcula la distancia entre  $\mathbf{A}(m, \cdot)$  y  $\mathbf{x}$  con  $n \in \{1, \dots, N_h\}$  como:

$$LB(\mathbf{A}(m, \cdot), \mathbf{x}) = \sum_{n=1}^{N_h} L(A_{m,n} \cdot f_n(\mathbf{x})) \quad (5.11)$$

donde  $A_{m,n} \cdot f_n(\mathbf{x})$  corresponde al margen y  $L$  es una función de pérdidas que depende de la naturaleza del clasificador binario. Las dos funciones de pérdidas más comunes son  $L(\theta) = -\theta$  (decodificación lineal (LLB)) y  $L(\theta) = e^{-\theta}$  (decodificación

exponencial (ELB)). Nótese que esta técnica atenúa el error para el símbolo ‘0’ mientras que mantiene el peso para todas las posiciones codificadas independientemente del número de ceros de cada palabra. La decisión final asigna la muestra a la clase  $\mathcal{C}_m$  para la que se obtiene la menor puntuación. Para clasificadores binarios construidos con SVM,  $f_n(\mathbf{x})$  coincide con la salida directa de cada clasificador.

- Decodificación laplaciana (LAP). El modo más simple de evitar el efecto del símbolo ‘0’ es ignorar las posiciones codificadas con el símbolo nulo. Esta técnica contabiliza el número de coincidencias entre el código de entrada y la palabra de la matriz de codificación. Con el objetivo de que todas las palabras trabajen en el mismo rango dinámico, la medida es normalizada respecto al número total de posiciones codificadas por  $\{+1, -1\}$ . El valor de salida de la decodificación laplaciana es definido de la siguiente manera:

$$LAP(\mathbf{A}(m, \cdot), \mathbf{x}) = \frac{\alpha_m + 1}{\alpha_m + \beta_m + K}, \quad (5.12)$$

donde  $K$  es un valor entero que codifica el número de clases considerado por el clasificador (2 en el caso de un clasificador binario) y  $\alpha_m$  y  $\beta_m$  son, respectivamente, el número de coincidencias y de no coincidencias en las posiciones codificadas por  $\{+1, -1\}$ .

- Decodificación con distribución de densidad  $\beta$  ( $\beta$ -DEN). En esta técnica el símbolo ‘0’ no tiene influencia y el rango dinámico para todos los códigos está definido en el mismo intervalo  $[0, 1]$ . Para ello, ignoramos las posiciones codificadas con ‘0’ y se aplica una normalización respecto al número total de posiciones codificadas con  $\{-1, +1\}$ . Así, se utiliza como medida  $d(\mathbf{A}(m, \cdot), \mathbf{o}(\mathbf{x})) = \frac{\alpha_m}{\alpha_m + \beta_m}$ , donde  $\alpha_m$  y  $\beta_m$  representan, respectivamente, el número de aciertos y errores en las posiciones codificadas con  $\{-1, +1\}$ . Basada en esta formulación, en [Escalera10] se define un método para estimar las funciones de densidad de probabilidad entre dos códigos. El principal éxito de este método es modelar a la vez el valor de predicción y la incertidumbre asociada. Para ello, se hace uso de una distribución binomial (distribución- $\beta$ ) definida como  $\psi_m(z, \alpha_m, \beta_m) = \frac{1}{K} z^{\alpha_m} (1 - z)^{\beta_m}$ , donde  $\psi_m$  es la distribución- $\beta$  entre una realización  $\mathbf{o}(\mathbf{x})$  y la palabra de la clase  $\mathcal{C}_m$  con  $z \in [0, 1]$  y  $K$  es un valor entero que codifica el número de clases consideradas por el clasificador, que en este caso será igual a 2 al tratar con particiones binarias. Es demostrable que el valor máximo de la función  $\psi_m$  se obtiene con  $z_m = \frac{\alpha_m}{\alpha_m + \beta_m}$ . Ahora, se puede hacer uso de la integral alrededor del máximo  $z_m$  de  $\psi_m$  para obtener una medida de confianza en la predicción. En este sentido, dada una palabra de test  $\mathbf{o}(\mathbf{x})$  y el conjunto de funciones asociadas a  $\psi(z, \alpha, \beta) = [\psi_1(z, \alpha_1, \beta_1), \dots, \psi_M(z, \alpha_M, \beta_M)]$ , la realización  $\mathbf{o}(\mathbf{x})$  es asignada a la clase  $\mathcal{C}_m$  con la que se consigue el mayor valor ( $z_m - s_m$ ), satisfaciendo  $s_m$  la siguiente equivalencia:

$$s_m : \int_{z_m - s_m}^{z_m} \psi_m(z, \alpha_m, \beta_m) = r \quad (5.13)$$

donde  $r$  es un parámetro umbral fijado a  $\frac{1}{3}$ .

- Decodificación basada en pérdidas ponderadas ( $LW$ ). Esta técnica de decodificación se basa en el hecho de que todos los clasificadores no aportan la misma fiabilidad para cada clase. Así, se define una matriz de implementación,  $\mathbf{H}$ , y a cada elemento  $H_{m,n}$  de la misma se le asigna un peso que corresponde al porcentaje de muestras de la clase  $\mathcal{C}_m$  clasificadas correctamente por el clasificador  $h_n$ , considerando nulas las posiciones de la matriz correspondientes a las clases no contempladas. Finalmente, se normaliza cada fila de la matriz  $\mathbf{H}$  para obtener la matriz de pesos  $\mathbf{W}$ , donde:

$$W_{m,n} = \frac{H_{m,n}}{\sum_{j=1}^{N_h} H_{m,j}}, \quad \forall m \in [1, \dots, M], \quad \forall n \in [1, \dots, N_h]. \quad (5.14)$$

Una vez obtenida la matriz de pesos,  $\mathbf{W}$ , ésta se introduce en la decodificación basada en pérdidas. La estimación de la decodificación se obtiene por medio de un modelo de decodificación  $L(\theta)$ , donde  $\theta = A_{m,n} \cdot f^n(\mathbf{x})$  es ponderado por  $W_{m,n}$ :

$$LW(\mathbf{A}(m, \cdot), \mathbf{x}) = \sum_{n=1}^{N_h} W_{m,n} \cdot L(A_{m,n} \cdot f^n(\mathbf{x})), \quad (5.15)$$

pudiéndose considerar el uso de la función de pérdidas lineal (LLW) o la función de pérdidas exponencial (ELW).

#### 5.4.4. Metodos de evaluación

La mayor parte de los trabajos sobre comparación de clasificadores utilizan como medida comparativa el promedio de los resultados de acierto o error a través de los diferentes conjuntos de test. No obstante, de acuerdo a [Webb00], es cuestionable si las tasas de acierto/error son comparables y por tanto, si el promedio de las mismas sobre diferentes dominios es realmente útil. Si los resultados no son comparables en los diferentes conjuntos de datos, sus promedios no son significativos. De hecho, los promedios son susceptibles a las muestras anómalas ‘*outliers*’ y puede darse la situación de que un clasificador obtenga una excelente realización en un conjunto de datos que compense la mala realización global. Por el contrario, se puede dar el caso de que un fallo en un dominio prevalezca sobre los resultados en el resto. Por las razones expuestas, en esta tesis doctoral se han utilizado los dos siguientes métodos estadísticos para comparar clasificadores:

- Test de McNemar. Se trata de un test no paramétrico propuesto en [McNemar47] y que en el campo de la clasificación de patrones permite comparar las predicciones

de dos clasificadores  $h_1$  y  $h_2$ . El interés se centra en comprobar si los resultados obtenidos con los dos clasificadores son iguales o si, por el contrario, existe algún cambio significativo entre ellos. Los resultados de salida de los dos clasificadores pueden recogerse en una tabla  $2 \times 2$  (ver tabla 5.10) en la que  $n_{00}$  denota el número de muestras para las que ambos algoritmos emiten una respuesta incorrecta y  $n_{11}$  el número de muestras para las que ambos algoritmos generan aciertos. El test de McNemar se basa en contrastar la hipótesis nula  $H_0 : n_{01} = n_{10}$  en la que no hay diferencias significativas en los dos métodos de clasificación frente a la hipótesis alternativa  $H_1 : n_{01} \neq n_{10}$ . Aquí  $n_{10}$  representa el número de muestras bien clasificadas por el clasificador  $h_1$  y mal clasificadas por  $h_2$ , mientras que  $n_{01}$  representa el número de muestras bien clasificadas por el clasificador  $h_2$  y mal clasificadas por  $h_1$ . Con el objetivo de caracterizar la discrepancia entre los algoritmos, el estadístico  $X^2$  de McNemar se define a partir de los parámetros  $n_{10}$  y  $n_{01}$  como:

$$X^2 = \frac{(n_{10} - n_{01})^2}{(n_{10} + n_{01})}, \quad (5.16)$$

que, bajo la hipótesis nula  $H_0$ , presenta una distribución asintótica  $\chi^2$  con un grado de libertad.

	Clasificador $h_1$	
Clasificador $h_2$	#Aciertos	#Fallos
# Aciertos	$n_{11}$	$n_{01}$
# Fallos	$n_{10}$	$n_{00}$

Tabla 5.10: Matriz de contingencia del test de McNemar.

El test resultará significativo de cara a rechazar la hipótesis nula si para un nivel de significancia  $\alpha$  se verifica que:

$$X^2 \geq \chi_{1,\alpha}^2. \quad (5.17)$$

Al utilizar un test de significación, es común requerir un nivel de confianza del 95% para aceptar una hipótesis. Para un test con significancia  $\alpha = 0.05$ , el valor crítico del estadístico es 3.84. La hipótesis nula no es rechazada si el estadístico es menor que 3.84 y es rechazada en caso contrario.

El estadístico con la corrección de Yates [Yates34, Edwards48] para continuidad es dado por:

$$X_C^2 = \frac{(|n_{10} - n_{01}| + 1)^2}{(n_{10} + n_{01})}. \quad (5.18)$$

- Test de Wilcoxon. Se trata de un test no paramétrico propuesto en [Wilcoxon45], el cual fue usado en [Demsar06] para establecer comparaciones entre clasificadores. Supuestos dos clasificadores  $h_1$  y  $h_2$ , cuyos resultados en el experimento  $i$ -ésimo de un total de  $N$  experimentos son  $\rho_1^i$  y  $\rho_2^i$ , respectivamente, se determina la diferencia  $d_i = \rho_1^i - \rho_2^i$  y se establece una lista de diferencias ordenadas de menor a mayor según el valor absoluto de  $d_i$ , definiendo la función  $\text{ord}(d_i)$  como la posición que ocupa dentro de la lista. Si en dos experimentos  $i, j$  se produce una igualdad  $|d_i| = |d_j|$ , el orden asignado será en ambos casos la mitad de la suma de las posiciones que ocupan dichas referencias. Se define  $R^+$  como la suma de los rangos para el conjunto de datos en el que el primer clasificador funciona mejor que el segundo y  $R^-$  como la suma de rangos para los conjuntos en los que el segundo clasificador funciona mejor que el primero, del siguiente modo:

$$\begin{aligned} R^+ &= \sum_{D_i > 0} \text{ord}(D_i) + \frac{1}{2} \sum_{D_i = 0} \text{ord}(D_i) , \\ R^- &= \sum_{D_i < 0} \text{ord}(D_i) + \frac{1}{2} \sum_{D_i = 0} \text{ord}(D_i). \end{aligned} \quad (5.19)$$

A partir de  $R^+$  y  $R^-$ , se define  $T = \min(R^+, R^-)$  y se establece el estadístico siguiente:

$$z = \frac{T - \frac{1}{4}N(N+1)}{\sqrt{\frac{1}{24}N(N+1)(2N+1)}}, \quad (5.20)$$

que sigue aproximadamente una distribución normal. La hipótesis nula, según la cual no existe diferencia entre ambos clasificadores, debe ser rechazada con un nivel de confianza del 95 % o equivalentemente con un nivel de significación  $\alpha = 0.05$  si  $z < -1.96$ .

#### 5.4.5. Comparativa de clasificadores

Los experimentos correspondientes a la etapa de clasificación tienen como objetivo evaluar las diferentes técnicas de multclasificación basadas en SVM binarias. De acuerdo a los resultados de 5.3.1, se ha optado por utilizar un descriptor HOG sin aplicación de técnicas previas de preprocesado con  $N_w = 6$  y  $N_b = 10$  como compromiso de calidad y tiempo de procesado, ya que para  $N_w \geq 7$  se demostró que el tiempo de procesado incrementa notablemente. Los clasificadores base se han construido con SVM basadas en kernels RBF. Para la generación del modelo correspondiente a cada color y forma, se han considerado los mismos valores para los parámetros de regularización  $C$  y  $\gamma$  de todos los clasificadores, independientemente de la técnica de clasificación, lo cual viene justificado por el hecho de evitar un posible sobreajuste. Además, los mismos hiperparámetros son utilizados para los diferentes clasificadores binarios dentro de cada combinación de color y forma. En el proceso de selección, ocurre en algunas ocasiones que hay varias parejas de parámetros que proporcionan la misma tasa de acierto en el proceso de validación cruzada

y en estos casos se ha seguido como criterio empírico elegir entre las mismas, la pareja con valor de  $C$  más pequeño para evitar un posible sobreajuste. Los dos hiperparámetros son ajustados usando una validación cruzada sobre el problema multiclase con 5 iteraciones. Los experimentos se han realizado sobre las mismas cinco particiones de datos utilizadas en las secciones 4.4.2 y 5.3, obteniéndose los resultados como promedio de los resultados parciales de las cinco iteraciones.

#### 5.4.6. Análisis de fiabilidad y carga computacional

Las técnicas de multclasificación utilizadas son OVA-1, OVA-2, OVO, DECOC y ECOC con aleatoriedad dispersa. En la configuración de la matriz ECOC mediante codificación con aleatoriedad dispersa, el número de matrices generadas de entre las que se selecciona la matriz óptima se ha fijado en 3000, descartando previamente aquellas matrices que contienen columnas y/o filas idénticas o que poseen columnas y/o filas con todos los elementos iguales. El uso de un umbral en el método OVA-1, en el que no se construye un clasificador para el ruido, permite descartar los objetos con salida menor al valor umbral y la variación del valor de este umbral proporciona diferentes parejas de valores (TPR, FPR) con las que es posible generar una curva ROC. No obstante, la clase de ruido es tratada como una clase adicional en el resto de estrategias de multclasificación adoptadas y no tiene sentido establecer un umbral análogo para obtener a la salida un barrido de valores (TPR, FPR). Específicamente, en el método OVA-2 el objeto se asigna a la clase con mayor probabilidad de salida, en el esquema OVO el criterio es seleccionar el objeto con mayor número de votos y en las estrategias DECOC y ECOC, la clase elegida se corresponde con la que obtiene menor distancia en la decodificación. En todos estos casos se obtiene un único punto (TPR, FPR) y para trabajar con el mismo criterio en todas las técnicas de clasificación se ha fijado el umbral de salida del método OVA-1 a un valor específico, en concreto 0.5. En cualquier caso, la pareja de parámetros (TPR, FPR) obtenidos a la salida de cada clasificador se puede combinar en un único parámetro conocido como factor de mérito  $F$ , de acuerdo a la ecuación 1.9, con el objetivo de establecer una comparativa de fiabilidad.

Debido a que la clasificación supone la etapa terminal del módulo de reconocimiento, las estrategias de clasificación propuestas se han evaluado con varios conjuntos de señales con la finalidad de obtener resultados más generalistas. Para los experimentos se han utilizado las señales rojas circulares, las rojas triangulares, las circulares acromáticas, las azules circulares y las azules rectangulares. Los resultados obtenidos se muestran en la tabla 5.11, donde para cada técnica de clasificación por color y forma se incluye la información correspondiente a la fiabilidad y a la velocidad. Como medidores de fiabilidad, se aporta la media y la desviación estándar, tanto de la tasa de acierto en la clasificación de positivos (TPR) como de falsos positivos (FPR) en las 5 iteraciones correspondientes a las cinco particiones de los datos y el promedio del factor de mérito  $F$ . Como indicadores del tiempo de procesado se incluye el número de clasificadores binarios ( $N_h$ ), el número

de vectores soporte ( $N_{sv}$ ) y el tiempo total de procesado por muestra ( $T_t$ ). De manera adicional, en las columnas  $Pos_F$  y  $Pos_T$  se indica el orden que ocupa cada técnica de clasificación dentro del conjunto de las cinco propuestas bajo los criterios de fiabilidad y velocidad, respectivamente. Por tanto, la primera posición del *ranking* en las columnas  $Pos_F$  y  $Pos_T$  corresponde, respectivamente, a los métodos con mejor desempeño en base a  $F$  y mayor velocidad o equivalentemente, menor consumo de tiempo  $T_t$ . Por otra parte, cabe destacar que tanto en el esquema DECOC como en el ECOC con aleatoriedad dispersa se han probado las diferentes técnicas de decodificación descritas en la sección 5.4.3, aunque por simplicidad en la tabla 5.11 se muestran únicamente los resultados conseguidos con la mejor de las mismas.

Respecto a los resultados de fiabilidad conseguidos, se puede comprobar que, en general, las diferencias entre todas las estrategias son relativamente pequeñas y no existe una arquitectura que destaque de manera rotunda sobre el resto para todos los conjuntos de datos. De hecho, el margen de variación de la figura de mérito  $F$  es de 0.27%, 1.86%, 0.51%, 0.21% y 0.40% para los cinco conjuntos de datos. No obstante, aunque todos los esquemas de clasificación son válidos con resultados por encima del 96% en cualquier situación, cabe notar que en los tres conjuntos con más muestras (señales rojas circulares, rojas triangulares y blancas circulares) los mejores resultados, desde el punto de vista de la figura de mérito  $F$ , son conseguidos con la estrategia OVA-2, que en los tres casos queda posicionada en primer lugar, seguida de las técnicas OVO y ECOC, mientras que, en contrapartida, las técnicas OVA-1 y DECOC son las ligeramente peor paradas. En los conjuntos de las señales azules circulares y azules rectangulares, con un número más reducido de muestras, los resultados no tan claros y es difícil obtener una conclusión sobre las mejores técnicas.

El análisis comparativo desde el criterio de carga computacional arroja grandes diferencias en el consumo de tiempo, observando claramente que la técnica DECOC es, sin duda, la más rápida en todos los casos, debido al fundamento de su estrategia en la que, a medida que se desciende a través de los nodos de su árbol binario, el número de clases a discriminar se va reduciendo considerablemente. En contrapartida, la técnica OVO es la más lenta debido al alto número de clasificadores que exige su arquitectura, lo cual causa, que a medida que aumenta el número de clases se dispara el número de clasificadores. Así, por ejemplo, el número de clasificadores para el conjunto de señales rojas circulares con  $M = 58 + 1$ , incluyendo el ruido, es 1711 frente al resto de técnicas que no sobrepasan los 59 clasificadores y la sobrecarga de tiempo percentual respecto a DECOC es de un 222.31%. El altísimo tiempo de cómputo de la técnica OVO, respecto al del resto de métodos, nos lleva a descartarla. Es interesante hacer notar que la técnica que, aparentemente, proporciona mayor fiabilidad, OVA-2, demanda tiempos de procesado comparables a la técnica más óptima desde el punto de vista computacional, DECOC. Aquí, la sobrecarga de tiempo percentual de OVA-2 respecto a DECOC es 9.46%, 12.67%, 11.80%, 20.56% y 9.60%. Obviamente, el tiempo de carga de la estrategia OVA-1 es ligeramente inferior al de OVA-2 debido a que en el primer caso no se construye un clasificador explícito para

(a) Señales rojas circulares ( $M = 58 + 1$ )

	$TPR(\%)$	$FPR(\%)$	$F(\%)$	$Pos_F$	$N_h$	$N_{sv}$	$T_t(\text{ms})$	$Pos_T$
OVA-1	98.57±0.16	0.87±0.09	98.85	4	58	4692	81.72	2
OVA-2	98.90±0.15	0.49±0.07	99.12	1	59	4854	83.77	3
OVO	98.37±0.23	0.34±0.07	99.00	3	1711	4886	246.07	5
DECOC	98.02±0.21	0.30±0.06	98.85	4	58	4026	76.53	1
ECOC	98.60±0.21	0.51±0.17	99.03	2	59	5896	109.89	4

(b) Señales rojas triangulares ( $M = 45 + 1$ )

	$TPR(\%)$	$FPR(\%)$	$F(\%)$	$Pos_F$	$N_h$	$N_{sv}$	$T_t(\text{ms})$	$Pos_T$
OVA-1	94.73±0.90	0.93±0.08	96.85	5	45	2689	48.17	2
OVA-2	98.38±0.28	0.96±0.18	98.71	1	46	2791	49.70	3
OVO	98.26±0.23	0.96±0.25	98.64	2	1035	3227	108.06	5
DECOC	97.19±0.24	1.44±0.25	97.86	4	45	2371	44.11	1
ECOC	98.13±0.16	0.90±0.25	98.61	3	55	4118	70.50	4

(c) Señales blancas circulares ( $M = 55 + 1$ )

	$TPR(\%)$	$FPR(\%)$	$F(\%)$	$Pos_F$	$N_h$	$N_{sv}$	$T_t(\text{ms})$	$Pos_T$
OVA-1	97.77±0.62	0.88±0.11	98.44	4	55	1532	30.21	2
OVA-2	98.86±0.12	0.96±0.31	98.95	1	56	1593	32.11	3
OVO	98.52±0.10	0.60±0.18	98.95	1	1540	1869	90.23	5
DECOC	98.25±0.12	1.19±0.24	98.53	3	55	1231	28.72	1
ECOC	98.57±0.06	0.81±0.26	98.88	2	58	2170	42.27	4

(d) Señales azules circulares ( $M = 10 + 1$ )

	$TPR(\%)$	$FPR(\%)$	$F(\%)$	$Pos_F$	$N_h$	$N_{sv}$	$T_t(\text{ms})$	$Pos_T$
OVA-1	99.33±0.39	0.85±0.29	99.24	4	10	497	12.82	3
OVA-2	99.16±0.35	0.65±0.61	99.25	3	11	514	13.13	4
OVO	98.97±0.34	0.65±0.61	99.15	5	55	467	12.59	2
DECOC	99.05±0.37	0.32±0.32	99.36	1	10	425	10.89	1
ECOC	99.27±0.27	0.65±0.61	99.31	2	35	508	13.14	5

(e) Señales azules rectangulares ( $M = 39 + 1$ )

	$TPR(\%)$	$FPR(\%)$	$F(\%)$	$Pos_F$	$N_h$	$N_{sv}$	$T_t(\text{ms})$	$Pos_T$
OVA-1	97.89±0.24	0.80±0.12	98.54	5	39	832	16.79	2
OVA-2	98.71±0.18	1.16±0.55	98.77	3	40	903	17.68	3
OVO	98.17±0.24	0.53±0.30	98.81	2	780	1089	36.29	5
DECOC	98.44±0.21	0.98±0.51	98.72	4	39	741	16.13	1
ECOC	98.56±0.27	0.67±0.38	98.94	1	53	1237	22.62	4

Tabla 5.11: Resultados de clasificación con diferentes arquitecturas basadas en SVM.

el ruido. Como conclusión, el doble compromiso de fiabilidad y eficiencia apunta en la dirección de que la técnica OVA-2 es la que proporciona, aparentemente, mejores resultados en líneas generales.

A la hora de considerar el número de vectores soporte, hay que tener en cuenta que un mismo vector de entrenamiento puede resultar vector soporte de diferentes clasificadores binarios en un sistema de multclasificación. Por este motivo, en la tabla 5.11 se indica el promedio del número de vectores soporte ‘únicos’ de los cinco experimentos, siendo redondeado este valor al entero más próximo. Como el tiempo de testeo de las SVM es dominado por la evaluación de los kernels, para reducir la carga computacional se calculan y almacenan en primer lugar todos los  $K(\mathbf{x}_s, \mathbf{x})$ , donde  $\mathbf{x}_s$  es cualquier vector soporte único y  $\mathbf{x}$  es el vector de test. De esta manera, los kernels ‘únicos’ son calculados una única vez, a pesar de que pueden ser reutilizados por varios clasificadores. El almacenamiento se realiza en una tabla de búsqueda y el índice del vector soporte permite acceder al valor del kernel. En la tabla 5.11 se puede ver que la técnica DECOC es la que requiere menor número de vectores soporte en todos los conjuntos de datos, mientras que por el contrario, la técnica ECOC con aleatoriedad dispersa destaca por ser la que incluye mayor número de vectores soporte en todos los casos, excepto en el conjunto de las señales azules circulares. Esto denota que las fronteras de decisión de los clasificadores binarios no comparten muchos vectores soporte entre ellos.

#### 5.4.7. Análisis de técnicas de decodificación

En las estrategias de multclasificación DECOC y ECOC la clase elegida se corresponde con la que obtiene menor distancia en la decodificación. Los resultados mostrados en la tabla 5.11 son los obtenidos con la técnica de decodificación que aporta mejor factor de mérito. No obstante, los resultados para las dos técnicas de decodificación empleadas, DECOC y ECOC con aleatoriedad dispersa, se recogen de manera detallada en la tabla 5.12 para las ocho técnicas de decodificación utilizadas: distancia euclídea (ED), laplaciana (LAP), distancia euclídea atenuada (AED), decodificación basada en pérdidas con evolución lineal (LLB) y evolución exponencial (ELB), decodificación con distribución de densidad  $\beta$  ( $\beta$ -DEN), decodificación basada en pérdidas ponderadas con evolución lineal (LLW) y evolución exponencial (ELW). Como se puede comprobar en la tabla 5.12, donde la mejor estrategia de decodificación se ha marcado en negrita para cada conjunto de datos, la decodificación exponencial basada en pérdidas ponderadas (ELW) es la técnica mejor librada. ELW es la mejor opción para la arquitectura DECOC en todos los conjuntos de datos y también en dos de las cinco combinaciones de la arquitectura ECOC dispersa, con un resultado muy próximo a la mejor técnica de decodificación en las otras tres combinaciones. La justificación de los mejores resultados de ELW respecto a otras técnicas viene dada porque en esta técnica los símbolos nulos de la matriz de codificación no son considerados, al asignar un peso nulo a las posiciones con el símbolo ‘0’ en la

matriz de codificación, y los valores de decodificación de todos los códigos se extienden en el mismo rango dinámico de salida  $[\frac{1}{e}, e]$ , tal como se justifica en [Escalera10].

(a) Arquitectura DECOC

	ED	LAP	AED	LLB	ELB	$\beta$ -DEN	LLW	ELW
Rojas circulares	98.70	98.70	98.70	97.93	98.50	98.70	98.57	<b>98.85</b>
Rojas triangulares	96.85	96.85	96.85	96.92	97.48	96.85	97.34	<b>97.86</b>
Blancas circulares	97.67	97.67	97.67	97.35	98.11	97.67	97.96	<b>98.53</b>
Azules circulares	99.18	99.18	99.18	98.88	99.35	99.18	99.31	<b>99.36</b>
Azules rectangulares	98.17	98.17	98.17	96.99	98.01	98.17	97.93	<b>98.72</b>

(b) Arquitectura ECOC con aleatoriedad dispersa

	ED	LAP	AED	LLB	ELB	$\beta$ -DEN	LLW	ELW
Rojas circulares	99.02	<b>99.03</b>	98.99	98.70	98.96	<b>99.03</b>	98.90	99.00
Rojas triangulares	98.47	98.50	98.28	97.17	98.40	98.49	98.27	<b>98.61</b>
Blancas circulares	98.79	98.77	98.73	98.27	98.78	98.77	98.68	<b>98.87</b>
Azules circulares	<b>99.31</b>	<b>99.31</b>	99.28	98.70	99.06	<b>99.31</b>	99.09	99.20
Azules rectangulares	98.92	98.88	<b>98.94</b>	97.56	98.81	98.90	98.44	98.90

Tabla 5.12: Resultados de clasificación en función de las técnicas de decodificación.

### 5.4.8. Análisis de significancia estadística

El análisis comparativo entre técnicas de clasificación bajo criterios de fiabilidad se realizó en la sección 5.4.6 mediante comparación directa del factor de mérito a partir de los parámetros TPR y FPR. No obstante, para confirmar la validez de estos resultados se ha aplicado una comparación pareada entre clasificadores usando dos tests de significancia estadística: el test de McNemar y el test de Wilcoxon bajo un intervalo de confianza del 95%. En adelante, cuando se use el término diferencia estadísticamente significativa, o simplemente diferencia significativa, nos referiremos a que la diferencia es estadísticamente significativa de acuerdo a los tests utilizados. En la tabla 5.13 se muestra la comparación pareada entre clasificadores, indicándose el número de victorias, empates y derrotas en este orden a partir de la comparación directa del promedio de la tasa de acierto de todas las clases (*Acc.Mean-Accuracy Mean*) y de la significancia estadística a partir de los tests de McNemar y Wilcoxon. Cada celda contiene, respectivamente, el número de victorias, empates y derrotas estadísticamente significativas entre el método de su correspondiente fila y el método de su columna para los 25 experimentos realizados en cada caso (5 particiones de datos  $\times$  5 conjuntos de datos). En la tabla 5.13 puede observarse claramente como OVA-2 supera significativamente al resto de clasificadores frecuentemente. Así, OVA-2 mejora muy frecuentemente a DECOC (16-9-0) y a OVA-1 (18-7-0) de acuerdo al test de McNemar, y con menor frecuencia a ECOC (6-19-0) y OVO (8-17-0), sin darse en

ningún caso una derrota significativa. Además, los resultados también dejan claro que la técnica ECOC supera muy significativamente a DECOC y OVA-1 y muestra cierta mejora significativa respecto a OVO, aunque sale derrotada frente a OVA-2. De este modo, se puede concluir que OVA-2 y ECOC son las dos técnicas de clasificación claramente superiores de las cinco propuestas desde el punto de vista de fiabilidad.

		DECOC	ECOC	OVO	OVA-2
OVA-1	Acc.Mean	10-0-15	1-0-24	8-0-17	3-0-22
	McNemar Test	5-12-8	0-12-13	1-13-11	0-7-18
	Wilcoxon Test	2-16-7	0-12-13	0-16-9	0-8-17
OVA-2	Acc.Mean	20-0-5	15-0-10	22-0-3	—
	McNemar Test	16-9-0	6-19-0	8-17-0	—
	Wilcoxon Test	15-10-0	5-20-0	7-18-0	—
OVO	Acc.Mean	17-0-8	9-0-16	—	—
	McNemar Test	11-14-0	0-20-5	—	—
	Wilcoxon Test	8-17-0	0-20-5	—	—
ECOC	Acc.Mean	22-0-3	—	—	—
	McNemar Test	14-11-0	—	—	—
	Wilcoxon Test	10-15-0	—	—	—

Tabla 5.13: Tests estadísticos sobre los conjuntos de datos. Para cada pareja de clasificadores enfrentados se aporta el número de victorias (W-Win), empates (T-Ties) y derrotas (L-Loss) del clasificador de la fila frente al clasificador de la columna basándose en el promedio de la tasa de acierto y en la significancia estadística a partir de los tests de McNemar y Wilcoxon.

## 5.5. Resumen

En este capítulo se ha realizado un estudio comparativo entre diferentes técnicas para la implementación del módulo de reconocimiento de señalización. Para ello, se ha estructurado el módulo en tres etapas correspondientes al preprocesado, a la extracción del descriptor y al clasificador. Los descriptores propuestos están basados en la concatenación de luminancia de píxeles en el dominio espacial, histogramas de gradientes orientados (HOG), imagen ‘spin’ (SPIN), transformada invariante a rotación (RIFT) y patrones locales binarios (LBP). Tras el análisis paramétrico, los resultados obtenidos, confirman la gran potencia discriminatoria del descriptor HOG, aún cuando se utiliza un número reducido de atributos.

La etapa de clasificación se ha implementado mediante arquitecturas basadas en clasificadores binarios. En concreto, se han utilizado las estrategias ‘uno contra todos’, ‘uno

contra uno' y dos técnicas basadas en códigos correctores de error, demostrándose que la técnica 'uno contra todos' es significativamente mejor que el resto de propuestas desde el doble compromiso de fiabilidad y velocidad. La emergente técnica DECOC, basada en la búsqueda de árboles binarios que generan la mejor partición de clases, es la más óptima desde el punto de vista computacional. Para el análisis comparativo de los diferentes métodos de clasificación se ha realizado un promediado de los resultados, contrastándose con un estudio estadístico.



# Capítulo 6

## Reconocimiento de señales basado en mapas de pictogramas

En esta tesis doctoral se propone un algoritmo con el objetivo de aligerar el ‘cuello de botella’ que supone el reconocimiento de señales viales en sistemas completos de detección y reconocimiento de señales de tráfico. La motivación del método propuesto viene dada por intentar reducir la carga computacional del módulo de reconocimiento a partir de la búsqueda de zonas de la señal donde se ubica el mensaje informativo. Las señales de tráfico son diseñadas, generalmente, mediante un pictograma cuyo mensaje cubre parte de la superficie de la señal sobre un fondo homogéneo. El objetivo del método se basa en buscar la información de las zonas relevantes sobre el fondo y adaptar las características de los descriptores de forma dinámica de acuerdo a la cantidad de información detectada en cada zona.

La distribución espacial del mensaje informativo es utilizada en el algoritmo para dos tareas: 1) agrupar las clases con distribuciones espaciales similares y 2) definir el proceso de extracción del descriptor de reconocimiento para cada zona. De esta manera, se consigue limitar el número de clases implicadas en el proceso de reconocimiento al centrarse en clusters, en vez de sobre todo el conjunto de clases, y por otra parte, se reduce la dimensión de los vectores, al codificar con menos características las zonas de la imagen con escasa información. Ambas ventajas permiten mejorar notablemente la eficiencia del sistema.

### 6.1. Mapa del pictograma

En esta sección se presenta un algoritmo para evaluar la distribución espacial de la información en las señales con el objetivo de identificar las zonas donde se extiende el pictograma.

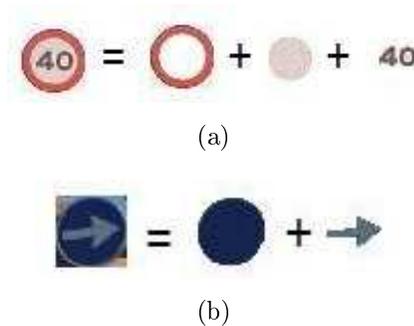


Figura 6.1: Modelos de señales como contribución de partes. (a) Modelo de tres partes correspondientes a orla, fondo y pictograma. (b) Modelo de dos partes: fondo y pictograma.

### 6.1.1. Proceso de extracción

El mensaje informativo de las señales de tráfico se representa mediante un pictograma que consta de uno o varios iconos y por lo general, no cubre la superficie completa de la señal, tal como puede verse en el Apéndice C. La mayoría de las señales de advertencia y prohibición pueden ser consideradas como una contribución de tres partes correspondientes a la orla externa de color rojo, el fondo y el pictograma, mientras que en el caso de las señales de información y obligación, sólo aparecen el fondo y el pictograma. La descomposición por partes de una señal es representada en la figura 6.1 con dos ejemplos. En cualquier caso, la orla externa de las señales de advertencia y prohibición no aporta información en el reconocimiento del pictograma y es ignorada mediante la aplicación de una máscara.

Para determinar la distribución espacial de los pictogramas, el proceso de extracción consta de tres pasos:

1. Obtención de la magnitud del gradiente del blob en escala de gris a partir de los gradientes direccionales en los ejes vertical  $G_v(x, y)$  y horizontal  $G_h(x, y)$  del plano imagen como  $G_{mag} = \sqrt{G_v^2(x, y) + G_h^2(x, y)}$ .
2. División del blob en una rejilla uniforme de celdas no solapadas con  $N_r$  celdas en dirección vertical y  $N_c$  celdas en dirección horizontal.
3. Cálculo de la contribución de cada celda como el sumatorio de la magnitud del gradiente en todos sus píxeles.

De esta manera, el blob de entrada es dividido en  $N_r \times N_c$  celdas no solapadas y el algoritmo calcula la suma del gradiente en cada celda. El descriptor generado con esta información es denominado Mapa del Ideograma (IM-*Ideogram Map*) y el proceso de extracción se muestra en la figura 6.2 con un ejemplo para una señal roja circular, donde  $N_r$  y  $N_c$  fueron fijados a 3. En la representación del IM (figura 6.2(f)), la magnitud del

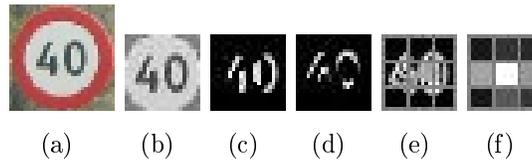


Figura 6.2: Ejemplo de extracción del IM para una señal roja circular con una rejilla de  $3 \times 3$  celdas. (a) Blob original. (b) Blob en escala de gris después de eliminar la orla externa mediante la máscara. (c) Gradiente vertical del blob  $G_v(x, y)$ . (d) Gradiente horizontal del blob  $G_h(x, y)$ . (e) Magnitud del gradiente  $G_{mag}(x, y)$ . (f) Mapa del pictograma  $IM$ .

gradiente normalizada en cada celda es representada con niveles de gris, de tal modo que las celdas más oscuras identifican aquellas de baja magnitud. La concatenación de estos valores proporciona un descriptor unidimensional  $\mathbf{z}$  dado por:

$$\mathbf{z} = [z_{1,1}, z_{1,2}, \dots, z_{1,N_c}, z_{2,1}, z_{2,2}, \dots, z_{2,N_c}, \dots, z_{N_r,1}, z_{N_r,2}, \dots, z_{N_r,N_c}] \quad (6.1)$$

con una longitud de  $N_r \times N_c$  en el que cada componente  $z_{r,c}$  del vector se calcula como:

$$z_{r,c} = \sum_{(x,y) \in R_{r,c}} G_{mag}(x, y), \quad (6.2)$$

denotando a través de  $R_{r,c}$  la región espacial de cada celda en el plano imagen con  $1 \leq i \leq N_r$  y  $1 \leq j \leq N_c$ . Para muestras de dimensiones  $W \times H$ , el tamaño de la celda es igual a  $w \times h$  con  $w = \lfloor W/N_c \rfloor$  y  $h = \lfloor H/N_r \rfloor$ . En la figura 6.3 se muestran los IMs asociados a las clases rojas circulares y azules rectangulares de la base de datos con un número de muestras significativas, habiéndose obtenido el mapa de cada clase como el promedio de los IMs sobre las muestras de entrenamiento pertenecientes a esa clase.

### 6.1.2. Agrupamiento de clases

Las señales de tráfico son diseñadas de modo que, en la mayor parte de los casos, el pictograma aparece centrado en la superficie de las mismas. Como consecuencia, la distribución espacial de los iconos presenta alta similitud entre las clases, tal como se puede observar en la figura 6.3. Debido a esta circunstancia, los IMs se utilizan como criterio de similitud para agrupar las clases. Considerando un conjunto de  $M$  clases denotado como  $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_M\}$ , el proceso de clustering produce una partición de  $\mathcal{C}$  en  $N$  conjuntos disjuntos  $\{S_1, S_2, \dots, S_N\}$ , con  $\bigcup_i^N S_i = \mathcal{C}$  y  $S_i \cap S_j = \emptyset$  para todo  $i \neq j$ . A la hora de realizar los agrupamientos de clases, en esta investigación se han obtenido los IMs de cada clase como promedio de los mapas de las muestras reales de entrenamiento, en vez de obtenerlo a partir de prototipos sintéticos. La justificación viene dada porque en este proceso es interesante incluir muestras reales afectadas por las siguientes particularidades:

- Los pictogramas definidos para cada clase pueden presentar diferentes tipografías.



- El sistema TSDRS empleado no consigue siempre una segmentación perfecta de la señal y ésto implica un desplazamiento de los píxeles en el dominio espacial.
- El módulo de clasificación no consigue siempre fijar las señales en la orientación de referencia, lo que puede causar que los blobs presenten ligeras rotaciones o deformaciones proyectivas.

La similitud entre dos IMs ( $\mathbf{z}_i, \mathbf{z}_j$ ) de longitud  $N_r \times N_c$  en el proceso de agrupamiento es evaluada mediante la distancia euclídea. Como un ejemplo, la figura 6.4 representa los agrupamientos resultantes para las señales rojas circulares utilizando el algoritmo  $K - Means$  cuando el número de agrupamientos  $N$  es fijado a 4. El algoritmo de clustering establece  $N$  centroides para los clusters  $\{\mathbf{u}_i\}_{i=1}^N$ , a los que nos referiremos como mapas representativos de los clusters (*CM-Cluster map*) y en la figura 6.4 se representa el correspondiente CM debajo de cada cluster. Esto genera un alfabeto de mapas:

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N], \quad (6.3)$$

en el que  $\mathbf{U} \in \mathbb{R}^{N \times N_r \times N_c}$ . Es importante destacar que el proceso de agrupamiento se realiza *off-line*.

## 6.2. Descripción del algoritmo

El método propuesto simplifica la tarea de reconocimiento introduciendo un agrupamiento previo que aglutina clases con similar distribución espacial del pictograma dentro de una misma combinación de color y forma. El algoritmo, que consta de dos etapas, se ilustra de manera gráfica en la figura 6.5. En la primera etapa, el algoritmo calcula la probabilidad de que el descriptor IM de la señal a testear pertenezca a cada uno de los clusters y selecciona aquellos clusters con más alta probabilidad. En la segunda etapa, se realiza la categorización del pictograma dentro de los clusters seleccionados.

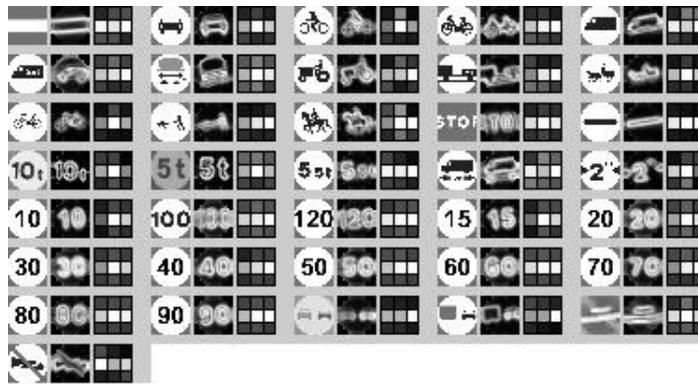
La reducción computacional se consigue en este algoritmo mediante dos aportaciones: 1) simplificación del problema de reconocimiento mediante el agrupamiento, ya que el número de clases dentro de un cluster es significativamente menor que el número de clases totales, y 2) reducción de la dimensión del descriptor, ya que éste se configura a partir de la cantidad de información en cada celda. Aunque el algoritmo planteado en esta tesis doctoral se ha aplicado a la identificación de señales de tráfico, también podría utilizarse para otros problemas de multi-clasificación de imágenes.

### 6.2.1. Selección de clusters

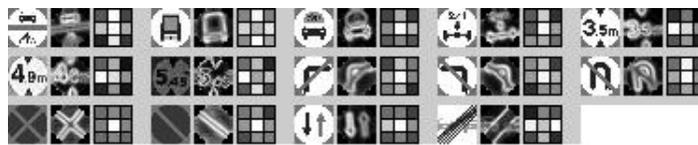
Como puede verse en la figura 6.5, la selección de clusters representa la primera etapa del método. El algoritmo selecciona, en base al IM, aquellos agrupamientos con mayor



(a)



(b)



(c)



(d)

Figura 6.4: Ejemplo de agrupamientos para señales rojas circulares con  $N = 4$ . (a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4.

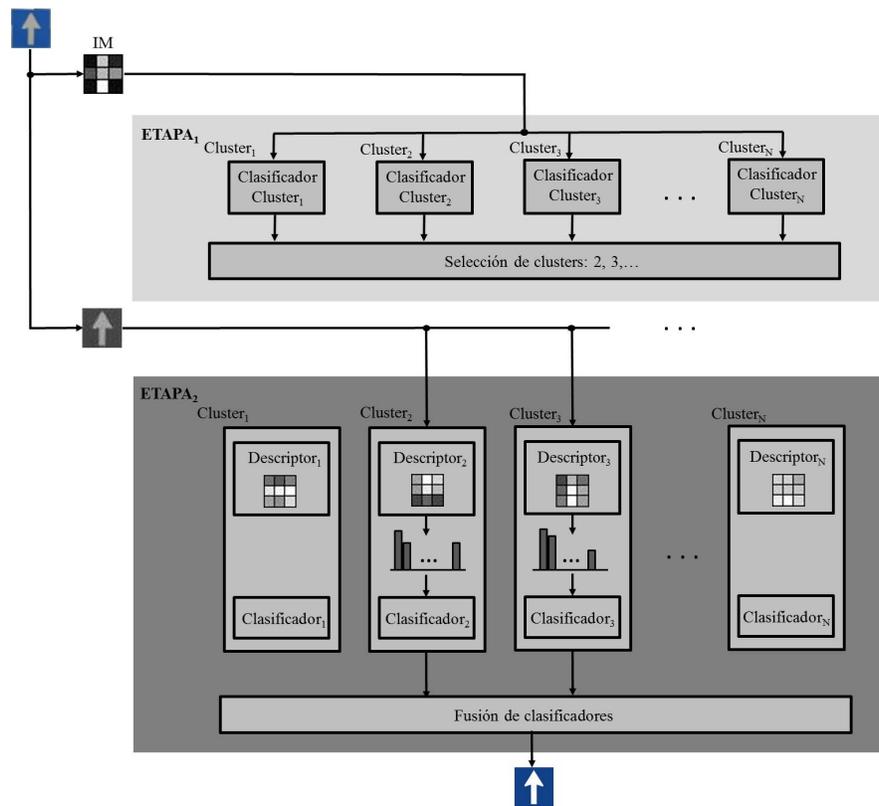


Figura 6.5: Diagrama conceptual del algoritmo, el cual construye el IM de cada muestra de entrada a partir de la magnitud del gradiente. El algoritmo consta de dos etapas: 1) selección de los clusters con mayor probabilidad y 2) identificación del ideograma a partir de los clusters seleccionados.

probabilidad de incluir la verdadera clase de la muestra a testear. Aunque para la selección de clusters podrían utilizarse diferentes clasificadores, en esta tesis se ha empleado una arquitectura basada en SVM debido a sus buenas propiedades de clasificación en la que la representación unidimensional del IM de la muestra de entrada es un vector,  $\mathbf{z}_i$ , que alimenta las entradas de las SVM. Mediante una estrategia OVA, el proceso de entrenamiento de las SVM encargadas de la selección de clusters se ha realizado con los IMs de muestras reales procedentes de nuestra base de datos. El conjunto positivo de entrenamiento del cluster  $n$ -ésimo,  $S_n$ , contiene los IMs correspondientes al conjunto de las  $M_n^+$  clases agrupadas, que vienen dadas por  $\mathcal{C}_n^+ = \{\mathcal{C}_{n,1}^+, \mathcal{C}_{n,2}^+, \dots, \mathcal{C}_{n,M_n^+}^+\}$ . Por su parte, el conjunto de entrenamiento negativo incluye los IMs de muestras correspondientes a las  $M_n^-$  clases contenidas en el resto de clusters, es decir  $\mathcal{C}_n^- = \{\mathcal{C}_{n,1}^-, \mathcal{C}_{n,2}^-, \dots, \mathcal{C}_{n,M_n^-}^-\}$ .

La precisión en la selección de clusters fue testeada mediante SVM lineales y SVM con kernels polinómicos y de base radial (RBF), obteniéndose los mejores resultados con el kernel RBF. Así, el kernel RBF a evaluar viene dado por:

$$K(\mathbf{z}_i, \mathbf{z}_s) = \exp\left(\frac{-\|\mathbf{z}_i - \mathbf{z}_s\|^2}{2\sigma^2}\right) \quad (6.4)$$

donde  $\|\mathbf{z}_i - \mathbf{z}_s\|^2$  puede ser identificada como la distancia euclídea entre el mapa IM de la muestra de entrada a testear,  $\mathbf{z}_i$ , y un vector soporte,  $\mathbf{z}_s$ . De esta manera, la selección de clusters implica un problema de multi-clasificación cuando el número de clusters en la partición es mayor que 2. Para la estrategia OVA utilizada en este algoritmo, se implementa un clasificador  $f_n$  por cada cluster con  $n = 1, \dots, N$ . Para que las salidas de los clasificadores de los clusters sean comparables, se realiza una calibración de las SVM ([Wu04] y [Lin07]), ajustando una función sigmoideal para cada clasificador. En el proceso de test, la probabilidad de salida de cada cluster viene dada por:

$$P(S_n|\mathbf{z}_i) = \frac{1}{1 + \exp(A_n f_n + B_n)}, \quad (6.5)$$

donde los parámetros  $A_n$  y  $B_n$  ajustan la sigmoide del modelo de cada cluster al conjunto de entrenamiento y  $f_n = f_n(\mathbf{z}_i)$  es el valor de salida de la SVM  $n$ -ésima para el descriptor  $\mathbf{z}_i$  del mapa de entrada. Finalmente, el cluster asignado vendrá definido por:

$$\hat{S} = \underset{n}{\operatorname{argmax}} \{P(S_n|\mathbf{z}_i)\}. \quad (6.6)$$

La figura 6.6 recoge un ejemplo que muestra las probabilidades a la salida de las SVM para una señal circular de entrada con  $N = 4$  clusters y una rejilla de  $3 \times 3$  celdas. En la parte superior de la figura se ha incluido la muestra original en color, la muestra en nivel de gris después de eliminar la orla externa, la magnitud del gradiente y el correspondiente mapa de celdas (IM). Como se puede observar, la mayor probabilidad se obtiene para el cluster de índice 2, el cual incluye la señal ‘Prohibido adelantar’(ver figura 6.4(b)). No obstante, los problemas mencionados en el apartado 6.1.2 pueden distorsionar los IMs y

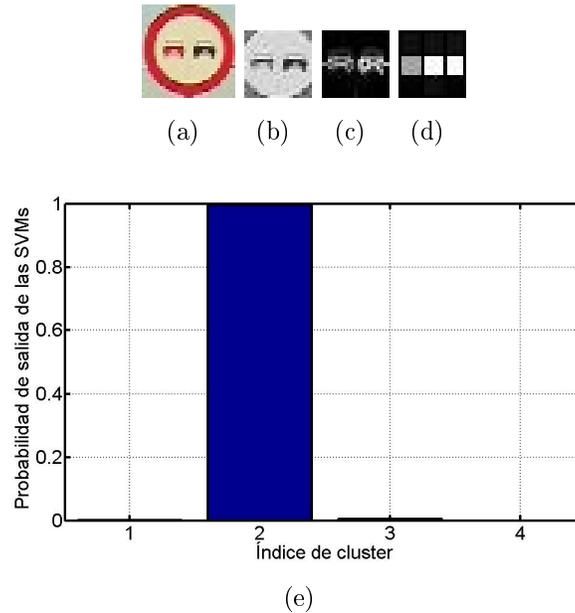


Figura 6.6: Ejemplo de correcta asignación de cluster. (a) Muestra original segmentada. (b) Muestra en escala de gris. (c) Muestra después de eliminar la orla. (d) Máscara de bordes. (e) Mapa del ideograma (IM). (f) Estimaciones de probabilidad a la salida de las SVM.

en algunos casos puntuales, la selección de clusters a partir de las probabilidades de salida no es tan clara como en el ejemplo anterior. Así, en las figuras 6.7 y 6.8 se muestran dos ejemplos de error en la asignación de clusters por diferentes motivos. En el primero de ellos, las sombras proyectadas en la señal provocan que el mapa sea asignado al segundo cluster, cuando en realidad debería ser asignada al tercero (ver figura 6.4). Otra asignación errónea de cluster se da en la figura 6.8 en la que una situación de contraluz impide la correcta visualización de uno de los dígitos en la señal de prohibición de velocidad y la muestra se asigna al tercer cluster, en vez de al segundo.

Para intentar minimizar el efecto de una incorrecta asignación de cluster, el algoritmo permite la posibilidad de activar para la muestra de entrada más de un cluster. De esta manera, el sistema asigna la muestra de entrada a los clusters con más alta probabilidad y en la etapa 2 del algoritmo (ver figura 6.5) sólo se alimentarán los correspondientes clasificadores de los clusters seleccionados. La figura 6.9 muestra la tasa de acierto en la asignación de clusters para el conjunto de test de las señales rojas circulares y azules rectangulares dependiendo del número de clusters seleccionados,  $n_c$ , cuando  $N = 4$  y la estructura de la rejilla es de  $3 \times 3$  celdas. Como se puede comprobar, la tasa de acierto es de 99.38 % y 99.62 %, respectivamente, para las rojas circulares y las azules rectangulares cuando se elige únicamente el cluster con mayor probabilidad ( $n_c = 1$ ). Según aumenta el valor del parámetro  $n_c$ , también lo hace la tasa de acierto hasta alcanzar el 100 %

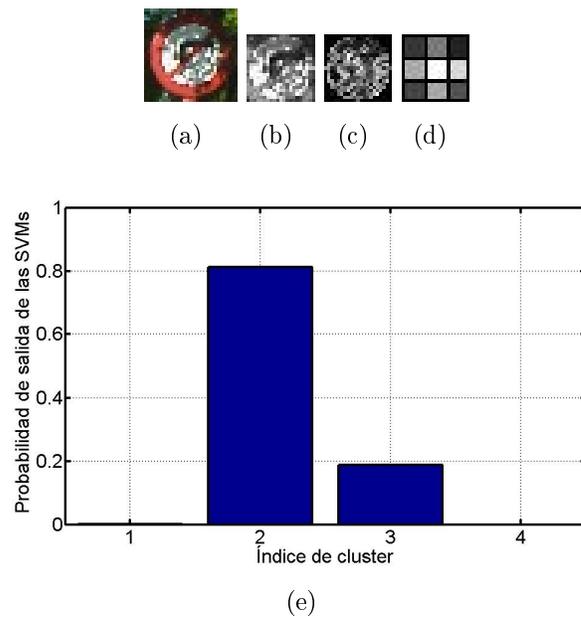


Figura 6.7: Ejemplo de una problemática asignación de cluster. (a) Muestra original segmentada. (b) Muestra en escala de gris. (c) Muestra después de eliminar la orla. (d) Máscara de bordes. (e) Mapa del ideograma. (f) Estimaciones de probabilidad a la salida de las SVM.

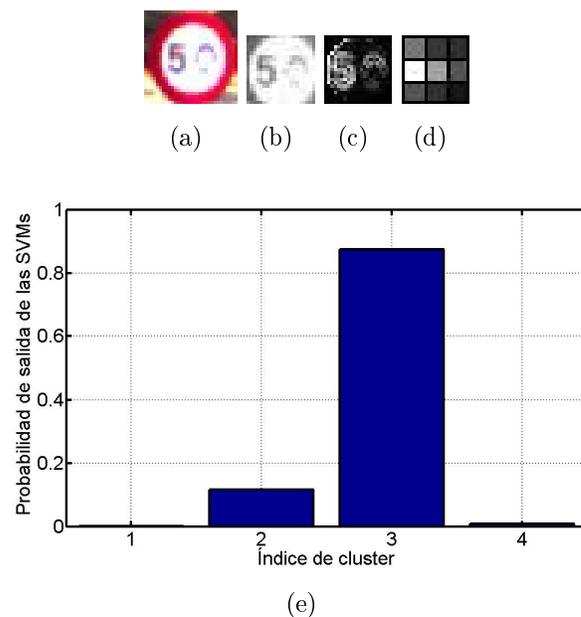


Figura 6.8: Ejemplo de una problemática asignación de cluster. (a) Muestra original segmentada. (b) Muestra en escala de gris. (c) Muestra después de eliminar la orla. (d) Máscara de bordes. (e) Mapa del ideograma. (f) Estimaciones de probabilidad a la salida de las SVM.

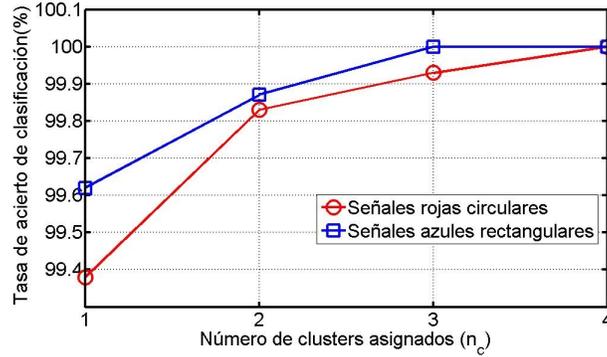


Figura 6.9: Tasa de acierto en la asignación de 4 clusters y rejilla de  $3 \times 3$  celdas.

cuando  $n_c = N$ . En la figura 6.9 se puede comprobar cómo en el caso de las señales azules rectangulares, la tasa de acierto alcanza el 100 % con  $n_c = N - 1$ .

### 6.2.2. Reconocimiento del ideograma

La segunda etapa del algoritmo (ver figura 6.5) consiste en categorizar cada muestra de entrada a partir de un descriptor característico. El proceso de reconocimiento requiere describir las muestras a través de un descriptor y clasificarlas entre las clases potenciales. Para una muestra de entrada con mapa de distribución  $\mathbf{z}_i$  y en el caso de trabajar con  $n_c = 1$ , la muestra se asigna a un cluster  $\hat{S}_n$ , cuyas clases potenciales a asignar vienen definidas por  $\mathcal{C}_n^+ = \{\mathcal{C}_{n,1}^+, \mathcal{C}_{n,2}^+, \dots, \mathcal{C}_{n,M_n}^+\}$ .

En el método propuesto, el proceso de extracción del descriptor es realizado en cada cluster independientemente. De hecho, dentro de cada cluster se construye un descriptor  $\mathbf{x}$  de acuerdo a la información del correspondiente mapa de cluster  $\mathbf{u}_n$  sobre la distribución del pictograma, lo cual significa que la dimensión de los descriptores varía entre clusters. El propósito es que la longitud del descriptor en la celda  $R_{r,c}$ , denotado por  $\mathbf{x}_{r,c}$ , dependa del valor del mapa de cluster en esa misma celda,  $u(r,c)$ . En esta investigación se ha utilizado un descriptor HOG de longitud variable, sin solapamiento de celdas ni agrupamiento de bloques, cuyo número de intervalos en cada celda viene dado por:

$$N_b^{r,c} = \lfloor u(r,c)N_b^{max} \rfloor, \quad (6.7)$$

donde  $N_b^{max}$  es el número máximo de intervalos posibles asignados para cada celda. Como el mapa del cluster es normalizado, se cumple que  $u(r,c) \leq 1$  y cuando  $u(r,c) = 1$ , entonces  $N_b^{r,c} = N_b^{max}$ . Después de extraer los descriptores de todas las celdas, se obtiene el descriptor total mediante concatenación que puede expresarse como:

$$\mathbf{x} = \bigcup_{r=1}^{N_r} \left\{ \bigcup_{c=1}^{N_c} \mathbf{x}_{r,c} \right\}, \quad (6.8)$$

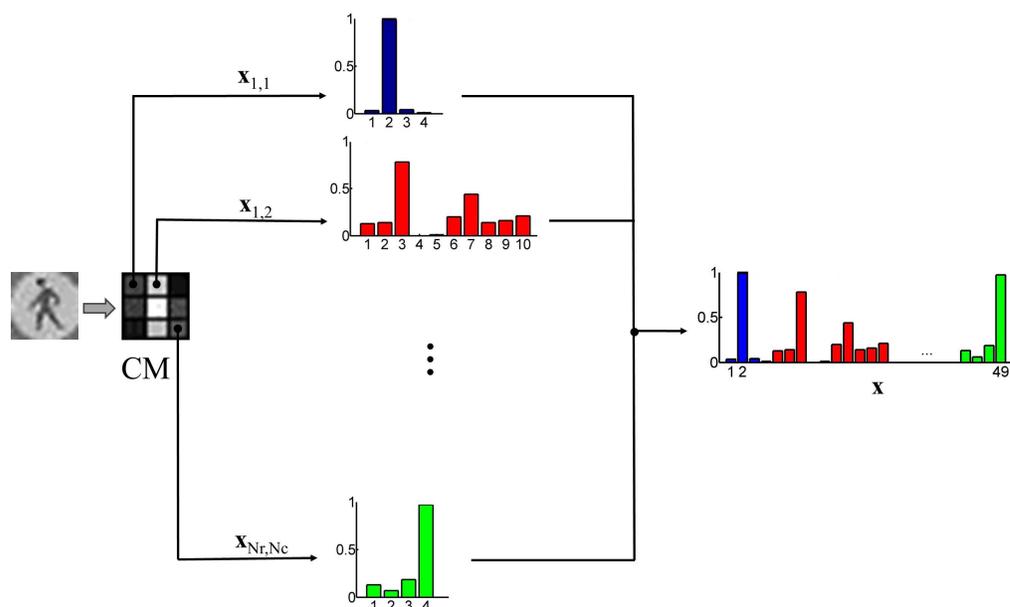


Figura 6.10: Extracción del descriptor basado en un vector HOG de longitud variable para cada celda.

donde  $\cup \{\cdot\}$  denota el operador de concatenación. El procedimiento completo de extracción del descriptor es mostrado en la figura 6.10.

Aunque la información de las celdas ha sido caracterizada mediante el descriptor HOG, otros descriptores podrían emplearse para describir la información. Así, en el caso de trabajar directamente con la concatenación de la luminancia de los píxeles, se podría introducir para cada celda un factor de muestreo en el dominio espacial dependiente del nivel  $u(r, c)$  del mapa del cluster.

Al igual que en el proceso de selección de clusters, los clasificadores dentro de cada cluster han sido implementados también mediante SVM con kernel RBF. Al adoptar una arquitectura OVA para la clasificación dentro de cada cluster, disponemos de una batería de clasificadores por cluster donde el número de clasificadores coincide con el número de clases agrupadas. Como el coste computacional de los clasificadores depende del número de clases contempladas, la carga computacional se reduce al trabajar a nivel de clusters respecto a cuando se trabaja con el conjunto completo de clases. De esta manera, cada cluster sólo *ve* sus propias clases.

De la posibilidad de que las falsas alarmas puedan alcanzar la etapa de reconocimiento en el TSDRS si no son eliminadas previamente por la etapa de clasificación de formas, se hace necesario incluir muestras negativas en el conjunto de entrenamiento de los clasificadores dentro de cada cluster. Así, el conjunto de entrenamiento positivo para un clasificador contiene muestras de su propia clase y el conjunto negativo incluye muestras de ruido y muestras del resto de clases agrupadas en el cluster.

## 6.3. Resultados

El algoritmo propuesto fue codificado en Matlab y ejecutado en una Estación de Trabajo Sun Java W1100z con un procesador de 2.6 GHz bajo el sistema operativo Ubuntu GNU/Linux. La precisión alcanzada por el método para el reconocimiento de señales fue cuantificada por medio de la tasa de acierto de positivos (TPR) y la tasa de falsa alarma (FPR). En los experimentos se seleccionó un valor de referencia para FPR del 5 %. En este apartado se estudiará la influencia de dos parámetros claves en el método: el número de clusters en que se particiona el conjunto total de clases y el número de celdas o regiones que define la rejilla. Además, para poder evaluar la mejora introducida por el método propuesto, se comparará éste con un sistema de reconocimiento basado en el descriptor HOG de longitud fija sin selección de clusters. Los experimentos para la evaluación de los parámetros han sido realizados sobre el conjunto de las señales rojas circulares al contener un mayor número de muestras. En cualquier caso, los resultados han sido extendidos a otros conjuntos de señales.

### 6.3.1. Influencia del número de clusters

Para analizar el efecto del número de clusters,  $N$ , se ha seleccionado una rejilla de  $5 \times 5$  celdas y se ha aplicado el algoritmo de clustering  $K - Means$  para las señales rojas circulares. Aunque existen varios algoritmos de agrupamiento, el  $K - Means$  es uno de los más populares por su simplicidad. Con el objetivo de validar los clusters resultantes, se ha utilizado el coeficiente de silueta [Bolshakova03], el cual proporciona una medida de la proximidad de un punto de un cluster a los puntos de los clusters vecinos. El rango de este coeficiente está acotado entre 1 (caso de puntos muy distanciados de los clusters vecinos) y 0 (para puntos que no son distinguibles de pertenecer a uno u otro cluster). Por tanto, cuanto mayor es el valor del perfil de silueta, mejor es la calidad del clustering. La figura 6.11 muestra el valor medio del perfil de silueta en función del número de clusters y se puede observar que aunque no hay grandes variaciones de valor, el comportamiento es ligeramente mejor para  $N = 2$  y  $N = 4$ , teniendo en ambos casos un valor ligeramente mayor de 0.5.

Los experimentos para evaluar la influencia del número de clusters en la tarea de reconocimiento del pictograma (segunda etapa de la figura 6.5) fueron realizados variando  $N$  entre 1 y 4, con un tamaño fijo de rejilla de  $5 \times 5$  y un número máximo de intervalos HOG  $N_b^{max} = 12$  sobre el conjunto de las señales rojas circulares. La tabla 6.1 incluye toda la información sobre los clasificadores implementados para el reconocimiento del ideograma, aportando por cada cluster el número de clases agrupadas más el ruido, el número de atributos que conforman el descriptor y la suma del número de vectores soporte de la batería de clasificadores dentro de cada cluster.

Observando la tabla 6.1, se puede comprobar la reducción significativa del número de características del descriptor en cada cluster respecto al HOG de longitud fija. El número

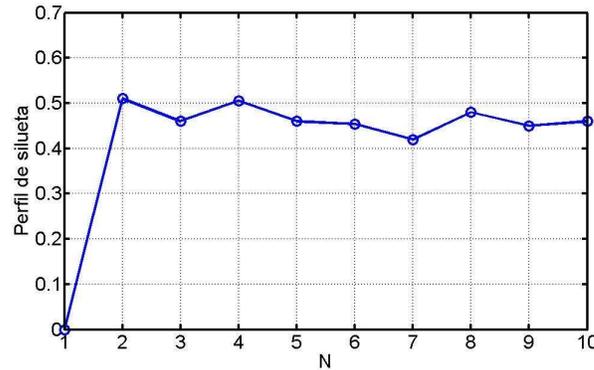


Figura 6.11: Coeficiente de silueta para clustering K-means.

de componentes para todos los clusters es siempre inferior a 200, en contraprestación al HOG de longitud fija que generaría vectores de longitud igual a  $5^2 \times 12 = 300$ , teniendo en cuenta que se ha fijado  $N_b^{max} = 12$ . Además, se puede comprobar en la tabla 6.1 que a medida que las clases se particionan en más clusters, el número total de vectores soporte decrece significativamente. La reducción del número de características y del número de vectores soporte implica que la carga computacional de los clasificadores, incluyendo el tiempo de procesado y el almacenamiento, decrece de manera considerable. Así, en la figura 6.12 se representa el perfil de tiempos de procesado por muestra considerando que la muestra es evaluada únicamente en el cluster con mayor probabilidad de asignación, lo cual implica fijar  $n_c = 1$ . El tiempo total de procesado  $T_t$  implica la suma de tres tiempos: el tiempo correspondiente a la obtención del descriptor  $IM$  ( $T_{desc}$ ), el tiempo empleado para la selección de clusters ( $T_{cluster}$ ) y el tiempo correspondiente al reconocimiento del ideograma ( $T_{rec}$ ). La figura 6.12 evidencia que el tiempo de reconocimiento supone la principal carga de trabajo y que a medida que aumenta el número de clusters, éste se reduce mientras que el tiempo de selección de clusters va cobrando importancia. En cualquier caso, se puede concluir que a medida que aumenta el número de clusters, el tiempo total de procesado disminuye y en promedio decrece 35.5% con  $N = 4$  respecto al caso de  $N = 1$ .

La figura 6.13 muestra el desempeño del método para las condiciones y estructura de clusters de la tabla 6.1 con  $N = 1, \dots, 4$  y  $N_b^{max} = 12$ . Para cada cluster se ha representado la familia de curvas ROC correspondiente a  $n_c = 1, \dots, N$ , indicándose la tasa de detección de positivos TPR para una FPR=5% y el área encerrada bajo la curva (AUC). Comparando la tasa de acierto con  $n_c = 1$  para los diferentes clusters se comprueba que la calidad del sistema experimenta una ligera degradación a medida que aumenta  $N$ . Por otra parte, se demuestra que no hay una clara mejora en la precisión a medida que aumenta el valor de  $n_c$  para un número fijado de clusters, lo cual indica que la fusión de salida de las SVM provoca errores de clasificación. La justificación se encuentra en que los clasificadores de cada cluster son entrenados únicamente con muestras de las

(a)  $N=1$  ( $N_{sv}$ : 6324)

Índice de cluster	# Clases	# Atributos	$N_{sv}$
1	57+1	171	6324

(b)  $N=2$  ( $N_{sv}$ : 5670)

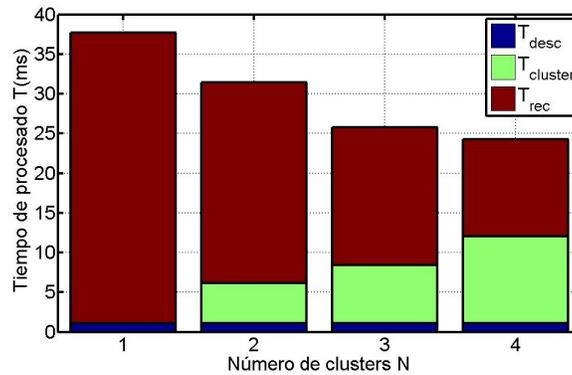
Índice de cluster	# Clases	# Atributos	$N_{sv}$
1	16+1	184	1357
2	40+1	159	4313

(c)  $N=3$  ( $N_{sv}$ : 5521)

Índice de cluster	# Clases	# Atributos	$N_{sv}$
1	7+1	141	447
2	35+1	157	3911
3	13+1	186	1163

(d)  $N=4$  ( $N_{sv}$ : 5255)

Índice de cluster	# Clases	# Atributos	$N_{sv}$
1	5+1	144	271
2	22+1	165	2657
3	12+1	192	1185
4	15+1	135	1142

Tabla 6.1: Información de los clusters con una rejilla de  $5 \times 5$ .Figura 6.12: Tiempo total de procesado por muestra en función del número de clusters con una rejilla de  $5 \times 5$  y  $n_c = 1$ .

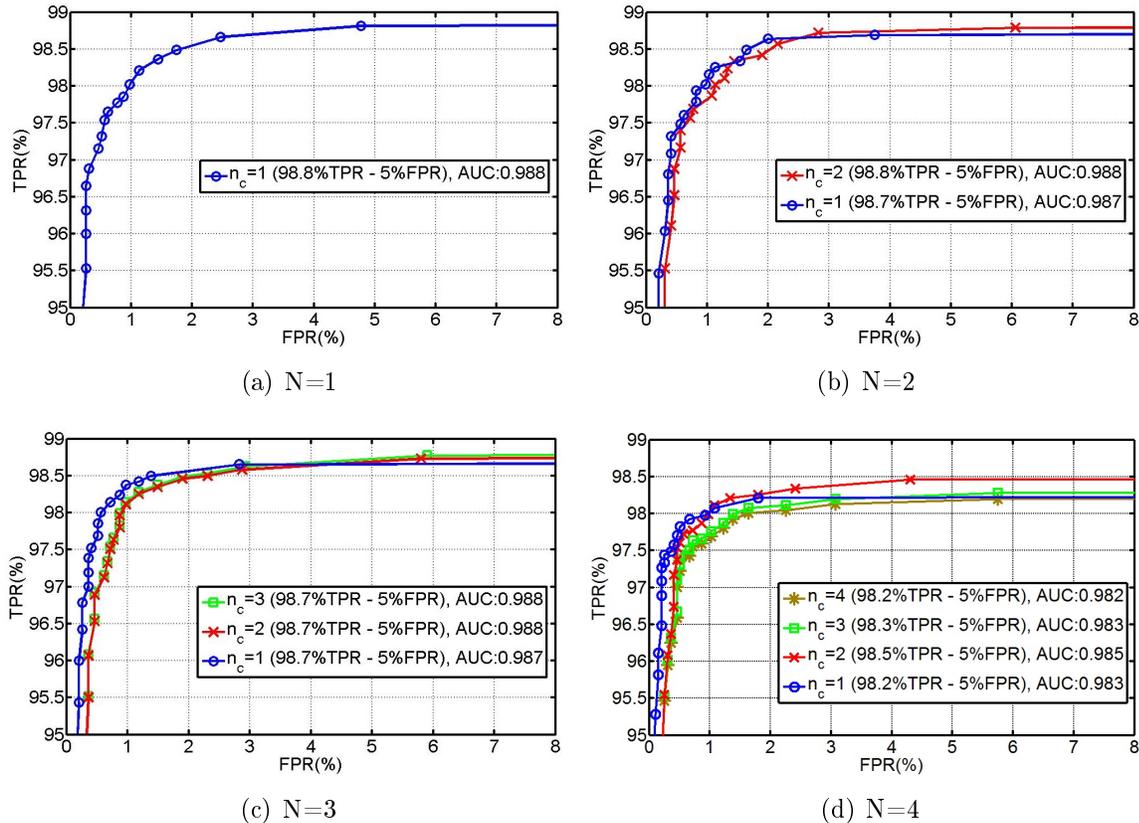


Figura 6.13: Curvas ROC en función del número de clusters.

clases de su propio cluster y de esta manera, dentro de cada cluster no existe conocimiento del resto de clases. Por tanto, el conocimiento es atomizado y la fusión resultante de las salidas de los clusters seleccionados no es fiable. Además, a medida que la muestra entra en más clusters, el tiempo de procesamiento aumenta de forma aproximadamente lineal (ver figura 6.14) y por tanto, estas circunstancias justifican que la mejor opción es realizar el proceso de reconocimiento únicamente dentro del cluster con más alta probabilidad.

### 6.3.2. Influencia del tamaño de la rejilla

Otro estudio fue realizado para analizar el impacto del tamaño de la rejilla en el método propuesto. Para ello, se probaron cinco estructuras diferentes correspondientes a  $2 \times 2$ ,  $3 \times 3$ ,  $4 \times 4$ ,  $5 \times 5$  y  $6 \times 6$  celdas para las señales rojas circulares con  $N = 4$ ,  $n_c = 1$  y  $N_b^{max} = 12$ . En la figura 6.15 se muestran las curvas ROC y se indican las tasas de acierto correspondientes a una FPR=5%. La curva ROC para la estructura  $2 \times 2$  no ha sido mostrada debido a su bajo desempeño: con una tasa de acierto igual a 75.44% para FPR=5%. Como se puede comprobar, a medida que aumenta el número de celdas, aumenta también la tasa de acierto debido a que el descriptor define la información con más detalle y los mejores

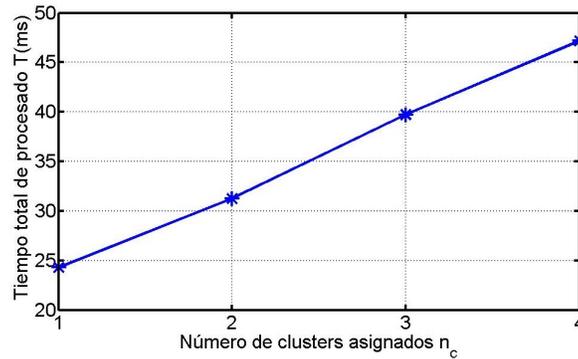


Figura 6.14: Tiempo total de procesado por muestra en función del número de clusters asignados con  $N = 4$  y una rejilla de  $5 \times 5$ .

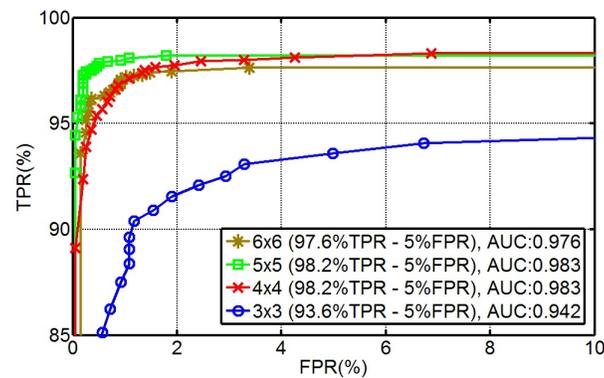


Figura 6.15: Tasa de acierto en función del tamaño de la rejilla con  $N = 4$  y  $n_c = 1$ .

resultados se consiguen con las estructuras de  $4 \times 4$  y  $5 \times 5$ . No obstante, a partir de  $6 \times 6$  la precisión sufre una lenta degradación debido a que para estructuras con tan alta partición de celdas, la variación es alta entre los IMs correspondientes a las clases agrupadas bajo un mismo cluster y el CM del cluster no es capaz de recoger las variaciones de la distribución espacial de los diferentes pictogramas. Como resultado cuantitativo de esta justificación, en la tabla 6.2 se ha recogido el promedio de las varianzas de los IMs de cada cluster.

Respecto a la carga computacional, las figuras 6.16 y 6.17 recogen, respectivamente, el número de atributos que define el descriptor de cada cluster y el número total de vectores soporte en función del tamaño de la rejilla. Ambas gráficas evidencian que a medida que aumenta el número de celdas de la rejilla, también lo hace el número de características y el número de vectores soporte de forma prácticamente exponencial. Esta circunstancia redundante en el aumento de la carga computacional y tal como puede comprobarse en la figura 6.18, el tiempo de procesado sigue una evolución exponencial, alcanzándose un consumo de tiempo excesivo a partir de la rejilla  $6 \times 6$ .

Índice Cluster	Tamaño de rejilla				
	2×2	3×3	4×4	5×5	6×6
1	0.02	0.03	0.05	0.06	0.06
2	0.01	0.03	0.02	0.02	0.03
3	0.01	0.01	0.03	0.06	0.04
4	0.01	0.01	0.03	0.03	0.07

Tabla 6.2: Promedio de las varianzas de los IMs en cada cluster en función del tamaño de la rejilla con  $N = 4$ .

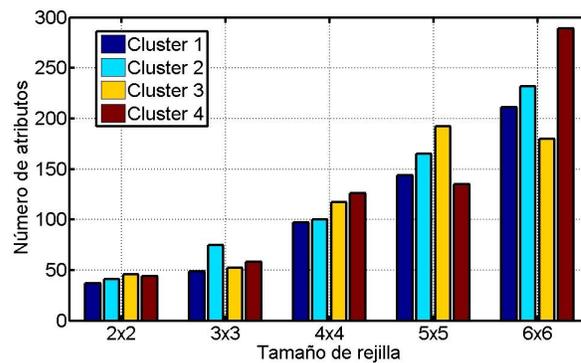


Figura 6.16: Número de características del vector HOG en cada cluster en función del tamaño de la rejilla con  $N = 4$ .

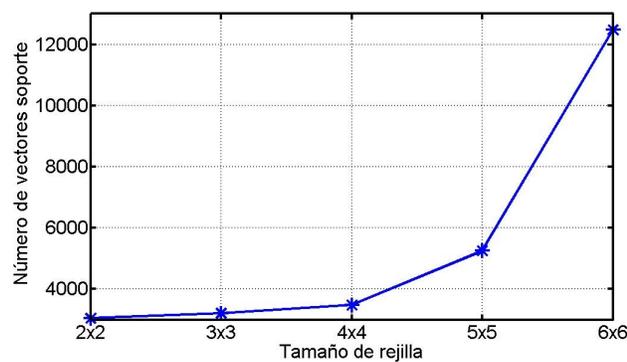


Figura 6.17: Número total de vectores soporte en función del tamaño de la rejilla con  $N = 4$ .

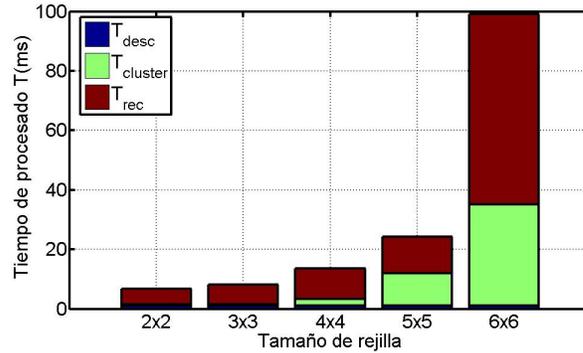


Figura 6.18: Tiempo de procesado por muestra para diferentes estructuras de rejilla con  $N = 4$  y  $n_c = 1$ .

### 6.3.3. Comparativa de métodos

En los apartados anteriores se ha analizado la influencia del número de clusters y el número de celdas sobre la precisión del sistema y su carga computacional. En general, ha quedado demostrado que se puede reducir notablemente la velocidad del sistema y mantener la precisión a medida que se aumenta el número de clusters, hasta un cierto límite. Para evaluar mejor el impacto real del método, se ha procedido a comparar la precisión y el tiempo con respecto a un sistema de reconocimiento con arquitectura HOG+SVM clásico, que puede verse como un caso particular del método propuesto con un único cluster en el que el número de intervalos del descriptor HOG es igual para todas las celdas (es decir,  $N_b = N_b^{max}$ ).

En la tabla 6.3 se incluye un análisis comparativo entre el algoritmo propuesto y el HOG clásico, indicando la precisión del sistema para una FPR=5% y el tiempo de carga por muestra, mostrando asimismo el porcentaje de reducción temporal  $\Delta t$  respecto al HOG clásico. Para este experimento se han fijado los siguientes parámetros: ( $N_r = N_c$ ) = 5,  $n_c = 1$  y  $N_b^{max} = 12$ . Los resultados muestran claramente que el método propuesto presenta un desempeño comparable al HOG clásico en todos los casos, llegando incluso a superarlo ligeramente para  $N$  igual a 1, 2 y 3, mientras que la reducción de tiempo de procesado respecto al HOG clásico es sumamente destacable: 62.7%, 69.3%, 74.5% y 76.0% para  $N = 1, 2, 3, 4$ , respectivamente. De este modo, queda demostrada la gran reducción de carga computacional que se consigue con el método propuesto.

## 6.4. Generalización

Para generalizar los resultados a otras combinaciones, aplicamos el método propuesto a los conjuntos de señales azules rectangulares y rojas triangulares. Para cada combinación, se ha seleccionado una rejilla de  $5 \times 5$  en ambos casos y se ha fijado  $n_c = 1$  y  $N_b^{max} = 12$ .

Método	$TPR(\%)$	$T_t(\text{ms})-\Delta t(\%)$
HOG	98.6	102.58 (—)
Selección cluster+HOG variable (N=1)	98.8	38.26 (62.7%)
Selección cluster+HOG variable (N=2)	98.7	31.43 (69.3%)
Selección cluster+HOG variable(N=3)	98.7	26.15 (74.5%)
Selección cluster+HOG variable(N=4)	98.2	24.61 (76.0%)

Tabla 6.3: Análisis comparativo entre el método propuesto (selección cluster+HOG variable) y HOG.

(a) Señales azules rectangulares (5×5)

Método	$TPR(\%)$	$T_t(\text{ms})-\Delta t(\%)$
HOG	98.2	32.45 (—)
Selección cluster+HOG variable (N=1)	98.7	20.18 (37.8%)
Selección cluster+HOG variable (N=2)	98.1	11.06 (65.9%)
Selección cluster+HOG variable(N=3)	97.8	9.31 (71.3%)
Selección cluster+HOG variable(N=4)	97.4	9.15 (71.8%)

(b) Señales rojas triangulares (5×5)

Método	$TPR(\%)$	$T_t(\text{ms})-\Delta t(\%)$
HOG	98.7	63.55 (—)
Selección cluster+HOG variable (N=1)	98.8	22.83 (64.07%)
Selección cluster+HOG variable (N=2)	99	16.16 (74.57%)
Selección cluster+HOG variable(N=3)	98.9	15.04 (76.33%)
Selección cluster+HOG variable(N=4)	98.7	14.61 (77.01%)

Tabla 6.4: Análisis comparativo entre el método propuesto (selección cluster+HOG variable) y HOG.

Los resultados se muestran en la tabla 6.4, indicando la precisión del sistema para una  $FPR=5\%$  y el tiempo de carga por muestra. Sobre estos nuevos conjuntos, también se observa una importantísima reducción de tiempo respecto al HOG clásico, con un 65.9% y un 77.01% para las señales azules rectangulares y las rojas triangulares, respectivamente, manteniendo la misma precisión. Así, se puede concluir que los excelentes resultados obtenidos avalan la eficacia del algoritmo.

## 6.5. Resumen

En este capítulo se ha presentado un nuevo algoritmo para optimizar la carga computacional de un sistema de reconocimiento basado en SVM para señalización vial. El algo-

ritmo realiza una búsqueda de las regiones donde se localiza la información con el doble objetivo de agrupar las señales con similar distribución del mensaje y efectuar el reconocimiento dentro de los agrupamientos. Además, los descriptores para el reconocimiento se adaptan de manera dinámica a cada región en función de la cantidad de información contenida. De esta manera, la optimización de la carga computacional se consigue a través de la menor dimensión de los vectores y de reducir el número de vectores soporte de los clasificadores.

Los resultados han demostrado que la alta tasa de acierto en la selección de clusters permite trabajar únicamente con las clases del primer cluster seleccionado para la muestra de test. Los experimentos llevados a cabo analizaron la sensibilidad cualitativa de la precisión y del tiempo de procesado en función del número de clusters en que se particionan las clases y del número de celdas en que se estructura la rejilla. Para los conjuntos de señales utilizados, la carga computacional se redujo en algunos conjuntos de datos hasta más de un 75%, lo cual demuestra la excelente optimización conseguida. Aunque este método se ha propuesto para el reconocimiento de señalización vial, puede ser orientado a cualquier otra aplicación para multclasificación de imágenes.



# Capítulo 7

## Contribuciones y futuras líneas de investigación

A continuación se enumeran las principales contribuciones originales que han sido aportadas en esta tesis. Posteriormente se comentarán las futuras líneas de investigación que quedan abiertas tras el desarrollo de la misma.

### 7.1. Contribuciones originales

#### 7.1.1. Sistema para Detección y Reconocimiento de Señalización Vertical

En el capítulo dedicado a la implementación del sistema completo para detección y reconocimiento de señalización se introducen algoritmos en cada una de las etapas correspondientes a la segmentación, la clasificación, el reconocimiento y el seguimiento. En este contexto se destacan las siguientes aportaciones originales:

- Creación de una base de datos anotada para señalización vertical con imágenes reales. La base de datos incluye más de 50 000 imágenes de entornos viales capturadas en diferentes rutas y ciudades de la geografía española, tanto en entornos urbanos como interurbanos. Las muestras de señalización extraídas a partir de las escenas permiten realizar los correspondientes procesos de entrenamiento y testeo de los sistemas supervisados.
- Se ha introducido un método para la segmentación de señales basado en la umbralización del espacio HSI. El modelo realiza un análisis de la distribución estadística de los canales de color propios de las señales para día y noche.
- Se ha introducido un novedoso descriptor para clasificación de formas geométricas basado en la medida de distancia de cada uno de los lados del *bounding-box* que circunscribe la señal al contorno más cercano de la misma. El algoritmo presenta

invarianza al escalado y al desplazamiento y además, posee un comportamiento robusto frente a las rotaciones.

- Se ha implementado un método de reconocimiento de señalización que emplea como descriptor los niveles de luminancia de los píxeles e introduce las SVM como herramienta de clasificación en este campo. El trabajo de investigación desarrollado ha supuesto el punto de partida de posteriores trabajos en la literatura científica alrededor de las SVM en el campo de la señalización.
- Se ha implementado un módulo de seguimiento basado en las características visuales de las señales, tanto a nivel de imagen independiente como a nivel secuencial entre imágenes. Asimismo, en el seguimiento secuencial se ha empleado un filtro de Kalman que integra la información de los sensores del sistema de captura.

### 7.1.2. Arquitecturas de reconocimiento basadas en SVM

En el capítulo de arquitecturas de reconocimiento basadas en SVM la investigación se centra en el módulo de identificación del ideograma. El módulo se estructura en tres etapas correspondientes al preprocesado, la extracción de descriptores y la clasificación, abordando un estudio comparativo de diferentes técnicas con el objetivo de combinar las que proporcionan mejores prestaciones desde el doble compromiso de precisión y velocidad. Una vez realizada esta propuesta, se han realizado las siguientes aportaciones:

- Se han propuesto diferentes técnicas de preprocesado y descriptores y se han comparado diferentes arquitecturas de multclasificación con SVM. Algunos de los descriptores, en concreto LBP, RIFT y SPIN, se han empleado por primera vez en el campo del reconocimiento de señales de tráfico. A través de las técnicas de preprocesado se intentan reducir las variaciones de iluminación y mediante los descriptores se buscan las mejores estrategias para cuantificar de manera eficiente la información de la imagen.
- En el marco del análisis comparativo se han determinado las mejores combinaciones preprocesado-descriptor-arquitectura de clasificación. En cualquier caso, la adecuación de las técnicas depende de la interrelación entre las tres etapas.
- Se ha realizado un estudio paramétrico con el objetivo de analizar la influencia de los parámetros vinculados a los diferentes algoritmos de cara a conseguir optimizar sus valores en el resultado final del reconocimiento.

### 7.1.3. Reconocimiento de señales basado en mapas de pictogramas

La aportación fundamental de este capítulo es la implementación de un novedoso algoritmo de reconocimiento multiclase con muy reducida carga computacional. Aunque el

método ha sido utilizado para un sistema de identificación de señales viales, éste es perfectamente exportable a la mayor parte de los sistemas de reconocimiento supervisados con descriptores basados en el dominio espacial, independientemente de la aplicación. La carga computacional es reducida significativamente en base a dos aspectos: 1) la tarea de reconocimiento se lleva a cabo sobre un número de clases más restringido, y 2) el descriptor a utilizar es diferente para cada uno de los clusters, adaptándose a las regiones donde se concentra el mensaje informativo en cada agrupamiento. El método puede generalizarse para cualquier otro descriptor basado en el dominio espacial y cualquier técnica de clasificación estadística. Las aportaciones de este capítulo pueden resumirse en los siguientes puntos:

- Se ha implementado un algoritmo de búsqueda de características comunes entre clases para la obtención de agrupamientos. De este modo, se reducen los tiempos de entrenamiento y testeo al subdividir el problema de multclasificación en problemas más sencillos de manejar.
- Se ha diseñado un método que permite trabajar con descriptores específicos de cada cluster, los cuales se centran en las regiones donde se extiende el pictograma de la señal. El hecho de manejar un descriptor adaptativo permite reducir la dimensión de los vectores característicos en las zonas donde no se expande el mensaje informativo de la señal.
- Se ha realizado un estudio sobre el impacto que tienen, a nivel de velocidad y precisión, los dos principales parámetros vinculados al algoritmo: el número de agrupamientos y el tamaño de la rejilla de partición de la muestra.

## 7.2. Futuras líneas de investigación

Los estudios realizados y la implementación de algoritmos necesarios para las distintas aplicaciones dejan abierta la posibilidad de abrir o ampliar distintas líneas de investigación.

En el apartado del diseño de un sistema para la detección y el reconocimiento de la señalización, las líneas de investigación son las siguientes:

- Aunque en esta tesis doctoral se ha trabajado en la detección y reconocimiento de señalización vertical, existe la posibilidad de exportar y ampliar los métodos de investigación a la detección e identificación de otros elementos de señalización vial menos frecuentes, tales como paneles informativos, marcas horizontales, elementos de balizamiento en tramos de obras y semáforos.
- Uno de los problemas a la hora de comparar algoritmos para detección y reconocimiento de señales es la falta de bases de datos estandarizadas. Hasta ahora, los

diferentes grupos de investigación han testeado sus métodos sobre sus propias imágenes. Aunque recientemente se han hecho públicas varias bases de datos, ninguna de ellas ha sido extensamente utilizada. Por tanto, queda pendiente definir un método para evaluar el desempeño de los diferentes algoritmos, adoptar una base de datos europea estándar realista y completa, que incluya un conjunto suficientemente representativo con muestras de todas las clases, y que aporte secuencias para los métodos de seguimiento.

- El interés de esta tesis se centra en el reconocimiento de la señalización en las imágenes captadas por el sistema de captura. Sin embargo, en el trabajo de la tesis no se ha contemplado la posibilidad de estudiar si las señales afectan a la vía de circulación del automóvil. Es muy probable que la investigación en la detección de carriles pueda aprovecharse en este objetivo.
- La detección automática de señalización tiene una importante aplicación en el inventariado automático de carreteras en el que, además de recoger la localización y la categoría de la señal, es interesante aportar también las dimensiones físicas reales de la señal y su estado de conservación. El grupo de investigación GRAM de la UAH ha desarrollado trabajos preliminares sobre ambos aspectos, especialmente, en el análisis del deterioro de los materiales en base a medidas de reflectividad. La continuidad de estos trabajos y su integración con el sistema RESET supondrá la evolución lógica hacia un sistema completo de caracterización de señalización.
- La etapa de clasificación del sistema RESET es capaz de identificar la forma geométrica de la señal y a partir de la detección de los vértices, ésta se reorienta de acuerdo a una posición de referencia. La inexistencia de vértices en el caso de las señales circulares conlleva la posibilidad de plantear métodos alternativos para detectar las rotaciones y deformaciones proyectivas en base únicamente al mensaje del pictograma.

Respecto al análisis comparativo de estrategias para la tarea de reconocimiento, las líneas futuras vendrán marcadas por los siguientes puntos:

- Aunque en esta tesis doctoral se ha trabajado con numerosos descriptores, se hace necesario estudiar el efecto de otras características actuales, tales como los filtros de Haar, que realizan una comparación de niveles de intensidad de la imagen a nivel local en regiones adyacentes, y los dipolos disociados, que comparan regiones separadas en la imagen. Ambas características ya han sido empleadas con éxito en aplicaciones orientadas al reconocimiento de peatones y de caras.
- Siguiendo en la misma línea, sería posible introducir algunas de las herramientas de multclasificación con buenos resultados en otros campos, tales como *AdaBoost*, *Bag of Words* y *RandomForest*.

- Las técnicas emergentes basadas en clasificadores débiles puede ser una interesante estrategia para encontrar características comunes entre las clases. La idea sería entrenar los detectores de forma conjunta, en vez de hacerlo de forma independiente. De este modo, cabe esperar que las características iniciales sean compartidas por muchas clases y progresivamente, se obtengan características más específicas. Aunque existen trabajos iniciales aplicados a la detección de objetos, aún no se ha explotado esta línea en el ámbito del reconocimiento de las señales viales.

En lo que respecta al algoritmo de reconocimiento basado en mapas de pictogramas, algunos futuros estudios serán:

- Respecto a la reducción de carga computacional de la etapa de reconocimiento mediante SVM, sería posible combinar el método propuesto con otros algoritmos publicados en la literatura orientados a reducir el número de vectores soporte en base a la búsqueda de combinaciones lineales entre vectores soporte, simplificación de la función de decisión mediante generalización de la curvatura a través de la exclusión de ciertos vectores soporte y agrupamiento de vectores soporte mediante la búsqueda de efectos similares y contrapuestos. Además, en esta misma línea es importante destacar que el cálculo del valor de salida de las SVM para los clasificadores correspondientes a las diferentes clases es paralelizable. Esta circunstancia podría explotarse en plataformas multinúcleo mediante unidades de procesamiento gráfico (GPU).
- Aunque el agrupamiento del método de optimización se ha realizado mediante el método K-means sería interesante explorar los resultados con otras técnicas de clustering, tales como clustering por consenso, el cual minimiza el número de desacuerdos entre un conjunto de clusters, el clustering jerárquico, que permite estructurar las clases en diferentes niveles de agrupamiento, y el clustering espectral.
- La búsqueda de regiones de interés en el dominio espacial para el agrupamiento de clases y para la extracción dinámica de los descriptores se ha realizado a partir de una rejilla de celdas rectangulares. Debido a la geometría circular y triangular de muchas señales, sería conveniente plantear una distribución de regiones acorde a la forma geométrica de la señal detectada.
- La búsqueda de regiones de interés se ha efectuado a partir de la magnitud del gradiente. Sería razonable probar con técnicas para detección de puntos de interés, tales como los detectores SIFT y SURF. Algunos de los métodos, tales como el detector de Harris presentan a priori una gran invarianza a la rotación, variación de iluminación y ruido.



# APÉNDICES



# APÉNDICE A

## Publicaciones a las que ha dado lugar la tesis

En este apéndice se enumeran las publicaciones a las que ha dado lugar la realización de la tesis, tanto a nivel de revistas como de congresos internacionales y nacionales.

### A.1. Revistas

- **S. Lafuente-Arroyo**, S. Maldonado-Bascón, H. Gómez-Moreno, P. Gil-Jiménez y R.J. López-Sastre. Traffic sign identification using clusters of ideogram maps. *IEEE Trans. on Intelligent Transportation Systems*. En proceso de segunda revisión. Enviado en diciembre de 2013.
- S. Maldonado-Bascón, F.J. Acevedo-Rodríguez, **S. Lafuente-Arroyo**, A. Fernández-Caballero y F. López-Ferreras. An optimization on pictogram identification for the road-sign recognition task using SVMs. *Computer Vision and Image Understanding*, 114(3):373-383, Marzo 2010. ISSN 1077-3142.
- **S. Lafuente-Arroyo**, S. Salcedo-Sanz, S. Maldonado-Bascón, J.A. Portilla-Figueras y R.J. López-Sastre. A decision support system for the automatic management of keep-clear signs based on support vector machines and geographic information systems. *Expert Systems with Applications*, 37(1):767-773, Enero 2010. ISSN 0957-4174.
- P. Siegmann, R.J. López-Sastre, P. Gil-Jiménez, **S. Lafuente-Arroyo** y S. Maldonado-Bascón. Fundamentals in luminance and retroreflectivity measurements of vertical traffic signs using a colour digital camera. *IEEE Trans. on Instrumentation and Measurement*, 57(3):607-615, Marzo 2008. ISSN 0018-9456.
- S. Maldonado-Bascón, **S. Lafuente-Arroyo**, P. Gil-Jiménez, H. Gómez-Moreno y F. López-Ferreras. Road-sign detection and recognition based on support vector

machines. *IEEE Trans. on Intelligent Transportation Systems*, 8(2):264-278, Junio 2007. ISSN 1524-9050. 10.1109/TITS.2007.895311.

## A.2. Congresos internacionales

- **S. Lafuente-Arroyo**, R.J. López-Sastre, S. Maldonado-Bascón y R. Martínez-Tomás. Discriminant splitting of regions in traffic sign recognition. En *5th International Work-conference on the Interplay between Natural and Artificial Computation (IWINAC)*, páginas: 395-403, Palma de Mallorca (España), Junio 2013.
- **S. Lafuente-Arroyo**, S. Maldonado-Bascón, H. Gómez-Moreno y P. Gil-Jiménez. False alarm filtering in a vision traffic sign recognition system-an approach based on Adaboost and heterogeneity of texture. En *Proceedings of the III International Conference on Agents and Artificial Intelligent (ICAART)*, páginas: 269-276, Roma (Italia), Enero 2011.
- **S. Lafuente-Arroyo**, S. Maldonado-Bascón, P. Gil-Jiménez y H. Gómez-Moreno. An intra-image tracking algorithm for traffic sign recognition. En *Proceedings of the IEEE International Conference on Electronics and Safety (ACVIS)*, páginas: 259-264, Columbia (USA), Septiembre 2008.
- **S. Lafuente-Arroyo**, S. Maldonado-Bascón, P. Gil-Jiménez, F.J. Acevedo-Rodríguez y R.J. López-Sastre. A tracking system for automated inventory of road signs. En *Proceedings of the IEEE Intelligent Vehicles Symposium*, páginas: 166-171, Estambul, Junio 2007.
- **S. Lafuente-Arroyo**, A. Sánchez-Fernández, S. Maldonado-Bascón, P. Gil-Jiménez y F.J. Acevedo-Rodríguez. Road sign recognition using spatial dimension reduction methods based on PCA and SVMs. En *Proceedings of the 9th International Work Conference on Artificial Neural Networks (IWANN)*, páginas: 725-732, San Sebastián (España), Junio 2007.
- **S. Lafuente-Arroyo**, S. Maldonado-Bascón, P. Gil-Jiménez, H. Gómez-Moreno y F. López-Ferreras. Road sign tracking with a predictive filter solution. En *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics (IECON)*, páginas: 3314-3319, París (Francia), Noviembre 2006.
- **S. Lafuente-Arroyo**, P. Gil-Jiménez, R. Maldonado-Bascón, F. López-Ferreras y S. Maldonado-Bascón. Traffic sign shape classification evaluation I: SVM using Distance to Borders. En *Proceedings of the IEEE Intelligent Vehicles Symposium*, páginas: 557 - 562, Las Vegas (USA), Junio 2005.

### A.3. Congresos nacionales

- M. González-Fernández, **S. Lafuente-Arroyo**, R.J. López-Sastre, S. Maldonado-Bascón. Descripción de un sistema para inventariado de elementos de señalización en carretera. En Actas IX Congreso Español Intelligent Transportation Systems, Andorra La Vella (España).

### A.4. Menciones

El artículo ‘Road-sign detection and recognition based on support vector machines’, publicado en la revista *IEEE Trans. on Intelligent Transportation Systems* en Junio de 2007, fue elegido como uno de los diez mejores de la década 2000-2009, tal como se recoge en la publicación ‘The Top and the Best: Toward Excellence in ITS Research and Development’, *IEEE Trans. on Intelligent Transportation Systems*, 14(3):1033-1034, Septiembre 2013. Número de citas: 289 (Fuente: Google Scholar).



# APÉNDICE B

## Máquinas de Vectores Soporte

Las máquinas de vectores soporte (*SVM-Support Vector Machines*) constituyen una técnica supervisada para clasificación de patrones y regresión basada en la teoría de aprendizaje estadístico. Esta teoría fue propuesta por Vapnik y su grupo de investigación en los laboratorios AT&T Bell en la década de los años 90 [Vapnik95, Vapnik98].

### B.1. Nociones sobre aprendizaje estadístico

En un problema genérico de clasificación en el que se dispone de un conjunto etiquetado de entrenamiento con  $N$  muestras independientes e idénticamente distribuidas  $\{(\mathbf{x}_i, y_i)_{i=1}^N\}$  y donde los vectores  $\mathbf{x}_i$  vienen dados por una función de densidad de probabilidad  $p(\mathbf{x})$ , la correspondencia entre las muestras y sus correspondientes etiquetas  $y_i$  se puede expresar mediante una función de densidad de probabilidad condicionada  $p(y|\mathbf{x})$  desconocida.

El sistema clasificador emplea una familia de funciones  $f(\mathbf{x}, \mathbf{w})$ , con  $\mathbf{w} \in W$ , para proporcionar una etiqueta  $y_i = f(\mathbf{x}_i, \mathbf{w})$ ,  $\mathbf{w} \in W$ , dada una entrada  $\mathbf{x}_i$ . El objetivo del proceso de aprendizaje es seleccionar de manera óptima la función  $f(\mathbf{x}, \mathbf{w})$ ,  $\mathbf{w} \in W$ , de modo que se minimicen las discrepancias entre las etiquetas verdaderas y las que proporciona la máquina. Para evaluar la bondad de esta aproximación se define una función de coste  $c(y, f(\mathbf{x}, \mathbf{w}))$  que cuantifica la penalización asociada a una cierta decisión  $\hat{y}_i = f(\mathbf{x}_i, \mathbf{w})$  cuando la etiqueta verdadera es  $y_i$ . Una vez establecida dicha penalización, se puede definir el riesgo en el que incurre la máquina al emplear la función  $f(\mathbf{x}, \mathbf{w})$  como:

$$R(f) = \int c(y, f(\mathbf{x}, \mathbf{w}))p(\mathbf{x}, y)d\mathbf{x}dy. \quad (\text{B.1})$$

Por lo general, no se conoce la función de densidad de probabilidad conjunta  $p(\mathbf{x}, y)$ , disponiéndose únicamente de un conjunto etiquetado de muestras de entrenamiento. En este caso, se define el riesgo empírico como:

$$R_{emp}(f) = \frac{1}{N} \sum_{i=1}^N c(y, f(\mathbf{x}, \mathbf{w})). \quad (\text{B.2})$$

El principio inductivo de minimización del riesgo empírico (*ERM-Empirical Risk Minimization*) recomienda la selección de la función  $f(\mathbf{x}, \mathbf{w})$ ,  $\mathbf{w} \in W$ , que minimiza la ecuación B.2. Sin embargo, el principio ERM no es un criterio óptimo para el problema de aprendizaje máquina, ya que no asegura incondicionalmente la minimización ni del riesgo real B.1 ni del error de testeo, debido al sesgo producido por disponer de un número limitado de datos de entrenamiento. Además de un riesgo empírico reducido, el entrenamiento debe producir máquinas capaces de generalizar, es decir, capaces de producir resultados satisfactorios ante muestras de entrada desconocidas.

Para conseguirlo, en la práctica se impone una serie de restricciones sobre las clases de funciones  $f(\mathbf{x}, \mathbf{w})$ ,  $\mathbf{w} \in W$ , susceptibles de ser implementadas por el sistema. Este conjunto de funciones debe poseer una capacidad de aprendizaje suficiente para el problema tratado pero, al mismo tiempo, debe controlarse su complejidad de forma que el entrenamiento no se ajuste excesivamente a las muestras disponibles. La teoría del aprendizaje estadístico plasma estas ideas en el principio de minimización del riesgo estructural (*SRM-Structural Risk Minimization*). El criterio de minimización del riesgo empírico, junto con la necesidad de controlar la capacidad de aprendizaje del conjunto de funciones empleado por la máquina, conducen a la definición del riesgo estructural  $R_{est}(f)$  como cota superior del riesgo real B.1:

$$R(f) \leq R_{est}(f) = R_{emp}(f) + \phi(h), \quad (\text{B.3})$$

donde  $h$  es una medida de la capacidad o complejidad del conjunto de funciones  $f(\mathbf{x}, \mathbf{w})$ ,  $\mathbf{w} \in W$ , y el término  $\phi(h)$  refleja el riesgo en el que se incurre al emplear dicha familia de funciones. Es preciso señalar que esta expresión es un resultado directo de la condición de convergencia uniforme en probabilidad de los riesgos [Vapnik98].

La cota del riesgo empleada en el ámbito de las SVM se basa en la dimensión VC (Vapnik-Chervonenkis) [Vapnik71] del conjunto de funciones  $f(\mathbf{x}, \mathbf{w})$ . Para el problema de clasificación la dimensión  $VC(h)$  se puede definir como el máximo número  $h$  de vectores que se pueden separar de las  $2^h$  formas posibles usando funciones de dicho conjunto. Si para cualquier  $N$  hay un conjunto de  $N$  vectores que se pueden separar de la forma señalada, se dice que la dimensión VC es infinita. A partir de la dimensión VC, y bajo ciertas simplificaciones, se demuestra que con probabilidad  $1 - \delta$  el riesgo real está acotado por:

$$R(f) \leq R_{emp}(f) + \sqrt{\frac{h(\ln(\frac{2N}{h}) + 1) - \ln(\frac{\delta}{4})}{N}}. \quad (\text{B.4})$$

## B.2. Clasificación mediante SVM

La idea básica de la clasificación con SVM es encontrar la mejor función de separación entre clases distribuidas en un espacio  $d$ -dimensional. El sistema de clasificación más simple corresponde al caso binario, entendiéndose por tal aquel que sólo trabaja con dos

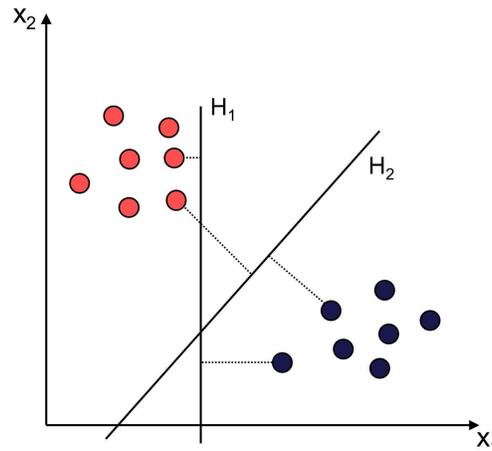


Figura B.1: Problema de separación de patrones con datos separables linealmente.

clases de objetos  $\mathcal{C}_1$  y  $\mathcal{C}_{-1}$ . El objetivo es encontrar un hiperplano que separe las dos clases, de tal modo que cuando se presente un nuevo objeto se le asigne la clase correcta. La figura B.1 muestra un problema de reconocimiento de patrones en dos dimensiones con dos clases linealmente separables. Aunque existen infinitas fronteras de separación que delimitan las dos clases sin error (en la figura B.1 sólo se representan dos fronteras), no se obtiene la misma garantía de éxito cuando se presentan nuevos casos a testear con cualquiera de ellas. En [Vapnik98] se demuestra que minimizar el riesgo estructural es equivalente a encontrar el hiperplano que separa linealmente las dos clases con máximo margen de separación.

### B.2.1. Datos separables linealmente

En un problema de clasificación binaria, donde los datos de entrenamiento vienen dados como un conjunto de  $N$  vectores  $\mathbf{x}_i \in \mathbb{R}^d$  con etiquetas  $y_i \in \{-1, +1\}$ , se construye en el proceso de entrenamiento un hiperplano que es capaz de separar las clases positiva (+1) y negativa (-1). La clasificación lineal se realiza calculando una función lineal de su vector de entrada:

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b, \quad (\text{B.5})$$

donde  $\mathbf{x}$  es el vector de test,  $\mathbf{w}$  el vector director del hiperplano y  $b$  el desplazamiento. La clasificación se realiza en base al producto escalar entre el vector director del hiperplano y el vector de test. Los puntos  $\mathbf{x}$  que se extienden en el hiperplano cumplen la igualdad  $\mathbf{w}^T \mathbf{x} + b = 0$ . Si  $d^+$  ( $d^-$ ) es la distancia desde el hiperplano de separación a la instancia positiva (negativa) más cercana, es posible definir el margen de separación de un hiper-

plano como  $d^+ + d^-$ . Para el caso linealmente separable, la SVM busca el hiperplano que maximiza este margen. Escalando los coeficientes de  $\mathbf{w}$ , se puede hacer que se verifique:

$$\mathbf{w}^T \mathbf{x}_i + b \geq +1 \quad \text{para } y_i = +1 \quad (\text{B.6})$$

y

$$\mathbf{w}^T \mathbf{x}_i + b \leq -1 \quad \text{para } y_i = -1, \quad (\text{B.7})$$

lo cual se combina en un conjunto de inecuaciones como:

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1 \geq 0 \quad \forall i = 1, 2, \dots, N. \quad (\text{B.8})$$

El hiperplano es determinado dentro de un factor de escala y para liberarnos de esta limitación, se aplica una normalización de los hiperplanos candidatos. Recordando que la distancia de un punto al hiperplano viene dada por:

$$z = \frac{|f(\mathbf{x})|}{\|\mathbf{w}\|^2}, \quad (\text{B.9})$$

podemos escalar  $\mathbf{w}$  para que el valor de  $f(\mathbf{x})$  particularizado en los puntos más próximos de  $\mathbf{w}$  sea igual a 1 para  $\mathcal{C}_1$  y -1 para  $\mathcal{C}_{-1}$ . De este modo, el margen de separación sería igual a  $2/\|\mathbf{w}\|$ .

Los puntos que satisfacen la igualdad en la ecuación B.6 se extienden en el hiperplano  $H_1 : \mathbf{w}^T \mathbf{x}_i + b = 1$ , tal como se muestra en la figura B.2. De forma análoga, los puntos que cumplen la igualdad en la ecuación B.7 se extienden en el hiperplano  $H_{-1} : \mathbf{w}^T \mathbf{x}_i + b = -1$ . Cabe notar que los planos  $H_1$  y  $H_{-1}$  son paralelos, ya que vienen definidos por el mismo vector normal, y por otra parte, que ningún punto de entrenamiento cae entre ellos. La búsqueda del hiperplano óptimo de máxima separación se puede plantear como el siguiente problema de optimización:

$$\text{minimizar} \quad J(\mathbf{w}) \equiv \frac{1}{2} \|\mathbf{w}\|^2 \quad (\text{B.10})$$

$$\text{sujeto a: } y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \quad i = 1, 2, \dots, N. \quad (\text{B.11})$$

Así, el hiperplano óptimo puede ser encontrado resolviendo un problema de optimización no lineal (cuadrática) sujeto a un conjunto de inecuaciones. Para ello se trabaja con la formulación de la función de Lagrange, la cual se puede definir como:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) \equiv \frac{\|\mathbf{w}\|^2}{2} - \sum_{i=1}^N \alpha_i [y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1], \quad (\text{B.12})$$

donde  $\boldsymbol{\alpha}$  es el vector de los multiplicadores de Lagrange, con  $\alpha_i \geq 0$  para  $i = 1, 2, \dots, N$ . Hay dos razones para la utilización de la función de Lagrange: 1) las restricciones de la ecuación B.11 son reemplazadas por las condiciones de los multiplicadores de Lagrange, las cuales son bastante más sencillas de manejar, y 2) los datos de entrenamiento en esta

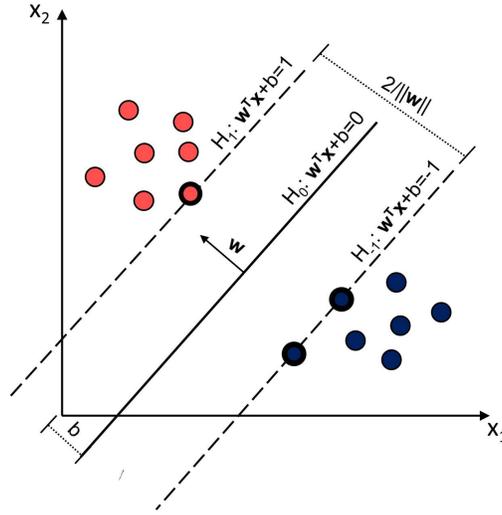


Figura B.2: Clasificación lineal óptima.

formulación sólo aparecen en forma de productos escalares entre los vectores, lo cual es una propiedad crucial que permitirá generalizar el procedimiento al caso no lineal.

El objetivo ahora es minimizar  $L$  con respecto a  $\mathbf{w}$  y  $b$ . Las derivadas parciales de  $L$ , sujetas todas ellas a la restricción  $\alpha_i \geq 0$ , deben anularse<sup>1</sup> en la solución:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial b} = \sum_{i=1}^N y_i \alpha_i = 0 \quad (\text{B.13})$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^N y_i \alpha_i \mathbf{x}_i = 0. \quad (\text{B.14})$$

Las ecuaciones anteriores permiten reescribir B.12 como:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^N \alpha_i y_i \mathbf{w}^T \mathbf{x}_i - b \sum_{i=1}^N y_i \alpha_i + \sum_{i=1}^N \alpha_i. \quad (\text{B.15})$$

Sustituyendo las ecuaciones B.13 y B.14 en la ecuación B.15 se obtiene:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \left( \sum_{i=1}^N y_i \alpha_i \mathbf{x}_i \right)^T \left( \sum_{j=1}^N y_j \alpha_j \mathbf{x}_j \right) - \sum_{i=1}^N y_i \alpha_i \left( \sum_{j=1}^N y_j \alpha_j \mathbf{x}_j \right)^T \mathbf{x}_i + \sum_{i=1}^N \alpha_i \quad (\text{B.16})$$

y equivalentemente

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (y_i y_j \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j) + \sum_{i=1}^N \alpha_i, \quad (\text{B.17})$$

<sup>1</sup>Nótese que sabemos de antemano que el extremo es único y se corresponde con un mínimo, al tratarse de una función cuadrática y convexa.

lo cual lleva a un problema de programación cuadrática que se conoce como forma primaria del problema. Sin embargo, este tipo de problema puede resolverse más fácilmente usando la forma dual del mismo a partir de la derivación parcial de la forma primaria y su igualación a 0. El nuevo problema de optimización sería:

$$\begin{aligned} \text{maximizar } L_D(\boldsymbol{\alpha}) &= -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (y_i y_j \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j) + \sum_{i=1}^N \alpha_i & (\text{B.18}) \\ \text{sujeto a: } & \sum_{i=1}^N y_i \alpha_i = 0, \quad \alpha_i \geq 0. \end{aligned}$$

La función dual  $L_D(\boldsymbol{\alpha})$  tiene una forma cuadrática que puede ser resuelta usando diferentes métodos de optimización. Un método particular para entrenar las SVM es el algoritmo SMO (*Sequential Minimal Optimisation*) [Platt99], el cual descompone el problema de programación cuadrática en problemas más simples. Durante el proceso de maximización, algunas variables duales  $\alpha_i$  toman un valor nulo. Las muestras de entrenamiento con  $\alpha_i > 0$  son los vectores soporte. La solución obtenida sería el hiperplano  $H_0$  de máximo margen buscado que tendrá la forma:

$$\mathbf{w} = \sum_{i=1}^N y_i \alpha_i \mathbf{x}_i = \sum_{i=1}^{N_s} y_i \alpha_i \mathbf{x}_i \quad (\text{B.19})$$

donde  $N_s$  es el número de vectores soporte, siendo  $N_s \leq N$ .

Esto significa que  $\mathbf{w}$ , que describe el hiperplano de separación, es determinado por la suma de los diferentes vectores de entrenamiento ponderados por sus multiplicadores de Lagrange  $\alpha_i$  (en la figura B.2 se muestran los vectores soporte del conjunto de entrenamiento, representándose con contorno grueso).

El parámetro  $b$  puede ser calculado a partir de uno de los vectores soporte, que cumplirá:

$$y_i(\mathbf{w}^T \mathbf{x}_i - b) = 1 \quad (\text{B.20})$$

y despejando quedaría:

$$b = \mathbf{w}^T \mathbf{x}_i - y_i. \quad (\text{B.21})$$

Así, la clasificación de un nuevo patrón de entrada  $\mathbf{x}$  puede ser resuelta a partir de la función de decisión  $f(\mathbf{x})$ :

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b) = \text{sign}\left(\sum_{i=1}^{N_s} y_i \alpha_i \mathbf{x}^T \mathbf{x}_i + b\right). \quad (\text{B.22})$$

El procedimiento y los resultados obtenidos hasta ahora pueden usarse sólo directamente en el caso de que los vectores sean separables y que, además, esta separación se realice mediante una frontera lineal. Sin embargo, en los casos más comunes de problemas de clasificación no se dan esas condiciones. Por tanto, el algoritmo debe ser ampliado para obtener una frontera de decisión en el caso de que los vectores no sean directamente separables y también para el caso en que la frontera de separación óptima no sea lineal.

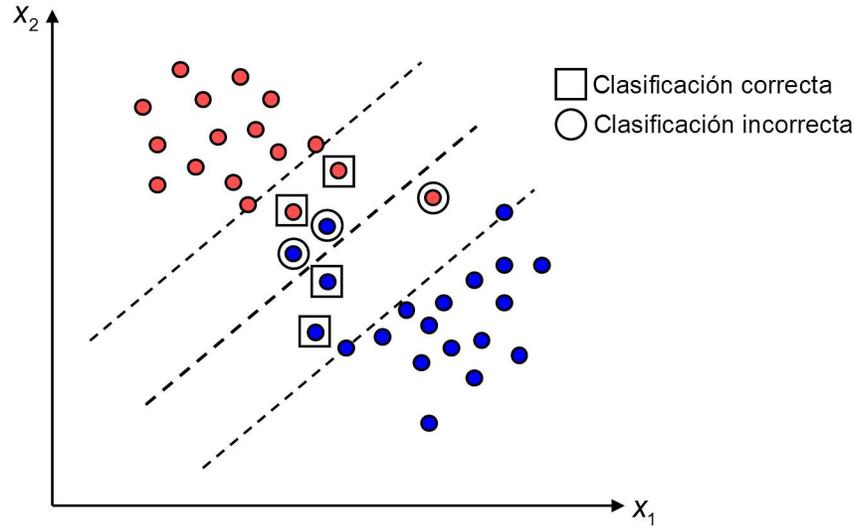


Figura B.3: Ejemplo de clases no separables con puntos que caen dentro del margen de separación.

### B.2.2. Datos no separables

En el caso de que las regiones correspondientes a las clases no sean linealmente separables, tal como se muestra en la figura B.3, los vectores de entrenamiento pueden pertenecer a una de las tres siguientes categorías:

- Vectores que caen fuera del margen de separación y son clasificados correctamente. Estos vectores cumplen la condición:

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \quad (\text{B.23})$$

- Vectores que caen dentro del margen de separación y son clasificados correctamente. Estos vectores, enmarcados con un rectángulo en la Figura B.3, cumplen:

$$0 \leq y_i (\mathbf{w}^T \mathbf{x}_i + b) < 1 \quad (\text{B.24})$$

- Vectores que son clasificados incorrectamente. Estos son enmarcados con un círculo en la figura B.3 y cumplen la siguiente inecuación:

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) < 0 \quad (\text{B.25})$$

A partir de la introducción de una variable laxa  $\xi_i$  para cada vector  $\mathbf{x}_i$ , las tres situaciones anteriores pueden ser recogidas en una única expresión como:

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i. \quad (\text{B.26})$$

Las instancias de la primera categoría se corresponden con  $\xi_i = 0$ , las de la segunda categoría con  $0 < \xi_i \leq 1$  y las de la tercera con  $\xi_i > 1$ . La búsqueda del hiperplano óptimo busca ahora, además de mantener el máximo margen de separación, minimizar el número de puntos con  $\xi_i > 0$ .

Al incorporar el coste de estos errores, la función objetivo a minimizar pasa de ser B.10 a venir dada como:

$$\begin{aligned} \text{minimizar} \quad J(\mathbf{w}) &\equiv \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i & (\text{B.27}) \\ \text{sujeto a:} \quad y_i (\mathbf{w}^T \mathbf{x}_i + b) &\geq 1 - \xi_i, \quad \xi_i \geq 0. \end{aligned}$$

Aquí el parámetro  $C$  es un parámetro de coste a elegir para cada conjunto de datos, de forma que cuanto mayor sea su valor, más se penalizarán los errores. La forma de encontrar la solución a este problema es análoga a la presentada en el caso separable y queda reflejada en la ecuación B.19, aunque en este caso los valores de los  $\alpha_i$  tendrán un límite superior en  $C$ . La función de Lagrange queda ahora:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) \equiv \frac{\|\mathbf{w}\|^2}{2} + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \mu_i \xi_i - \sum_{i=1}^N \alpha_i [y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1 + \xi_i], \quad (\text{B.28})$$

donde  $\mu_i$  son los multiplicadores de Lagrange asociados a la restricción de que todos los  $\xi_i$  sean mayores o iguales que 0. Al igualar las derivadas parciales de  $\mathbf{w}$ ,  $b$  y  $\xi_i$  a cero, se llega a:

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i \quad (\text{B.29})$$

$$0 = \sum_{i=1}^N \alpha_i y_i \quad (\text{B.30})$$

$$0 = C - \alpha_i - \mu_i. \quad (\text{B.31})$$

Sustituyendo B.29, B.30 y B.31 en B.28, queda la forma dual para el caso no linealmente separable:

$$L_D(\mathbf{w}, b, \boldsymbol{\alpha}) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (y_i y_j \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j), \quad (\text{B.32})$$

lo que parece igual que para el caso separable B.19. La diferencia es que la restricción B.31 unida a que las  $\mu_i$  son mayores o iguales que 0, da lugar a que en el caso no separable

habrá de verificarse  $0 \leq \alpha_i \leq C$ . Es decir, ahora el valor de  $\alpha_i$  está acotado también superiormente. Esto se basa en que el Lagrangiano para el caso no separable ha de minimizarse respecto a  $\mathbf{w}$  y  $b$ , pero a la vez ha de maximizarse respecto a  $\alpha_i$ .

La solución del vector  $\boldsymbol{\alpha}$  clasifica los patrones de entrada en tres tipos diferentes:

1. Aquellos con  $\alpha_i = 0$ . Estos patrones son correctamente clasificados y no aportan información para la obtención del hiperplano de separación.
2. Si  $0 < \alpha_i < C$ , los patrones correspondientes están localizados en los márgenes máximos. Estos aportan información y son conocidos como vectores soporte tipo 1.
3. Con  $\alpha_i = C$  los patrones correspondientes o bien están mal clasificados o caen dentro del margen. También son vectores soporte y se les conoce como de tipo 2.

Todos los vectores con  $\alpha_i \neq 0$  formarán parte de la frontera de decisión que tendrá la misma forma que la presentada en la ecuación B.22.

### B.2.3. Datos no separables linealmente

En datos de aplicaciones reales las clases no siempre son separables linealmente. De hecho, las funciones lineales presentan dos principales limitaciones:

- Las funciones con las que se intenta establecer el aprendizaje no tienen siempre una representación simple y por tanto, pueden ser difícilmente verificables.
- Los datos de entrenamiento son, por lo general, ruidosos y no hay garantía de que exista una función que clasifique correctamente los datos de entrenamiento.

A pesar de que a veces es imposible obtener un hiperplano de separación en el espacio de entrada, existe la posibilidad de conseguirlo en otro espacio. La clave es mapear los datos de entrada a otro espacio, generalmente de mayores dimensiones, donde la clasificación podría realizarse de forma lineal. El problema de la alta dimensionalidad es que la carga computacional es muy elevada, incluso a veces inabordable. Para evitar este problema, se usa el ‘truco’ del kernel y el producto escalar es usado como medida de similitud. A cambio de realizar el mapeado del espacio de entrada y calcular los productos escalares en el nuevo espacio, es posible usar las funciones kernel que combinan ambos pasos.

El espacio  $X$  es referido como el espacio de entrada, mientras que  $F = \{\phi(\mathbf{x}) : \mathbf{x} \in X\}$  es denominado el espacio de características. Seleccionando un kernel adecuado, se realiza un mapeado no-lineal entre el espacio de entrada y el espacio de características. Esto significa que cada vector en el espacio de entrada se corresponde con un vector en el espacio de características.

La figura B.4 muestra un ejemplo de mapeado de un espacio de entrada bidimensional a otro espacio de características tridimensional. En el espacio original de entrada los datos no pueden ser separados mediante una función lineal. Sin embargo, un mapeado

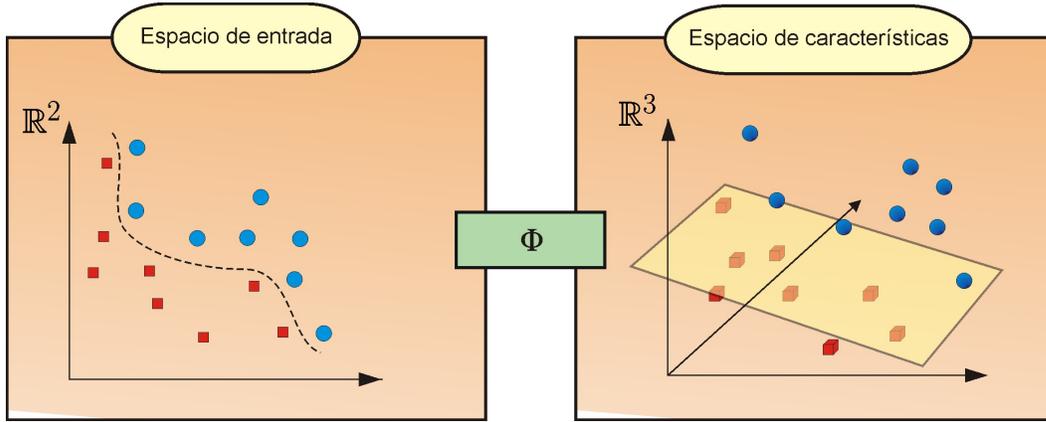


Figura B.4: Ejemplo de transformación de espacios  $\mathbb{R}^2 \rightarrow \mathbb{R}^3$ .

de características puede simplificar la tarea de clasificación debido a que los datos en el nuevo espacio de características son linealmente separables.

La función de decisión no lineal en el espacio de entrada es equivalente a una función lineal en el espacio de características y por tanto, la función B.5 utilizada en el aprendizaje lineal ahora es modificada del siguiente modo:

$$f(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b = \sum_{i=1}^N w_i \phi(\mathbf{x}) + b \quad (\text{B.33})$$

donde  $\phi(\mathbf{x}) : X \rightarrow F$  es la función de mapeado del espacio de entrada al espacio de características. La función de decisión puede ser expresada como una combinación lineal de los puntos de entrenamiento para que la regla de decisión pueda ser evaluada como el producto escalar entre los puntos de entrenamiento y los puntos a testear como:

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i y_i \phi(\mathbf{x}_i)^T \phi(\mathbf{x}) + b. \quad (\text{B.34})$$

En cualquier caso, es posible construir una máquina no lineal en la que el producto escalar  $\phi(\mathbf{x}_i)^T \phi(\mathbf{x})$  puede ser calculado directamente en el espacio de características como una función de los vectores originales de entrada.

#### B.2.4. Kernels

Un kernel es una función tal que para todo  $\mathbf{x}$  y  $\mathbf{z} \in X$  cumple:

$$K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^T \phi(\mathbf{z}), \quad (\text{B.35})$$

donde  $\phi$  es un mapeado de  $X$  a un espacio de características  $F$ . Una función kernel puede ser interpretada como una medida de similitud entre dos muestras de entrada. La regla de decisión ahora es evaluada como:

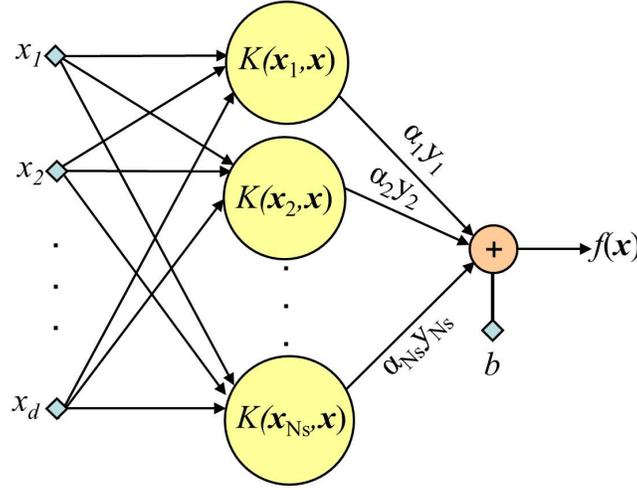


Figura B.5: Arquitectura SVM empleando funciones kernel.

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b = \sum_{i=1}^{N_s} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b. \quad (\text{B.36})$$

Entre los kernels típicos de aplicación en reconocimiento de patrones, se encuentran los siguientes:

- Lineal:

$$K(\mathbf{x}, \mathbf{z}) = \mathbf{x}^T \mathbf{z} \quad (\text{B.37})$$

- Polinómico:

$$K(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^T \mathbf{z})^d \quad (\text{B.38})$$

- De base radial (RBF-*Radial Basis Functions*):

$$K(\mathbf{x}, \mathbf{z}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{z}\|^2}{\sigma^2}\right) \quad (\text{B.39})$$

- Sigmoidal: es la función común de activación en redes neuronales. Una SVM con un kernel sigmoidal es equivalente a un perceptrón de dos capas.

$$K(\mathbf{x}, \mathbf{z}) = \tanh(\gamma(\mathbf{x}^T \mathbf{z}) + r) \quad (\text{B.40})$$

La figura B.5 muestra la arquitectura correspondiente a la regla de decisión asociada a B.36, donde el número de nodos es determinado por el número de vectores soporte. Los nodos simbolizan el producto escalar entre el mapeado del vector a testear  $\mathbf{x}$  y el mapeado de los vectores soporte en el espacio de características a través del kernel.

Llegados a este punto, es interesante analizar las propiedades que debe satisfacer una función de similitud para comportarse como una función kernel. Para asegurar que la función kernel sea útil en el espacio de características, ésta debe ser simétrica:

$$K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^T \phi(\mathbf{z}) = \phi(\mathbf{z})^T \phi(\mathbf{x}) = K(\mathbf{z}, \mathbf{x}) \quad (\text{B.41})$$

y satisfacer la desigualdad de Cauchy-Schwartz:

$$K(\mathbf{x}, \mathbf{z})^2 = (\phi(\mathbf{x})^T \phi(\mathbf{z}))^2 \leq \|\phi(\mathbf{x})\|^2 \|\phi(\mathbf{z})\|^2 \quad (\text{B.42})$$

con

$$K(\mathbf{x}, \mathbf{z})^2 = \phi(\mathbf{x})^T \phi(\mathbf{x}) \phi(\mathbf{z})^T \phi(\mathbf{z}) = K(\mathbf{x}, \mathbf{x}) K(\mathbf{z}, \mathbf{z}). \quad (\text{B.43})$$

# APÉNDICE C

## Catálogo Oficial de Señales de Tráfico de España



Figura C.1: Señales de advertencia de peligro

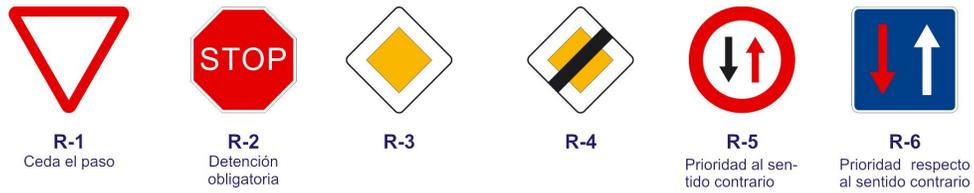


Figura C.2: Señales de reglamentación - Prioridad.



Figura C.3: Señales de reglamentación - Prohibición de entrada.



Figura C.4: Señales de reglamentación - Otras prohibiciones o restricciones.



Figura C.5: Señales de reglamentación - Señales de obligación.



Figura C.6: Señales de reglamentación - Fin de prohibición.



Figura C 7: Señales de indicación - Indicaciones generales

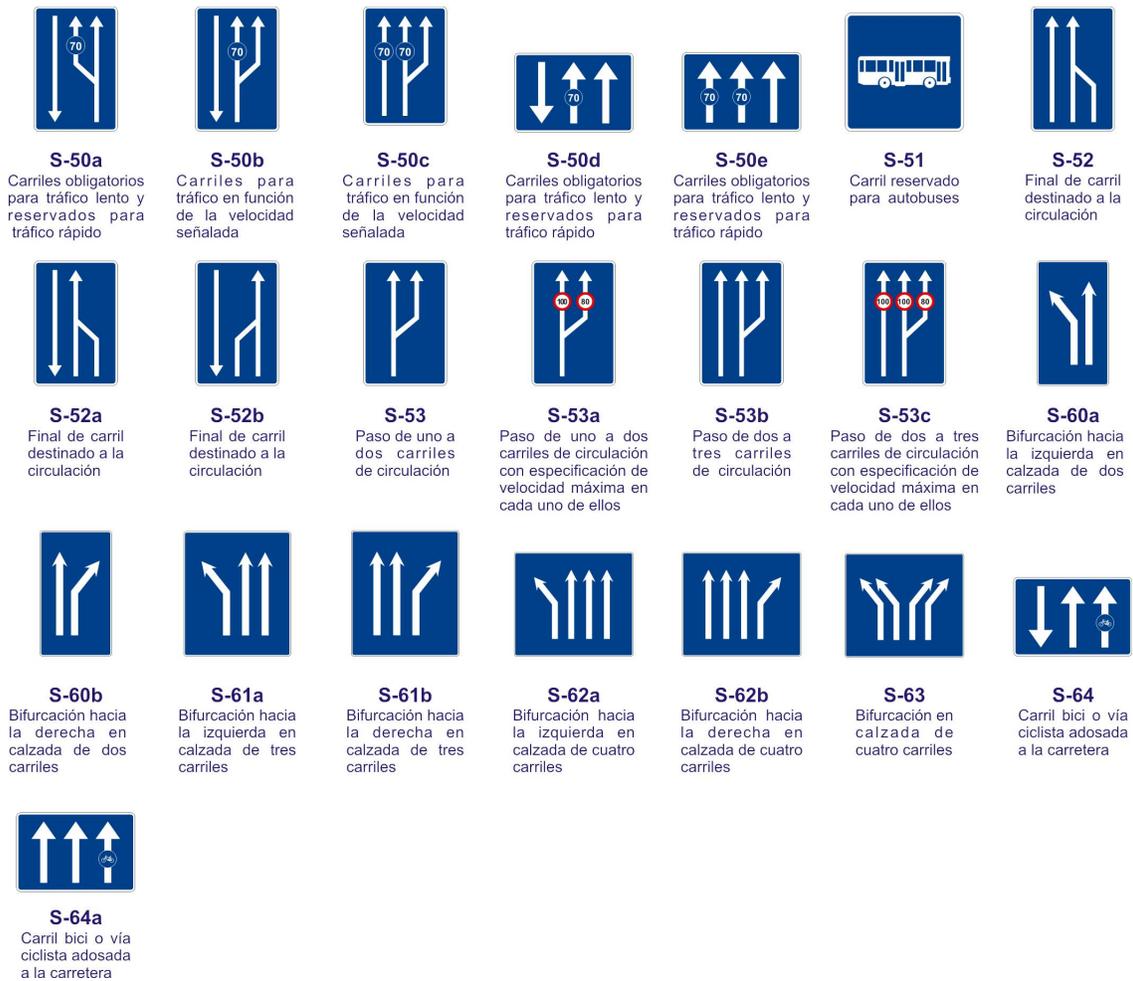


Figura C.8: Señales de indicación - Carriles.

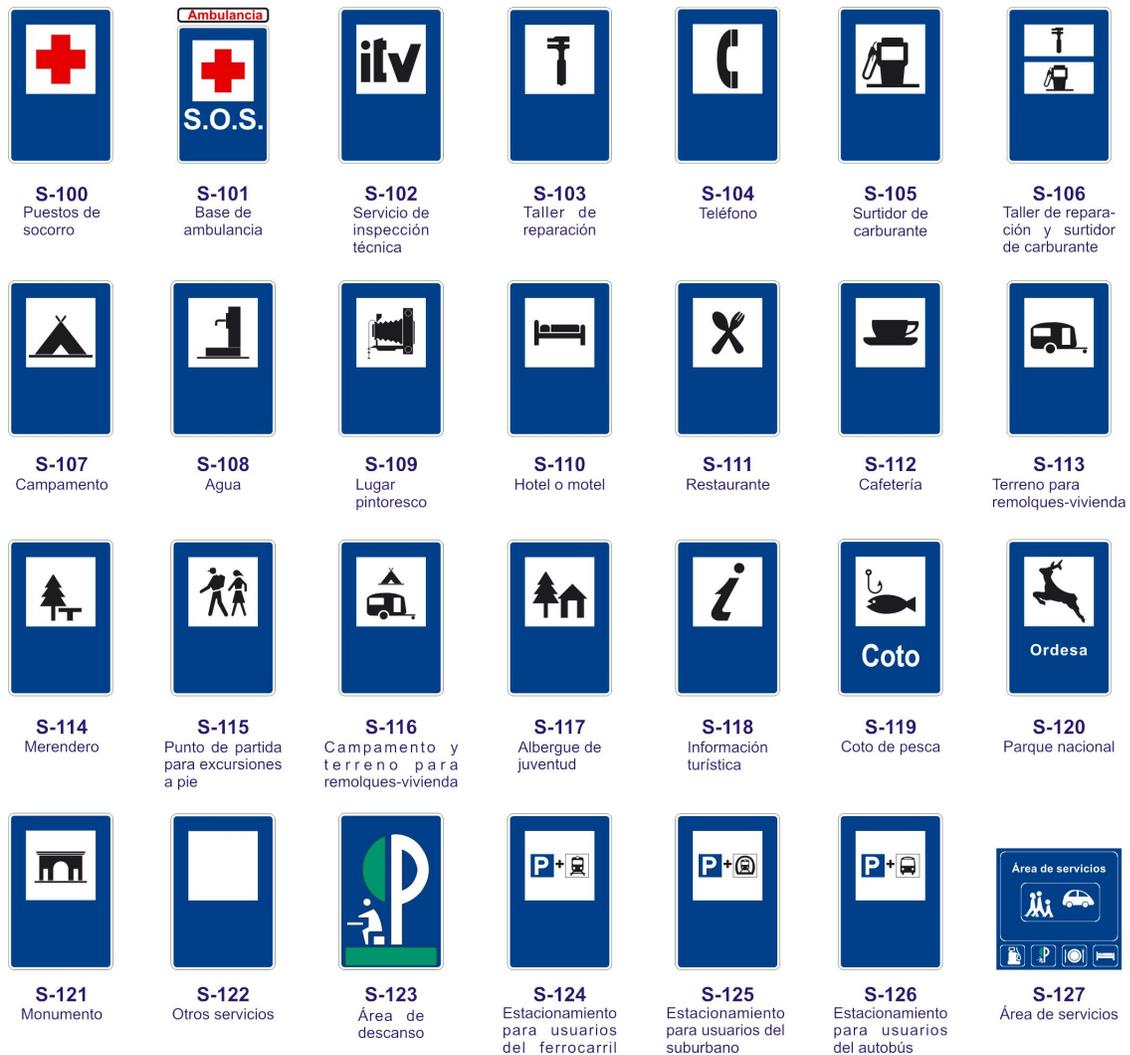


Figura C.9: Señales de indicación - Servicio.

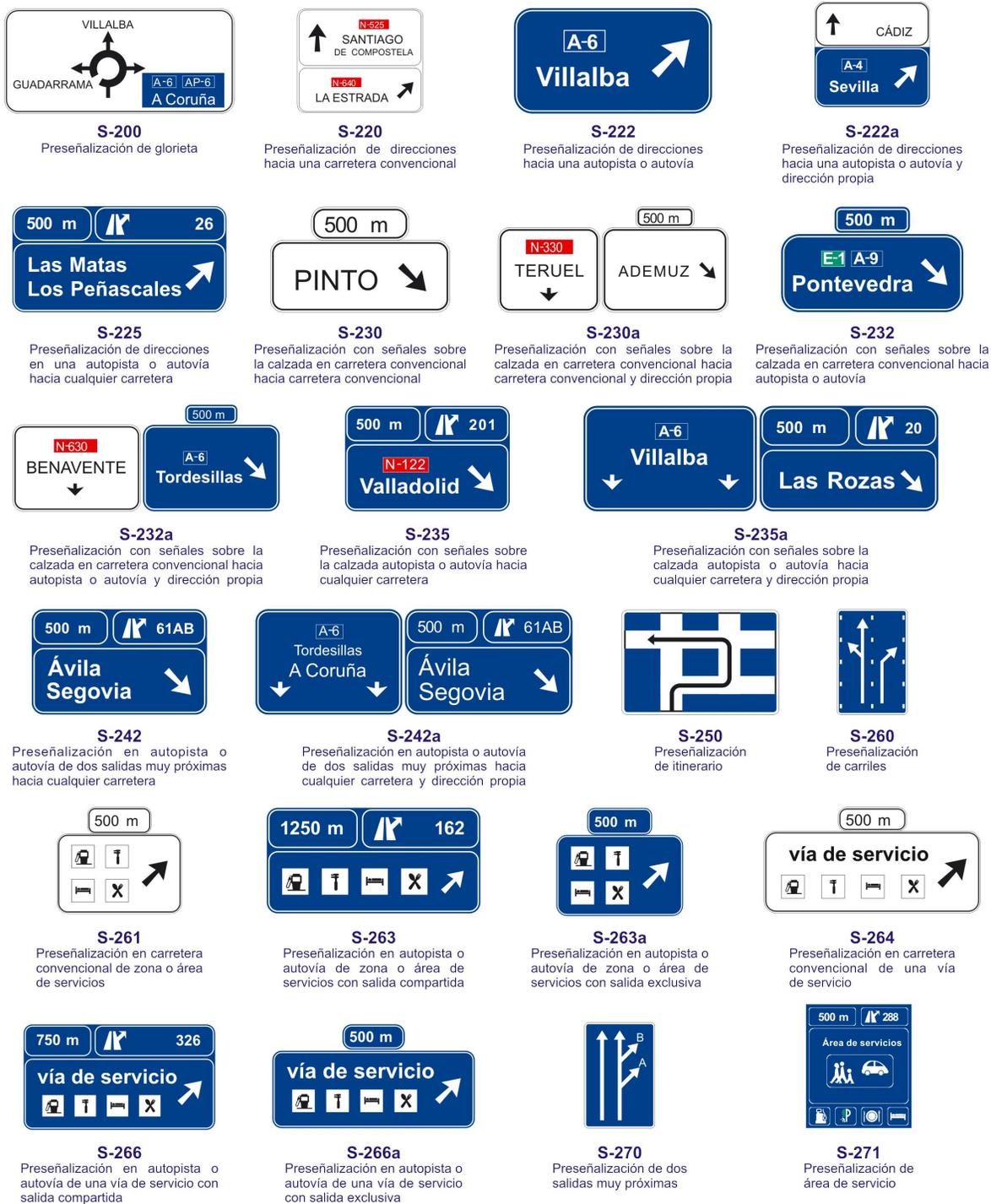


Figura C.10: Señales de orientación - Preseñalización.

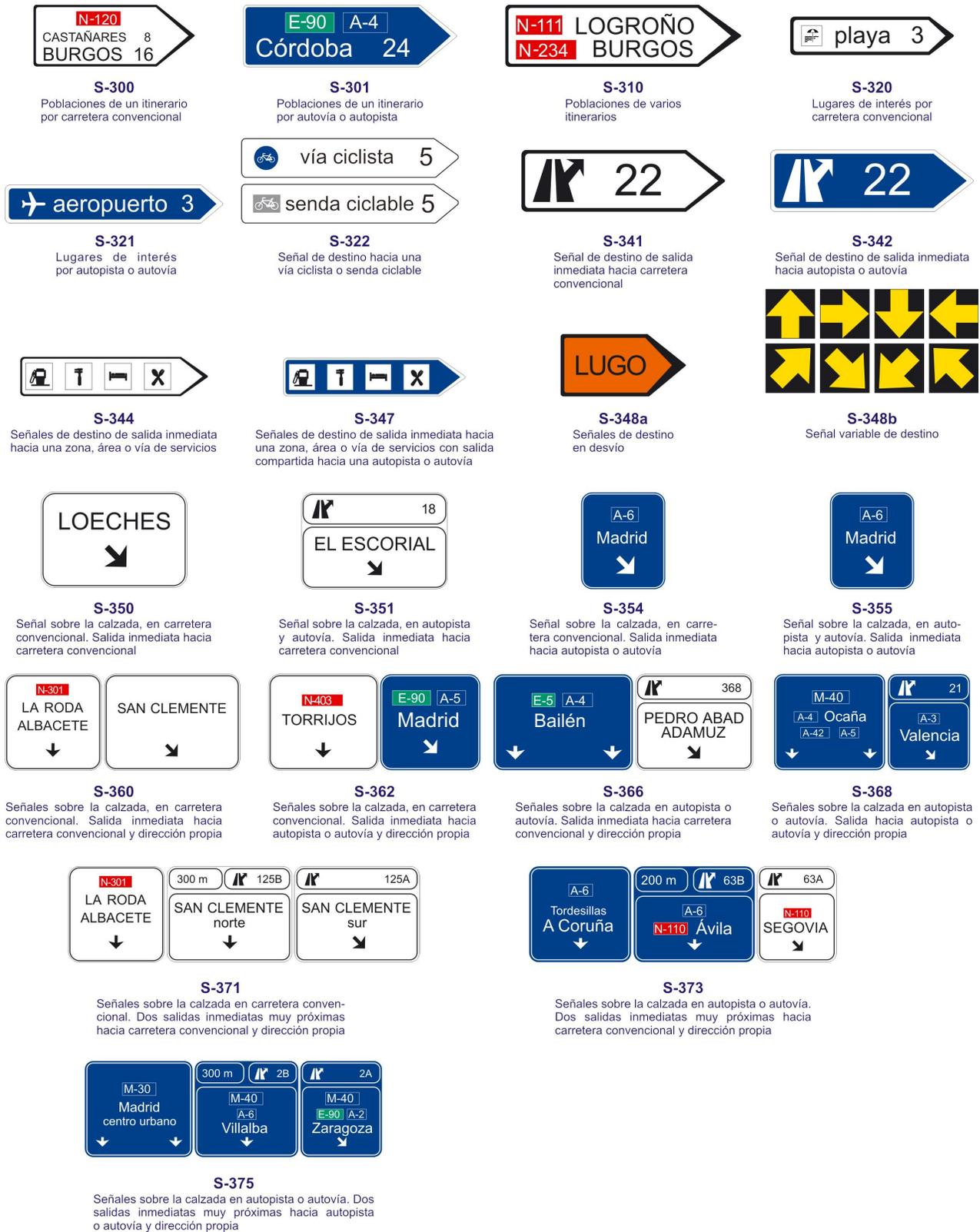


Figura C.11: Señales de orientación - Dirección.



Figura C.12: Señales de orientación - Localización.



Figura C.13: Señales de orientación - Confirmación.



Figura C.14: Señales de orientación - Uso específico de poblado.

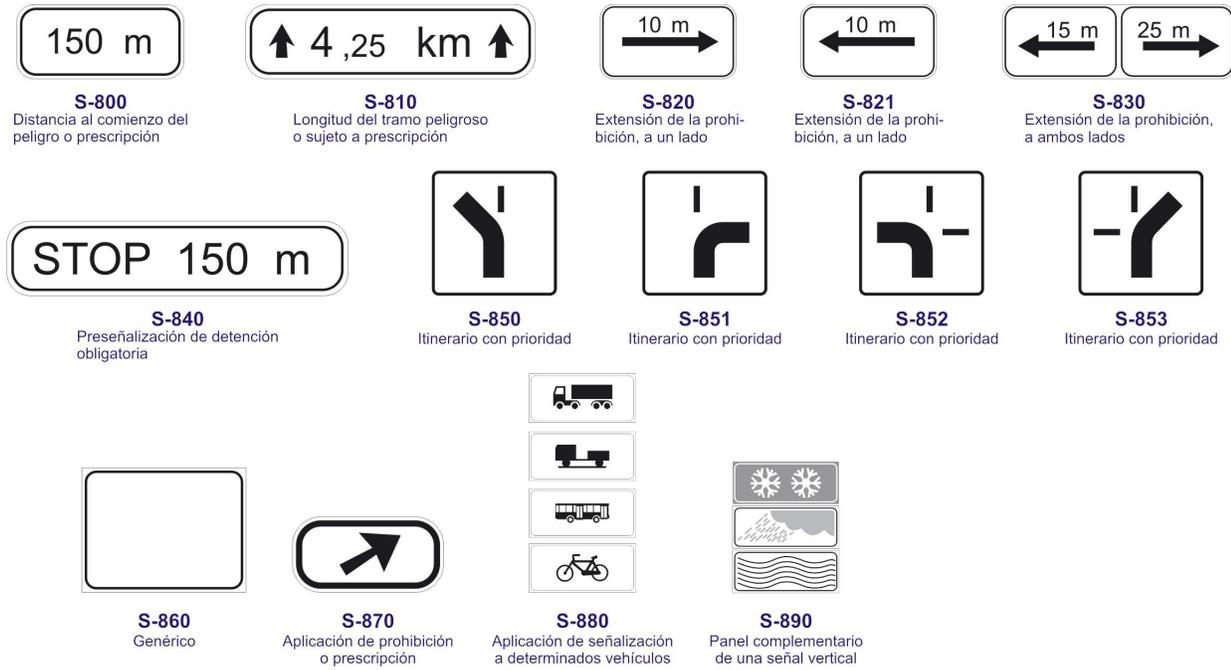


Figura C.15: Señales de orientación - Paneles complementarios.

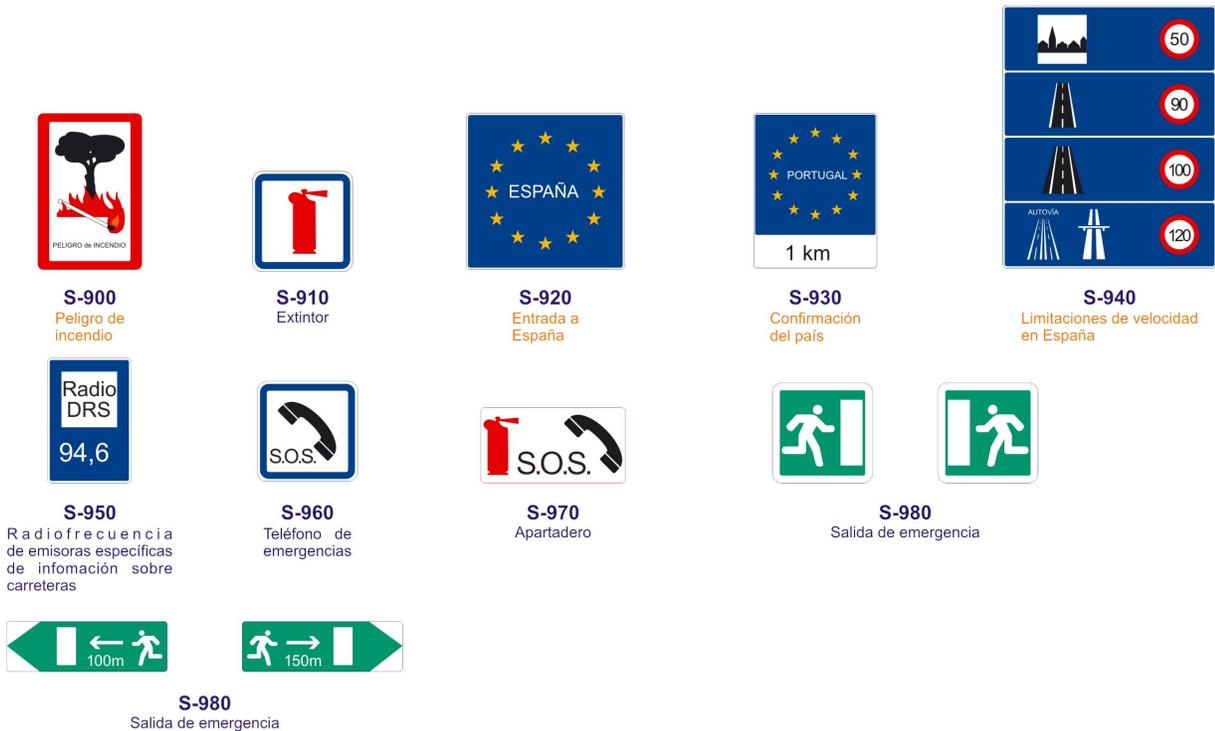


Figura C.16: Señales de orientación - Otras señales.



Figura C.17: Señales de obras - Advertencia de peligro.



Figura C.18: Señales de obras - Prohibiciones.

# Bibliografía

- [Adorni96] ADORNI, G., V. D'ANDREA, G. DESTRI, y M. MORDONINI. «Shape searching in real world images: a CNN-based approach.» En *Proc. of IV IEEE Inter. Workshop on Cellular Neural Networks and Their Applications*. 1996.
- [Akatsuka87] AKATSUKA, H., y S. IMAI. «Road sign posts recognition systems.» En *Proc. of the SAY vehicle highway infrastructure: safety compatibility*. 1987, 189–196.
- [Allwein02] ALLWEIN, E.L., R.E. SCHAPIRE, y Y. SINGER. «Reducing multiclass to binary: a unifying approach for margin classifiers.» *Journal of Machine Learning Research*, 1, (2002), 113–141.
- [Aoyagi96] AOYAGI, Y., y T. ASAKURA. «A study on traffic sign recognition in scene image using genetic algorithms and neural networks.» En *Proc. of the 22nd. IEEE Int. Conf. Industrial Electronics, Control and Instrumentation*. Taipeh, Taiwan, 1996, tomo 3, 1838–1843.
- [Arnoul96] ARNOUL, P., M. VIALA, J.P. GUERIN, y M. MERGY. «Traffic signs localisation for highways inventory from a video camera on board a moving collection van.» En *Proc. of IEEE Intelligent Vehicles Symp.* Tokyo, Japan, 1996, 141–146.
- [Bahlmann05] BAHLMANN, C., Y. ZHU, V. RAMESH, M. PELLKOFER, y T. KOEHLER. «A System for Traffic Sign Detection, Tracking, and Recognition Using Color, Shape, and Motion Information.» En *Proc. of IEEE Intelligent Vehicles Symp.* Las Vegas, USA, 2005, 255–260.
- [Bahlmann04] —. «A System for Traffic Sign Detection, Tracking, and Recognition Using Color, Shape, and Motion Information.» En *Proc. of IEEE Intelligent Vehicles Symp.* Las Vegas, USA, 2005, 607–612.
- [Bargeton08] BARGETON, A., F. MOUTARDE, F. NASHASHIBI, y B. BRADAI. «Improving pan-european speedlimit signs recognition with a new global

- number segmentation before digit recognition.» En *Proc. of IEEE Intelligent Vehicles Symp.* 2008.
- [Barnes08] BARNES, N., A. ZELINSKY, y L.S. FLETCHER. «Real-Time Speed Sign Detection Using the Radial Symmetry Detector.» *IEEE Transactions on Intelligent Transportation Systems*, 9, n<sup>o</sup> 2, (2008), 322–332.
- [Baro09] BARÓ, X., S. ESCALERA, J. VITRIA, O. PUJOL, y P. RADEVA. «Traffic Sign Recognition using Evolutionary Adaboost detection and Forest-ECOC classification.» *IEEE Trans. on Intelligent Transportation Systems*, 10, n<sup>o</sup> 1, (2009), 113–126.
- [Besserer93] BESSERER, B., S. ESTABLE, y B. ULMER. «Multiple knowledge sources and evidential reasoning for shape recognition.» En *Proc. of Int. Conference on Computer Vision*. 1993, 624–631.
- [Betke94] BETKE, M., y N. MAKRIS. «Fast object recognition in noisy images using simulated annealing.» *Inf. téc.*, Massachusetts Institute of Technology, 1994.
- [Bishop95] BISHOP, C.M. *Neural networks for pattern recognition*. 1995.
- [Blancard92] BLANCARD, M. «Road sign recognition: A study of vision-based decision making for road environment recognition.» *Vision Based Vehicle Guidance, Germany: Springer-Verlag*, (1992), 162–175.
- [Boe03] BOE. «Boletín Oficial del Estado, 23 de diciembre de 2003. Real Decreto 1428/2003. Número 306.»
- [Boe06] —. «Boletín Oficial del Estado, 5 de septiembre de 2006. Real Decreto 965/2006. Número 212.»
- [Bolshakova03] BOLSHAKOVA, N., y F. AZUAJE. «Cluster Validation Techniques for Genome Expression Data.» *Signal Processing*, 83, n<sup>o</sup> 4, (2003), 825–833.
- [Bradley52] BRADLEY, R. A., y Milton E. TERRY. «Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons.» *Biometrika*, 39, n<sup>o</sup> 3/4, (1952), 324–345.
- [Broggi07] BROGGI, A., P. CERRI, P. MEDICI, P. P. PORTA, y G. GHISIO. «Real Time Road Signs Recognition.» En *Proc. of IEEE Intelligent Vehicles Symp.* Istanbul, Turkey, 2007, 981–986.

- [Buker95] BUKER, U., y B. MERTSCHING. «Parallel evaluation of hierarchical image databases.» *Journal of Parallel and Distributed Computing*, 31, nº 2, (1995), 141–152.
- [Busin08] BUSIN, L., N. VANDENBROUCKE, y L. MACAIRE. «Color spaces and image segmentation.» En *Advances in Imaging and Electron Physics*. Elsevier Inc., Orlando, FL, USA, 2008, tomo 151, 65–168.
- [Chang00] CHANG, C.C., y C.J. LIN. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [CIE89] CIE. «International Lighting Vocabulary. CIE Publications 17.4.» *Inf. téc.*, Commission International de L’Eclairage, 4th edition, 1989.
- [Ciresan11] CIRESAN, D., U. MEIER, J. MASCI, y J. SCHMIDHUBER.
- [Crammer01] CRAMMER, K., y Y. SINGER. «On The Algorithmic Implementation of Multiclass Kernel-based Vector Machines.» *Machine Learning Research*, 2, (2001), 265–292.
- [Dalal05] DALAL, N., y B. TRIGGS. «Histograms of oriented gradients for human detection.» En *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*. 2005, 886–893.
- [Davis06] DAVIS, J., y M. GOADRICH. «The relationship between precision-recall and ROC curves.» En *Proc. International Conference on Machine Learning (ICML)*. 2006, 233–240.
- [Demsar06] DEMSAR, J. «Statistical Comparisons of Classifiers over Multiple Data Sets.» *Journal of Machine Learning Research*, 7, (2006), 1–30.
- [Dietterich95] DIETTERICH, T.G., y G. BAKIRI. «Solving multiclass learning problems via error-correcting output codes.» *Journal Artificial Intelligence Research*, 2, (1995), 263–286.
- [Duin05] DUIN, R.P., y E. PEKALSKA. «Open Issues in Pattern Recognition.» *Computer Recognition Systems*, (2005), 27–42.
- [Ebner04] EBNER, M. «A parallel algorithm for color constancy.» *Journal Parallel and Distributed Computing*, 64, (2004), 79–88.
- [Edwards48] EDWARDS, A. «Note on the correction for continuity in testing the significance of the difference between correlated proportions.» *Psychometrika*, 13, (1948), 185–187.

- [Eichner08] EICHNER, M., y T. P. BRECKON. «Integrated speed limit detection and recognition from real-time video.» En *Proc. IEEE Intelligent Vehicles Symposium*. Eindhoven, Netherlands, 2008.
- [Escalera03] DE LA ESCALERA, A., J. M. ARMINGOL, y M. MATA. «Traffic sign recognition and analysis for intelligent vehicles.» *Image and Vision Computing*, 21, (2003), 247–258.
- [Escalera04] DE LA ESCALERA, A., J. M. ARMINGOL, J. M. PASTOR, y F. J. RODRÍGUEZ. «Visual sign information extraction and identification by deformable models for intelligent vehicles.» *IEEE Trans. on Intelligent Transportation Systems*, 15, n<sup>o</sup> 2, (2004), 57–68.
- [Escalera97] DE LA ESCALERA, A., L. E. MORENO, M. A. SALICHS, y J. M. ARMINGOL. «Road traffic sign detection and classification.» *IEEE Trans. on industrial electronics*, 44, n<sup>o</sup> 6, (1997), 848–859.
- [Escalera10] ESCALERA, S., O. PUJOL, y P. RADEVA. «On the Decoding Process in Ternary Error-Correcting Output Codes.» *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32, n<sup>o</sup> 1, (2010), 120–134.
- [Estable94] ESTABLE, S., J. SCHICK, F. STEIN, R. JANSSEN, R. OTT, W. RITTER, y Y.-J. ZHENG. «A real-time traffic sign recognition system.» En *Proc. of the Intelligent Vehicles '94 Symposium*. 1994, 213–218.
- [Fairchild01] FAIRCHILD, M. D. «Status of Cie color appearance models.» *Technical report*, Rochester Institute of Technology, 2001.
- [Fang03-b] FANG, C. Y., C. S. FUH, S. W. CHEN, y P. S. YEN. «A road sign recognition system based on dynamic visual model.» En *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. CA, USA, 2003, tomo 1.
- [Fleyeh06] FLEYEH, H. «Shadow and highlight invariant colour segmentation algorithm for traffic signs.» En *Proc. IEEE Conf. on Cybernetics and Intelligent Systems*. 2006.
- [BOE00] DE FOMENTO, Ministerio. «Orden 1798, Boletín Oficial del Estado.» , n<sup>o</sup> 25, (2000), 4049–4106.
- [Freeman91] FREEMAN, W., y E. ADELSON. «The design and use of steerable filters.» *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13, n<sup>o</sup> 9, (1991), 891–906.

- [Gao13] GAO, H., Z. LIU, y Y. JIANG. «Traffic Sign Recognition Technology Based on BOW Model.» *Journal of Software*, 8, nº 10, (2013), 2652–2659.
- [Gao06] GAO, Podladchikova L. Shaposhnikov D. Hong K., X. W., y N. SHEVTSOVA. «Recognition of traffic signs based on their colour and shape features extracted using human vision models.» *Journal of Visual Communication and Image Representation*, 17, nº 4, (2006), 675–685.
- [Garcia06] GARCÍA-GARRIDO, M., M. SOTELO, y E. MARTÍN-GOROSTIZA. «Fast traffic sign detection and recognition under changing lighting conditions.» En *Proc. IEEE Intelligent Transportation Systems*. Toronto, Canada, 2006, 811–816.
- [Gavrila98] GAVRILA, D. M., y V. PHILOMIN. «Real-time object detection using distance transform.» En *Proc. IEEE Intelligent Vehicles Symposium*. Stuttgart, Germany, 1998, 274–279.
- [Gevers99] GEVERS, T., y A.W.M. SMEULDERS. «Color based object recognition.» *Pattern Recognition*, 32, (1999), 453–464.
- [Ghica95] GHICA, D., S. W. LU, y X. YUAN. «Recognition of traffic signs by artificial neural network.» En *Proc. IEEE Int. Conf. on Neural Networks*. 1995, tomo 3, 1444–1449.
- [Gil08] GIL-JIMÉNEZ, P., S. MALDONADO-BASCÓN, H. GÓMEZ-MORENO, S. LAFUENTE-ARROYO, y F. LÓPEZ-FERRERAS. «Traffic sign shape classification and localization based on the normalized FFT of the signature of blobs and 2D homographies.» *Signal Processing*, 88, nº 12, (2008), 2943–2955.
- [Gomez10] GÓMEZ-MORENO, H., S. MALDONADO-BASCÓN, P. GIL-JIMÉNEZ, y S. LAFUENTE-ARROYO. «Goal evaluation of segmentation algorithms for traffic sign recognition.» *IEEE Intelligent Transportation Systems*, 11, nº 4, (2010), 917–930.
- [Gonzalez93] GONZÁLEZ, R.C., y R.E. WOODS. *Digital Image Proccesing*. Addison-Wesley, 1993.
- [Grigorescu03] GRIGORESCU, C., y N. PETKOV. «Distance sets for shape filters and shape recognition.» *IEEE Transactions on Image Processing*, 12, nº 10, (2003), 1274–1286.
- [Guo10] GUO, Zhenhua, L. ZHANG, D. ZHANG, y X. MOU. «Hierarchical multiscale LBP for face and palmprint recognition.» En *Proc. of the 17th IEEE*

- International Conference on Image Processing (ICIP)*. Hong Kong, 2010, 4521 – 4524.
- [Hanowski06] HANOWSKI, R. J., R. L. OLSON, J. S. HICKMAN, y T. A. DINGUS. «The 100-Car Naturalistic Driving Study: A Descriptive Analysis of Light Vehicle-Heavy Vehicle Interactions from the Light Vehicle Driver's Perspective, Data Analysis Results.» *Report data*, Sponsoring Agency Name and Address National Highway Traffic Safety Admin. (NHTSA), Marzo 2006.
- [Hibi96] HIBI, T. «Vision based extraction and recognition of road sign region from natural color image, by using HSL and coordinates transformation.» En *Proc. of the 29th International Symposium on Automotive Technology and Automation, Robotics, Motion and machine vision in the automotive industries*. Florence, Italy, 1996.
- [Houben13] HOUBEN, S., J. STALLKAMP, J. SALMEN, M. SCHLIPSING, y C. IGEL. «Detection of traffic signs in real-world images: the German Traffic Sign Detection Benchmark.» En *Proc. of the IEEE International Joint Conference on Neural Networks (IJCNN)*. Dallas, USA, 2013.
- [Hsien06] HSIEN, J. C., Y. S. LIOU, y S. Y. CHEN. «Road sign detection and recognition using hidden Markov model.» *Asian Journal of Health and Information Sciences*, 1, n<sup>o</sup> 1, (2006), 85–100.
- [Hsu02] HSU, C.-W., y C.-J. LIN. «A comparison of methods for multi-class support vector machines.» *IEEE Trans. on Neural Networks*, 13, n<sup>o</sup> 2, (2002), 415–425.
- [Huang00] HUANG, C. L., y S.H. HSU. «Road sign interpretation using matching pursuit method.» En *Proc. of the 4th IEEE Southwest Symposium on Image Analysis and Interpretation*. Washington, DC, USA, 2000.
- [Huang08] HUANG, H., C. CHEN, Y. JIA, y S. TANG. «Automatic Detection and Recognition of Circular Road Sign.» En *IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA*. Beijing, 2008, 626–630.
- [Janssen93] JANSSEN, R., W. RITTER, F. STEIN, y S. OTT. «Hybrid approach for traffic sign recognition.» En *Proc. IEEE Intelligent Vehicles Symposium*. 1993, 390–395.
- [Jost11] JOST, G., R. ALLSOP, M. STERLU, y M. POPOLIZIU. «2010 Road Safety Target Outcome: 100,000 fewer deaths since 2001.» *5th road safety pin report*, European Transport Safety Council, Junio 2011.

- [Judd64] JUDD, D., D. MACADAM, y G. WYSZECKI. «Spectral distribution of typical daylight as a function of correlated color temperature.» *Journal of the Optical Society of America*, 54, nº 8, (1964), 1031–1040.
- [Kalata84] KALATA, P. R. «The tracking index: A generalized parameter for  $\alpha - \beta$  and  $\alpha - \beta - \gamma$  target trackers.» *IEEE Transactions on Aerospace and Electronics Systems*, 20, nº 2, (1984), 174–182.
- [Kalman60] KALMAN, R.E. «A New Approach to Linear Filtering and Prediction Problems.» *IEEE Transactions of the ASME-Journal of Basic Engineering*, 82, (1960), 35–45.
- [Kamada90] KAMADA, H., S. NAOI, y T. GOTOH. «A compact navigation system using image processing and fuzzy control.» En *Proc. Southeastcon*. New Orleans, 1990, tomo 1, 337–342.
- [Kang94] KANG, D.S., N.C. GRISWOLD, y N. KEHTARNAVAZ. «An invariant traffic sign recognition system based on sequential color processing and geometrical transformation.» En *Proc. of the IEEE Southwest Symposium on Image Analysis and Interpretation*. Dallas, TX, 1994, 88–93.
- [Kehtarnavaz93] KEHTARNAVAZ, N., N. C. GRISWOLD, y D. S. KANG. «Stop-sign recognition based on color/shape processing.» *Machine Vision and Applications*, 6, nº 4, (1993), 206–208.
- [Kellmeyer94] KELLMEYER, D., y H. ZWAHLEN. «Detection of highway warning signs in natural video images using color image processing and neural networks.» En *Proc. of the IEEE Int. Conf. on Neural Networks*. Orlando, FL, USA, 1994, 4226 – 4231.
- [Kim97] KIM, S.K., y D.A. FORSYTH. «A new approach for road sign detection and recognition algorithm.» En *Proc. of the 30th International Symposium on Automotive Technology and Automation*. Florence, Italy, 1997, 171–178.
- [Koenderik87] KOENDERINK, J., y A. VAN DOORN. «Representation of local geometry in the visual system.» *Biological Cybernetics*, 55, (1987), 367–375.
- [Kouzani07] KOUZANI, A. Z. «Road-Sign Identification Using Ensemble Learning.» En *Proc. of the IEEE Intelligent Vehicles Symposium*. Istanbul, Turkey, 2007.
- [Kus08] KUS, M. C., M. GOKMEN, y S. ETANER-UYAR. «Traffic sign recognition using Scale Invariant Feature Transform and color classification.» En *Proc. of the IEEE 23rd International Symposium on Computer and Information Sciences*. Istanbul, Turkey, 2008.

- [Lalonde95b] LALONDE, M., y Y. LI. «Detection of road signs using color indexing.» *Inf. téc.*, Centre de recherche informatique de Montréal, 1995.
- [Lalonde95] —. «Road sign recognition. Survey of the state of the art for subproject 2.4.» *Inf. téc.*, Centre de recherche informatique de Montréal, 1995.
- [Larsson11] LARSSON, F., y M. FELSBERG. «Using Fourier descriptors and spatial models for traffic sign recognition.» En *Proc. of the 17th Scandinavian Conference on Image Analysis*. Ystad, Sweden, 2011, 238–249.
- [Lazebnik03] LAZEBNIK, S., C. SCHMID, y J. PONCE. «Sparse texture representation using affine-invariant neighborhoods.» En *Proc. of the Conference on Computer Vision and Pattern Recognition*. Wisconsin, USA, 2001, 319–324.
- [Lin07] LIN, H.T., C.J. LIN, y R.C. WENG. «A note on Platt’s probabilistic outputs for support vector machines.» *Machine Learning*, 68, (2007), 267–276.
- [Liu02] LIU, H., D. LIU, y J. XIN. «Real-Time recognition of road traffic sign in motion image based on genetic algorithm.» *Proc. of the 1st. Int. Conference on Machine Learning and Cybernetics*, (2002), 83–86.
- [Loy04] LOY, G., y N. BARNES. «Fast shape-based road sign detection for a driver assistance system.» En *Proc. of International Conference on Intelligent Robots and Systems (IROS)*. Senday, Japan, 2004, 70–75.
- [Loy03] LOY, G., y A. ZELINSKY. «Fast radial symmetry for detecting points of interest.» *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25, nº 8, (2003), 959–973.
- [Luo92] LUO, R.C., H. POTLAPALLI, y D. HISLOP. «Natural scene segmentation using fractal based autocorrelation.» En *Proc. of International Conference on Industrial Electronics, Control, Instrumentation and Automation*. San Diego, CA, 1992, 700–705.
- [Malik07] MALIK, R., J. KHURSHID, y S. AHMAD. «Road sign detection and recognition using colour segmentation, shape analysis and template matching.» En *Proc. of International Conference on Machine Learning and Cybernetics*. 2007, tomo 6, 3556–3560.
- [Mayoraz99] MAYOORAZ, E., y E. ALPAYDIN. «Support Vector Machines for Multi-class Classification.» En *Proc. of International Workshop on Artificial Neural Networks (IWANN99)*. 1999, tomo 2, 833–842.

- [McNemar47] MCNEMAR, Q. «Note on the sampling error of the difference between correlated proportions or percentages.» *Psychometrika*, , n<sup>o</sup> 2, (1947), 153–157.
- [Miura00] MIURA, J., T. KANDA, y Y. SHIRAI. «An active vision system for real-time traffic sign recognition.» *Proc. IEEE Intelligent transportation systems*, (2000), 52–57.
- [Nguwi08] NGUWI, Y. Y., y S. Y. CHO. «Two-tier self-organizing visual model for road sign recognition.» En *Proc. of the IEEE International Joint Conference on Neural Networks (IJCNN)*. Hong Kong, 2008, 794–799.
- [Nguwi06] NGUWI, Y. Y., y A. KOUZANI. «Automatic road sign recognition using neural networks.» En *Proc. of the IEEE International Joint Conference on Neural Networks (IJCNN)*. 2006, 3955–3962.
- [Nguyen13] NGUYEN, K., D. LE, y D. DUONG. «Efficient Traffic Sign Detection Using Bag of Visual Words and Multi-scales SIFT.» En *Proc. of the 20th Int. Conference on Neural Information Processing (ICONIP)*. Daegu, South Korea, 2013, 433–441.
- [Ohta80] OHTA, Y., T. KANADE, y T. SAKAI. «Color information for region segmentation.» *Computer Graphics and Image Processing*, 13, n<sup>o</sup> 3, (1980), 222–241.
- [Ojala02] OJALA, T., M. PIETIKÄINEN, y T. MÄENPÄÄ. «Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns.» *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24, n<sup>o</sup> 7, (2002), 971–987.
- [Otsu79] OTSU, N. «A threshold selection method from gray-level histograms.» *IEEE Trans. on Systemms, Man and Cybernetics*, 9, n<sup>o</sup> 1, (1979), 62–66.
- [Overett09] OVERETT, G., L. PETERSSON, L. ANDERSSON, y N. PETTERSSON. «Boosting a heterogeneous pool of fast HOG features for pedestrian and sign detection.» En *Proc. of the IEEE Intelligent Vehicles Symposium*. Xi'an, China, 2009, 584–590.
- [Overett11] OVERETT, G., y N. PETERSSON. «Large Scale Sign Detection using HOG Feature Variants.» En *Proc. of the IEEE Intelligent Vehicles Symposium*. Baden, 2011, 326 – 331.
- [Paclik99] PAČLÍK, P. «Road sign recognition survey.», (1999), Url: <http://euler.fd.cvut.cz/research/rs2/files/skoda-rs-survey.html>.

- [Paclik06] PAČLÍK, P., J. NOVÁKOVÁ, y R. P. W. DUIN. «Building Road-Sign Classifiers Using a Trainable Similarity Measure.» *IEEE Trans. on Intelligent Transportation Systems*, 7, n<sup>o</sup> 3, (2006), 309–321.
- [Paclik00] PAČLÍK, P., J. NOVÁKOVÁ, P. PUDIL, y P. SOMOL. «Road sign classification using laplace kernel classifier.» *Pattern Recognition Letter*, 21, (2000), 1165–1173.
- [Paletta02] PALETTA, L. «Detection of traffic signs using posterior classifier combination.» En *Proc. of the 16th Int. Conf. on Pattern Recognition*. 2002, 705–708.
- [Peden04] PEDEN, M., R. SCURFIELD, D. SLEET, D. MOHAN, A.A. HYDER, E. JARAWAN, y C. MATHERS. «World report on road traffic injury prevention.» *Inf. téc.*, World Health Organization, 2004.
- [Perez02] PEREZ, E., y B. JAVIDI. «Nonlinear distortion-tolerant filters for detection of road signs in background noise.» *IEEE Trans. on Vehicular Technology*, 51, n<sup>o</sup> 3.
- [Passerini04] —. «Nonlinear distortion-tolerant filters for detection of road signs in background noise.» *IEEE Trans. on Vehicular Technology*, 51, n<sup>o</sup> 3.
- [Perez94] PEREZ, F., y C. KOCH. «Toward color image segmentation in analog vlsi: algorithm and hardware.» En *Proc. of Int. Journal of Computing Vision*. 1994, tomo 12, 17–42.
- [Piccioli94] PICCIOLI, G., E. DE MICHELI, P. PARODI, y M. CAMPANI. «Robust road sign detection and recognition from image sequences.» En *Proc. of the Intelligent Vehicles '94 Symposium*. 1994, 278–283.
- [Piccioli96] —. «A robust method for road sign detection and recognition.» *Image and Vision Computing*, 14, n<sup>o</sup> 3, (1996), 209–223.
- [Platt00] PLATT, J.C. «Probabilities for SV machines.» *Inf. téc.*, MIT Press, 1998.
- [Platt99] —. «Fast training of support vector machines using sequential minimal optimization.» *Advances in kernel methods - Support Vector Learning*, (1999), 185–208.
- [Powers11] POWERS, David M.W. «Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.» *Journal of Machine Learning Technologies*, 2, n<sup>o</sup> 1, (2011), 37–63.

- [Perez05] PÉREZ-CRUZ, F., C. BOUSOÑO-CALZÓN, y A. ARTÉS-RODRÍGUEZ. «Convergence of the IRWLS Procedure to the Support Vector Machine Solution.» *Neural Computation*, 17, nº 1, (2005), 7–18.
- [Price94] PRICE, D., S. PERSONNAZ, L. KNERR, y G. DREYFUS. «Pairwise Neural Network Classifiers with Probabilistic Outputs.» *Advances in Neural Information Processing Systems*, 7, (1999), 1109–1116.
- [Priese94] PRIESE, L., J. KLIEBER, R. LAKMANN, y R. SCHIAN. «New results on traffic sign recognition.» En *Proc. of the IEEE Intelligent Vehicles '94 Symposium*. 1994, 249–254.
- [Priese93] PRIESE, L., V. REHRMEN, R. SCHIAN, R. LAKMANN, y L. BILDERKENNEN. «Traffic Sign Recognition Based on Color Image Evaluation.» En *Proc. of the IEEE Intelligent Vehicles '93 Symposium*. 1993, 95–100.
- [Pudil94] PUDIL, P., F. FERRI, J. NOVOCICOVA, y J. KITTLER. «Floating Search Methods for Feature Selection with Nonmonotonic Criterion Functions.» En *Proc. of the Int. Conference on Pattern Recognition*. 1994, 279–283.
- [Pujol06] PUJOL, O., P. RADEVA, y J. VITRIA. «Discriminant ECOC: A Heuristic Method for Application Dependent Design of Error Correcting Output Codes.» *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28, nº 6, (2006), 1001–1007.
- [Rajesh11] RAJESH, K., R. and. Rajeev, K. SUCHITHRA, V.P. LEKHESH, V. GOPAKUMAR, y N.K. RAGESH. «Coherence Vector of Oriented Gradients for Traffic Sign.» En *Proc. of the IEEE International Joint Conference on Neural Networks*. San José, California, USA, 2011, 907–910.
- [Refregier91] REFREGIER, P., y F. VALLET. «Probabilistic approach for multiclass classification with neural networks.» En *Proc. of the International Conference on Artificial Networks*. 1991, 1003–1007.
- [Trafico05] RODRÍGUEZ, José Ignacio. «El reto de sobrevivir.» *Tráfico y Seguridad Vial*, , nº 175, (2005), 12–18.
- [Ruta09] RUTA, A., Y. LI, y X. LIU. «Real-time traffic sign recognition from video by class-specific discriminative features.» *Pattern Recognition*, 43, (2009), 416–430.
- [Ruta11] RUTA, A., F. PORIKLI, S. WATANABE, y Y. LI. «In-vehicle camera traffic sign detection and recognition.» *Machine Vision and Applications*, 22, (2011), 359–375.

- [Sandoval00] SANDOVAL, H., T. HATTORY, S. KITAGAWA, y Chigusa. Y. «Angle-dependent edge detection for traffic signs recognition.» En *Proc. of the IEEE Intelligent Vehicles Symposium*. 2000, 308–313.
- [Sermanet11] SERMANET, P., y Y. LECUN. «Traffic Sign Recognition with Multi-Scale Convolutional Networks.» En *Proc. of the 2011 IEEE International Joint Conference on Neural Networks*. California, USA, 2011, 2809–2813.
- [Shadeed03] SHADEED, W. G., D. I. ABU-AL-NADI, y M. J. MISMAR. «Road traffic sign detection in color images.» En *Proc. of the IEEE 10th International Conference on Electronics, Circuits and Systems*. Cairo, Egypt, 2003, 890–893.
- [Shi08] SHI, M., H. WU, y H. FLEYEH. «Support vector machines for traffic signs recognition.» En *Proc. of the IEEE International Joint Conference on Neural Networks*. Hong Kong, 2008, 3820 – 3827.
- [Shuang-dong05] SHUANG-DONG, Z., Z. YI, y L. XIAO-FENG. «Detection for Triangle Traffic Sign Based on Neural Network.» En *Proc. of the IEEE 10th International Conference on Vehicular Electronics and Safety*. 2005, 25–28.
- [Simon09] SIMON, L., P. J. TAREL, y R. BREMOND. «Alerting the Drivers about Road Signs with Poor Visual Saliency.» En *Proc. of the IEEE Intelligent Vehicles Symposium*. Xi'an, China, 2009, 48–53.
- [Siogkas09] SIOGKAS, G. K., y E. S. DERMATAS. «Detection, Tracking and Classification of Road Signs in Adverse Conditions.» En *Proc. of the IEEE Mediterranean Electrotechnical Conference (MELECON)*. Málaga, Spain, 2006, 537–540.
- [Skarbek94] SKARBEEK, W., y A. KOSCHAN. «Colour Image Segmentation. A survey.» *Inf. téc.*, Institute of Computer Science (Polish Academy of Sciences) and Institute for Technical Informatics (Technical University of Berlin), octubre 1994.
- [Trafico00] SÁNCHEZ, Juana. «Si piensas, no conduzcas.» *Tráfico y Seguridad Vial*, n.º 143, (2000), 25–27.
- [Thomson01] THOMSON, M., y S. WESTLAND. «Colour-Imager characterization by parametric fitting of sensor responses.» *Colour Research and Application*, 26, n.º 6, (2001), 442–449.

- [Timofte11] TIMOFTE, R., K. ZIMMERMANN, y L. VAN GOOL. «Multi-view traffic sign detection, recognition, and 3D localisation.» En *Machine Vision and Applications* (Springer-Verlag, ed.).
- [Tominoga90] TOMINOGA, S. «A color classification method for color images using a uniform color space.» En *Proc. of the 10th Int. Conference on Pattern Recognition*. 1990, 803–807.
- [Tsai08] TSAI, L. W., J. W. HSIEH, C. H. CHUANG, y Y. J. TSENG. «Road sign detection using eigen colour.» *IET Computer Vision*, 2, nº 3, (2008), 164–177.
- [Ue] UNIÓN EUROPEA. «Official website of the European Union.» Disponible en Internet en el enlace [http://europa.eu/pol/trans/index\\_es.htm](http://europa.eu/pol/trans/index_es.htm).
- [Vapnik95] VAPNIK, V. *The Nature of Statistical Learning Theory*. Springer-Verlog, New York, 1995.
- [Vapnik98] —. *Statistical Learning Theory*. John Willey and Sons, Inc, New York, 1998.
- [Vapnik71] VAPNIK, V., y A. CHERVONENKIS. *On the Uniform Convergence of Relative Frequencies of Events to their Probabilities.*, tomo 17. Theoretical Probability and its applications, 1971.
- [Vicen07] VICEN-BUENO, R., R. GIL-PITA, P. JARABO-AMORES, y F. LÓPEZ-FERRERAS. «Complexity Reduction in neural networks applied to traffic sign recognition tasks.» *Proceedings of the 13th European Signal Processing Conference (EUSIPCO)*, 6, nº 4, (2007), 378–390.
- [Viola01] VIOLA, P., y M. J. JONES. «Robust real-time object detection. Technical Report CRL 2001/01.» *Inf. téc.*, Cambridge Research Laboratory, 2001.
- [Vitabile02] VITABILE, S., A. GENTILE, y F. SORBELLO. «A neural network based automatic road signs recognizer.» En *Proc. of the Int. Joint Conf. on Neural Networks (IJCNN)*. 2002, tomo 3, 2315–2320.
- [Vitabile01] VITABILE, S., G. POLLACCIA, G. PILATO, y F. SORBELLO. «Road signs recognition using a dynamic pixel aggregation technique in the hsv color space.» En *Proc. of the Int. Conf. on Image Analysis and Processing*. Washington, DC, USA, 2001.
- [Webb00] WEBB, G. I. «Multiboosting: A technique for combining boosting and wagging.» *Machine Learning*, 40, (2000), 159–197.

- [Wilcoxon45] WILCOXON, F. «Individual comparisons by ranking methods.» *Biometrics*, 1, (1945), 80–83.
- [Won08] WON. «Implementation of road traffic signs detection based on saliency map model.» En *Proc. of the IEEE Intelligent Vehicles Symposium*. Eindhoven. Netherlands, 2008.
- [WorldHealth09] WORLD HEALTH ORGANIZATION. «Global Status Report on Road Safety: time for action.» *Inf. téc.*, 2009. Disponible en el enlace [www.who.int/violence\\_injury\\_prevention/road\\_safety\\_status/2009](http://www.who.int/violence_injury_prevention/road_safety_status/2009).
- [Wu04] WU, T.-F., C.-J. LIN, y R.C. WENG. «Probability estimates for multi-class classification by pairwise coupling.» *Journal of Machine Learning Research*, 5, (2005), 975–1005.
- [Xie09] XIE, Y., L. F. LIU, C. R. LI, y Y. Y. QU. «Unifying Visual Saliency with HOG Feature Learning for Traffic Sign Detection.» En *Proc. of the IEEE Intelligent Vehicles Symposium*. Xi'an, China, 2009, 24–29.
- [Yates34] YATES, F. «Contingency table involving small numbers and the  $\chi^2$  test.» *Supplement to the Journal of the Royal Statistical Society*, 1, nº 2, (1934), 217–235.
- [Zadeh97] ZADEH, M., T. KASVAND, y C. SUEN. «Localization and recognition of traffic signs for automated vehicle control systems.» En *Proc. of the Int. Conf. on Intelligent Transportation Systems*. 1997.
- [Zakir10] ZAKIR, U., I. ZAFAR, y E. A. EDIRISINGHE. «Road Sign Detection and Recognition by Using Local Energy based Shape Histogram (LESH).» *International Journal of Image Processing (IJIP)*, 4, nº 6, (2010), 567–583.
- [Zaklouta12] ZAKLOUTA, F., y B. STANCIULESCU. «Real-Time Traffic-Sign Recognition Using Tree Classifiers.» *IEEE Trans. on Intelligent Transportation Systems*, 13, nº 4, (2012), 1507–1514.
- [Zaklouta11] ZAKLOUTA, F., B. STANCIULESCU, y O. HAMDOUN. «Traffic sign classification using K-d trees and Random Forests.» En *Proc. of the 2011 IEEE International Joint Conference on Neural Networks (IJCNN)*. San José, California, USA, 2011, 2151–2155.
- [Zhang08] ZHANG, R., y D. METAXAS. «Ro-svm: Support vector machine with reject option for image categorization.» *Inf. téc.*, Computer Science Department. State University of New Jersey, 2008.