

Esta Tesis está dedicada:

A Pablo que me ha dado fuerzas para terminar,  
a Sandra porque sin ella no hubiera tenido ganas de hacer nada,  
a mama porque el arbolito desde pequeñito,  
a papa porque esto tiene un truco,  
a mi hermano porque hablar con él me despeja la mente,  
a David porque de la discusión sale la luz,  
a Santiago, Juan Carlos, Sergio y Alberto que me han acompañado en este largo camino,  
a primos, tíos, cuñados y suegros por su calor,  
a Ángel que me ha metido en este lío,  
a Esteban porque siempre tiene un punto de vista enriquecedor,  
a Julia, Peña, Ortega, Casado, Ángela, Lupiani y a toda la gente de IPSA porque siempre han respondido con interés,  
a Belén, Abraham, Antonio, Patxi, Gabriela, Juanjo, Mika, Raul y a toda la gente de la URJC porque son mi sexto sentido,  
y a todo el que la lea porque la he escrito para él.





Universidad Rey Juan Carlos

Departamento de Ciencias de la Computación

**Un Sistema Off-Line  
de Verificación de Firmas Manuscritas  
para un Entorno Real  
de Procesamiento de Documentos**

Tesis Doctoral

Autor: José Francisco Vélez Serrano

Director: Ángel Sánchez Calle

Junio 2008



# Índice General

---

<b>Capítulo 1 Introducción.....</b>	<b>1</b>
1.1 Motivación.....	2
1.2 Procesamiento automático de firmas manuscritas.....	3
1.2.1 Captura de firmas.....	4
1.2.2 Segmentación y reconocimiento de firmas.....	5
1.2.3 Identificación y verificación de firmas.....	5
1.2.4 Requisitos de un sistema realista de verificación off-line de firmas.....	6
1.3 Hipótesis.....	7
1.3.1 Hipótesis relativas a la segmentación off-line de firmas.....	7
1.3.2 Hipótesis relativas a la verificación off-line de firmas.....	8
1.4 Objetivos.....	8
1.5 Estructura de la Tesis.....	10
<b>Capítulo 2 Problemática y estado del arte del tratamiento automático de firmas.....</b>	<b>13</b>
2.1 La escritura como medio clásico de autenticación.....	13
2.1.1 El peritaje caligráfico.....	14
2.1.2 Documentoscopia.....	14
2.1.3 Grafoscopia.....	14
2.1.4 Alteraciones de la escritura.....	17
2.1.5 Grafopatología, Grafología y Grafoterapia.....	19
2.2 Problemática relativa al tratamiento automático de las firmas.....	19
2.2.1 Problemas en la verificación automática off-line de firmas.....	20
2.2.2 Problemática de la identificación automática off-line de firmas.....	24
2.2.3 Problemática de la digitalización.....	24
2.2.4 Problemas en la segmentación y el reconocimiento.....	33
2.3 Técnicas de verificación automática de firmas off-line.....	38
2.3.1 Tipología de los trabajos sobre verificación off-line de firmas.....	39
2.3.2 Trabajos relevantes en el problema de verificación automática de firmas off-line.....	42
2.4 Técnicas de identificación automática de firmas off-line.....	47
2.4.1 Trabajos relevantes en el problema de la identificación de firmas.....	47
2.5 Técnicas de segmentación automática de firmas off-line.....	48
2.5.1 Trabajos relevantes en el problema de la segmentación de firmas.....	48
2.6 Resumen del capítulo.....	49
<b>Capítulo 3 Base de datos de firmas: captura y segmentación.....</b>	<b>51</b>
3.1 Base de datos de firmas.....	51
3.2 Obtención de firmas mediante formularios.....	52
3.2.1 Descripción del proceso de recogida de firmas.....	56
3.2.2 Descripción del proceso de digitalización y preproceso.....	57
3.2.3 Resultado del proceso de captura sobre formularios.....	58
3.3 Muestra procedente de documentos reales.....	59
3.4 Segmentación automática de la muestra real.....	61
3.4.1 Etapa de filtrado morfológico.....	61
3.4.2 Etapa de selección y descarte de objetos conexos.....	66
3.4.3 Resultados de los experimentos de segmentación sobre muestra real.....	72
3.5 Resumen del capítulo.....	73
<b>Capítulo 4 Verificación off-line de firmas utilizando muestra sintética.....</b>	<b>75</b>
4.1 Generación automática de muestra sintética.....	75
4.1.1 Factores de variabilidad y transformaciones equivalentes.....	76

4.1.2	Resultado de la generación sintética de firmas.....	82
4.2	Verificación de firmas utilizando una Red de Neuronas.....	82
4.2.1	Detalle del proceso de verificación basado en la imagen.....	83
4.2.2	Experimentos sobre la muestra de test.....	85
4.3	Verificación basada en características discriminantes.....	86
4.3.1	Descripción de las características discriminantes.....	87
4.3.2	Análisis de las características.....	93
4.3.3	Clasificador utilizado.....	94
4.3.4	Experimentos sobre la muestra de test.....	94
4.4	Verificación de firmas mediante Redes de Compresión.....	97
4.4.1	Uso de las Redes de Compresión para la verificación de firmas.....	97
4.4.2	Resultados del uso de las Redes de Compresión.....	99
4.5	Método de Cortes Posicionales para la verificación de firmas.....	103
4.5.1	Resultados del método de Cortes Posicionales.....	105
4.6	Resumen del capítulo.....	106
<b>Capítulo 5</b>	<b>Verificación de firmas mediante snakes.....</b>	<b>109</b>
5.1	Formulación original de los snakes.....	109
5.1.1	Aplicación de los snakes para la verificación off-line de firmas.....	110
5.2	Generación de snakes usando una imagen modelo.....	112
5.2.1	Obtención on-line del snake modelo.....	112
5.2.2	Generación automática off-line de snakes.....	113
5.3	Ajuste de un snake a la imagen de una firma.....	117
5.3.1	Uso del algoritmo original de snakes.....	117
5.3.2	Modificaciones adicionales a la propuesta original.....	121
5.3.3	Propuesta del uso del ángulo de los trazos como dimensión adicional.....	126
5.4	Formulación borrosa de la energía del snake.....	137
5.5	Características discriminantes para la verificación de firmas mediante snakes.....	141
5.5.1	Características habituales de ajuste elástico.....	141
5.5.2	Energía del snake como característica discriminante.....	143
5.5.3	Índice de coincidencia como característica discriminante.....	143
5.5.4	Índice de distancia como característica discriminante.....	143
5.5.5	Análisis de las características discriminantes.....	145
5.5.6	Entrenamiento de un clasificador.....	145
5.6	Resultados del sistema de verificación basado en snakes.....	147
5.7	Resumen del capítulo.....	149
<b>Capítulo 6</b>	<b>Conclusiones.....</b>	<b>151</b>
6.1	Conclusiones respecto al estado del arte.....	151
6.2	Conclusiones respecto a la base de datos de firmas.....	152
6.2.1	Conclusiones a la segmentación automática de firmas.....	152
6.3	Conclusiones a los métodos de verificación propuestos.....	153
6.3.1	Conclusiones respecto a la verificación usando muestra sintética.....	153
6.3.2	Conclusiones particulares a la verificación usando snakes.....	154
6.4	Resumen de publicaciones originadas por esta Tesis.....	155
6.5	Futuros trabajos.....	156
<b>Anexo A</b>	<b>Introducción a los snakes.....</b>	<b>157</b>
A.1	Los snakes.....	157
A.1.1	Energía interna.....	157

A.1.2 Energía de la imagen o externa.....	158
A.2 Minimización de la energía del snake.....	158
<b>Anexo B Redes Neuronales: Perceptrones Multicapa y Redes de Compresión.....</b>	<b>161</b>
B.1 Introducción a las Redes de Neuronas artificiales.....	161
B.1.1 El proceso de aprendizaje de una red.....	162
B.2 Estructura del Perceptrón Multicapa.....	163
B.3 Propiedades del Perceptrón Multicapa.....	165
B.3.1 Selección del número de capas ocultas.....	166
B.4 Algoritmos de aprendizaje para el Perceptrón Multicapa.....	167
B.4.1 La regla Delta.....	167
B.4.2 Generalización de la regla Delta.....	169
B.4.3 Las Redes de Compresión.....	171
<b>Anexo C Diagramas UML de Actividad.....</b>	<b>175</b>
C.1 Elementos de un Diagrama de Actividad.....	175
C.1.1 Acciones y nodos de actividad.....	175
C.1.2 Objetos de valor.....	175
C.1.3 Flujos de control y de objetos.....	176
<b>Anexo D Bibliografía.....</b>	<b>179</b>
<b>Anexo E Índice alfabético.....</b>	<b>191</b>



## Índice de figuras

---

Figura 1.1.-Principales métodos biométricos de verificación de identidad.....	2
Figura 1.2.- Áreas relativas al procesamiento automático de firmas manuscritas.....	4
Figura 1.3.- Diagrama UML de Actividad que describe un sistema de verificación.....	9
Figura 1.4.- Representación en árbol de un esquema de esta Tesis.....	10
Figura 2.1.- Diferentes grados de legibilidad de las firmas.....	17
Figura 2.2.- Ejemplo de la variabilidad en la firma de una persona.....	20
Figura 2.3.- Curvas utilizadas para presentar los resultados de un sistema de verificación.....	23
Figura 2.4.- Imagen de una firma capturada a diferentes resoluciones.....	26
Figura 2.5.- Imágenes de un cheque capturadas a diferentes resoluciones radiométricas.....	27
Figura 2.6.- Ejemplo de artefactos introducidos por el formato de compresión JPEG.....	30
Figura 2.8.- Diagrama típico para el flujo de información en una instalación.....	31
Figura 2.9.- Factores que se tienen en cuenta al construir una instalación.....	32
Figura 2.10.- Ejemplo de una línea sobre la que se requiere la firma.....	33
Figura 2.11.- Imagen de un cheque con un fondo pictórico.....	35
Figura 2.12.- Trama de fondo de un documento en forma de líneas oblicuas.....	35
Figura 2.13.- Imágenes de firmas sobre las que se aplicado un sello.....	36
Figura 2.14.- Imagen de firmas sobre texto preimpreso y mecanografiado.....	37
Figura 2.15.- Imagen en color de un conjunto de dígitos manuscritos.....	37
Figura 2.16.- Ruido como producto de la binarización.....	38
Figura 2.17.- Histograma de referencias de los artículos.....	40
Figura 2.18.-Principales autores en verificación automática de firmas off-line.....	43
Figura 3.2.- Ejemplo de la segunda hoja de muestras.....	55
Figura 3.3.- Firma correspondiente a un sujeto de 90 años.....	56
Figura 3.4.- Resultado de la umbralización de la Figura 3.3.....	58
Figura 3.5.- Ejemplo de documento proporcionado por la empresa Investigación y Programas.....	60
Figura 3.6.- Detalle del proceso de limpiado sobre documentos reales.....	60
Figura 3.7.- Ejemplo de un documento manuscrito y otro mecanografiado.....	61
Figura 3.8.- Ejemplo de oscilación en una línea horizontal.....	63
Figura 3.9.- Diagrama UML que presenta las 5 fases del algoritmo de eliminación de líneas.....	63
Figura 3.10.- Ejemplo de eliminación de líneas sobre el cuarto inferior de un cheque.....	65
Figura 3.11.- Número de componentes conexas por firma.....	66
Figura 3.12.- Relación entre el área de los rectángulos horizontales.....	67
Figura 3.13.- Tamaño de las componentes conexas.....	67
Figura 3.14.- Árbol que muestra las combinaciones posibles de las componentes conexas.....	69
Figura 3.15.- Comparación entre las envolventes conexas.....	71

Figura 3.16.-Ejemplo de ruido que no ha sido eliminado en la segmentación.....	71
Figura 3.17.- Ejemplo del etiquetado manual realizado sobre el resultado de la segmentación.....	73
Figura 4.1.- Posición del eje de mínima inercia sobre diferentes firmas.....	76
Figura 4.3.- Ejemplo que muestra la variación en la longitud de los extremos de los trazos.....	77
Figura 4.4.- Diferencias en los algoritmos de esqueletizado.....	78
Figura 4.5.- Ejemplo de modificación de los extremos de los trazos.....	79
Figura 4.6.- Ejemplares que muestran diferencias de grosor.....	80
Figura 4.7.- Ejemplo que manifiesta elongación inicial y final de la firma.....	81
Figura 4.8.- Elongación de los extremos de una firma.....	81
Figura 4.9.- Árbol de transformaciones que da origen a la muestra sintética.....	82
Figura 4.10.- Diagrama de Actividad para la verificación utilizando una Redes de Neuronas.....	83
Figura 4.11.- Representación del proceso de verificación usando la imagen y un Perceptrón.....	84
Figura 4.12.- Curvas de error obtenidas al usar una Red de Neuronas sobre la muestra B.....	86
Figura 4.13.- Curvas de error obtenidas al usar una Red de Neuronas sobre la muestra E.....	86
Figura 4.14.- Diagrama de Actividad UML para el método basado en características globales.....	87
Figura 4.15.- Histograma sobre la variación de las proporciones de los rectángulos.....	88
Figura 4.16.- Ejemplo de envolvente conexa y de posición de los ejes.....	89
Figura 4.17.- Ejemplo de detección de bucles formados por los trazos.....	90
Figura 4.18.- Patrones utilizados para detectar los puntos de cruce entre los trazos de una firma.....	91
Figura 4.19.- Puntos de cruce entre trazos sobre la imagen esqueletizada de una firma.....	91
Figura 4.21.- Ejemplo de cálculo de la BSL, del UL y del LL de una firma.....	93
Figura 4.22.- Media y desviación de los Coeficientes de Fisher para diferentes características.....	95
Figura 4.23.- Curva de error obtenido durante el entrenamiento.....	96
Figura 4.24.- Curvas de error obtenidas al usar el método basado en características discriminantes.....	96
Figura 4.25.- Curvas de error obtenidas al usar el método basado en características discriminantes.....	97
Figura 4.26.- Ejemplo de la Red de Compresión.....	98
Figura 4.27.- Entrenamiento de la Red de Compresión.....	100
Figura 4.28.- Resultado de presentar a la red del individuo 4 una firma del individuo 4.....	100
Figura 4.29.- Resultado de presentar a la red del individuo 4 una firma del individuo 2.....	100
Figura 4.30.- Error cometido por la Red de Compresión del individuo 1.....	101
Figura 4.31.- Diferencias entre la entrada y la salida de las Redes de Compresión.....	102
Figura 4.32.- Curvas de error obtenidas al usar el método basado en Redes de Compresión.....	103
Figura 4.33.- Curvas de error obtenidas al usar el método basado en Redes de Compresión.....	103
Figura 4.34.- Diagrama de Actividad UML que describe el método de Cortes Posicionales.....	104

Figura 4.35.- Curvas de error obtenidas al usar el método de Cortes Posicionales.....	106
Figura 4.36.- Curvas de error obtenidas al usar el método de Cortes Posicionales.....	106
Figura 5.1.- Diagrama de Actividad que describe el uso de snakes para la verificación de firmas.....	111
Figura 5.2.- Contexto de la generación automática de snakes.....	115
Figura 5.3.- Ejemplo en el que el esqueletizado no sigue de manera natural al trazo.....	115
Figura 5.4.- Resultado de la aplicación del algoritmo de localización de puntos extremos.....	115
Figura 5.5.- Ejemplo del resultado de la generación del snake a partir de una imagen modelo.....	118
Figura 5.6.- Imagen sintética utilizada en las pruebas.....	119
Figura 5.7.- Ejemplo que muestra cómo se ajusta un snake a la figura de una 'S'.....	119
Figura 5.8.- Ejemplo que ilustra el problema de la localidad de los snakes.....	121
Figura 5.9.- Ejemplo que ilustra el problema de la ausencia de memoria de los snakes.....	121
Figura 5.10.- Resultado del cálculo del mapa de potencial sobre la imagen de la 'S'.....	122
Figura 5.11.- Ejemplo de movimiento del snake cuando se utiliza un mapa de potencial.....	124
Figura 5.12.- Ejemplo de funcionamiento del algoritmo modificado de snakes:.....	127
Figura 5.13.- Energía obtenida para las firmas de la muestra A por el snake con memoria.....	127
Figura 5.14.- Ejemplo de error en el ajuste del snake.....	128
Figura 5.15.- Ejemplo de distancia en un mapa de potencial bidimensional.....	129
Figura 5.16.- Ejemplo de distancia en un mapa de potencial que considera el ángulo.....	129
Figura 5.17.- Representación tridimensional de una imagen bidimensional.....	130
Figura 5.18.- Ejemplo de la determinación de la recta tangente a un punto p del trazo.....	131
Figura 5.19.- Ejemplo de acierto en el ajuste del snake.....	134
Figura 5.20.- Ajuste de un snake triangular sobre una imagen con dos figuras geométricas.....	135
Figura 5.21.- Detalle de ajuste del snake tras el proceso iterativo de convergencia.....	136
Figura 5.22.- Energía obtenida para las firmas de la muestra A por el snake tridimensional.....	136
Figura 5.24.- conjuntos borrosos para el cálculo de la energía borrosa de un snake.....	139
Figura 5.25.- Resultados de energía borrosa obtenidos al aplicar las reglas borrosas.....	140
Figura 5.26.- Energía obtenida para las firmas de la muestra A por el snake fuzzy.....	141
Figura 5.27.- Ejemplo del proceso de convergencia del snake en un caso de no coincidencia.....	142
Figura 5.28.- Ejemplos de ajuste del snake de una persona a las imágenes de dos firmas.....	144
Figura 5.29.- Posición de los patrones de la muestra A respecto a sus características.....	146
Figura 5.30.- Esquema del MLP utilizado como clasificador.....	147
Figura 5.31.- Curva del error cuadrático medio obtenido durante el entrenamiento.....	147
Figura 5.32.- Salida del MLP de la figura 5.30 una vez entrenada.....	148
Figura 5.33.- Curvas de error obtenidas al usar snakes tridimensionales con memoria.....	148

Figura 5.34.- Curvas de error obtenidas al usar snakes tridimensionales con memoria.....	149
Figura B.1.- Ejemplo de una Redes de Neuronas genérica.....	161
Figura B.2.- Ejemplo del entrenamiento de un Red de Neuronas.....	163
Figura B.3.- Aspecto de las Funciones de Activación Escalón (arriba) y Sigmoide (abajo).....	164
Figura B.4.- Patrones 2D separados por un hiperplano.....	165
Figura B.5.- Esquemas de los tipos de Perceptrones.....	165
Figura B.6.- Los patrones de una muestra siempre se pueden agrupar en regiones convexas.....	166
Figura B.7.- Esquema general de una conexión entre dos neuronas en un Perceptrón Multicapa. .	170
Figura B.8.- Formato de una Red de Compresión.....	173

## Índice de algoritmos

---

Algoritmo 3.1 de eliminación de líneas horizontales.....	64
Algoritmo 3.2 de segmentación de firmas manuscritas.....	72
Algoritmo 4.1 de mejora al proceso de esqueletizado.....	78
Algoritmo 4.2 de verificación basado en Cortes Posicionales.....	105
Algoritmo 5.1 de sobresegmentación de trazos.....	116
Algoritmo 5.2 de unión de trazos sobresegmentados.....	117
Algoritmo 5.3 de verificación de firmas mediante snakes.....	120
Algoritmo 5.4 de creación del mapa de potencial.....	123
Algoritmo 5.5 de cálculo de distancia en un mapa de potencial 3D.....	132
Algoritmo 5.6 optimizado de cálculo de distancia en un mapa de potencial 3D.....	133



## Índice de tablas

---

Tabla 2.2.- Tamaño en bytes obtenido al digitalizar un A4.....	29
Tabla 2.3.- Algunos tipos de clasificadores utilizados para la verificación de firmas.....	42
Tabla 2.4.- Resumen de los trabajos que tratan el problema de las falsificaciones aleatorias.....	46
Tabla 2.5.- Resumen de los trabajos que tratan el problema de las falsificaciones habilidosas.....	47
Tabla 3.1.- Resumen de la descripción de la muestra recogida.....	59
Tabla 3.2.- Características de las imágenes de la muestra.....	73
Tabla 3.3.- Resultados de la segmentación sobre documentos reales.....	73
Tabla 4.1.- Características discriminantes implementadas.....	88
Tabla 5.1.- Ventajas e inconvenientes de los diferentes dispositivos de captura on-line de firmas...113	
Tabla 5.2.- Análisis de las características discriminantes calculadas tras la convergencia del snake. .145	



# Capítulo 1 Introducción

Según la primera acepción dada por el diccionario de la lengua de la Real Academia Española, “la firma es el nombre y apellido, o título, que una persona escribe de su propia mano en un documento, para darle autenticidad o para expresar que aprueba su contenido”.

Las personas utilizamos la firma manuscrita desde tiempos inmemoriales para identificar la autoría de nuestras creaciones, pero en los últimos tiempos, la Sociedad de la Información ha hecho que la firma cobre aún más importancia. Raro es el día que, por un motivo u otro, las personas de las civilizaciones occidentales no firmamos varios documentos. Una pequeña compra con una tarjeta de crédito o con un cheque, el alquiler de un coche, la reserva de una sala, o un contrato de compra-venta, son algunos ejemplos de situaciones en las que se suelen requerir una o varias firmas manuscritas.

La Sociedad de la Información también ha venido acompañada de una herramienta: el ordenador. Este dispositivo, ya indispensable, continúa integrándose en muchos aspectos de nuestra vida. Una de las últimas fronteras, que esta herramienta trata de alcanzar, consiste en la realización de tareas que exigen un alto nivel de comprensión e intuición, y que hasta hace poco eran consideradas específicamente humanas. Escuchar y entender una conversación, o leer y comprender un texto, son ejemplos de este tipo de tareas entre las que también se incluye el tratamiento automático de la firma manuscrita.

Desde cierta perspectiva, relativa al procesamiento automático, las tareas principales que se pueden realizar sobre un documento que contiene firmas manuscritas son cuatro: la captura del documento físico y su conversión a formato digital, la segmentación de las firmas presentes en el documento, la identificación del autor cada una de las firmas y la verificación de las mismas. Como veremos, la verificación es la tarea que, debido a su potencial aplicación para la detección de fraudes, ha despertado mayor interés. De hecho, el resto de tareas no suelen considerarse por separado, sino como partes de un sistema completo de verificación de firmas.

Esta Tesis recopila diversos trabajos que el autor ha realizado, con objeto de lograr un sistema automático de verificación de firmas para un entorno real de procesamiento de documentos. Algunos de estos trabajos ya han sido presentados en congresos y revistas específicos de este área de investigación, y otros se presentan en esta Tesis por primera vez. En general, este trabajo pretende contribuir al estudio del problema del procesamiento automático de firmas manuscritas sobre documentos digitalizados mediante varias aportaciones entre las que se pueden destacar:

- Una revisión del estado del arte respecto al procesamiento automático de firmas manuscritas.
- La creación de una base de datos pública de imágenes digitalizadas de firmas manuscritas, que refleje los problemas reales de los documentos firmados.
- La presentación de varios algoritmos novedosos, basados en técnicas de Visión Artificial, orientados a resolver los problemas de segmentación y de verificación de una firma manuscrita sobre un documento original digitalizado en condiciones realistas.

## 1.1 Motivación

La firma manuscrita es un rasgo biométrico<sup>1</sup> que constituye uno de los principales métodos de verificación de autoría que manejan las personas. Los cheques bancarios, las tarjetas de crédito y muchos documentos legales utilizan la firma como elemento de comprobación de autoría [1]. La firma también se utiliza para la identificación de autoría en aplicaciones forenses o en textos históricos [2].

La Figura 1.1 presenta una sencilla clasificación de diversas modalidades biométricas para verificar la identidad de una persona. Puede observarse que de todas ellas sólo la firma, y quizás la huella dactilar, permiten ser estampadas en un documento, de manera que no se requiera la presencia física o virtual del sujeto en el momento de la comprobación.

Una de las grandes ventajas del uso de la firma como mecanismo de autoría radica en la facilidad que tienen las personas para realizarla [4]. Esto se debe a que casi todas las personas han adquirido, directa o indirectamente como parte de su formación académica, la destreza necesaria para realizar su firma. Además, el acto de firmar no precisa de grandes medios pues apenas se necesita un papel y un bolígrafo.

Otra gran ventaja de la firma consiste en que cualquier persona es capaz de realizar una verificación visual básica [4]. Además, en los casos en los que se precise detectar cuidadosas falsificaciones se puede recurrir a un experto, como por ejemplo un perito calígrafo, que utilizando medios adecuados (como un microscopio) puede verificar con un margen de error muy pequeño y con validez legal la autenticidad de una firma. Todo esto ha motivado que la firma sea un mecanismo de verificación de autoría aceptado por la sociedad en general, y por organismos públicos y privados en particular.

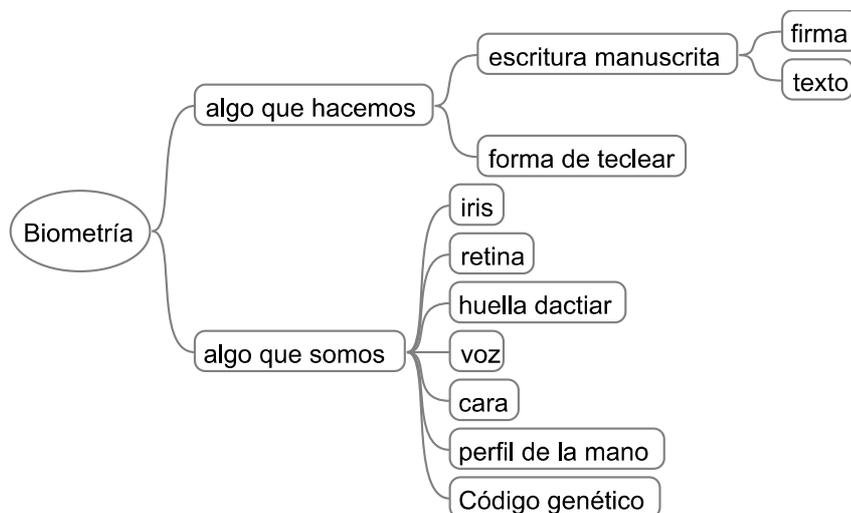


Figura 1.1.-Principales métodos biométricos de verificación de identidad.

Finalmente, se puede añadir que la firma no es un mecanismo de verificación de identidad intrusivo, como puede ocurrir con otras modalidades biométricas como las huellas dactilares, el uso del iris o la voz. Se pueden citar algunas objeciones comunes a estas técnicas más intrusivas. Por ejemplo, en el caso de las huellas dactilares son comunes las reticencias a que se manchen las manos al tocar algo

1 El término biometría clásicamente se aplica a la ciencia dedicada al estudio estadístico de las características cuantitativas de los seres vivos (p. ej. el peso o la longitud). En épocas más recientes, este término se utiliza también para referirse a los métodos automáticos que analizan determinadas características físicas (p. ej. la huella dactilar o la imagen facial) o de comportamiento (p. ej. la voz o el modo de caminar) con el fin de identificar y autenticar a las personas [3].

que no tenga una higiene especial. En el caso del reconocimiento del iris o la retina podemos poner objeciones a que nos iluminen los ojos aunque sea fugazmente. Incluso podemos tener reticencias a que se grabe nuestra voz con objeto de que actúe un sistema de huella vocal. Sin embargo, está socialmente aceptado el firmar un documento si se está conforme con lo que en él se expresa.

Las administraciones públicas y especialmente el sector bancario utilizan de manera habitual la firma como medio de verificación de autoría. Así, cuando se abre una cuenta se firma un contrato (*contracts*), cuando se paga con un cheque se debe firmar (*checks*), cuando se recibe una letra se firma que se ha pagado, cuando se utiliza una tarjeta en un establecimiento se firma un pequeño recibo (*credit card slip*). La firma se utiliza mucho, en realidad, el volumen de documentos firmados en todas estas operaciones es tan elevado que las entidades financieras no tienen posibilidad material de verificar cada una de las firmas de los documentos que gestionan. Por ello, se limitan a almacenar estos documentos de forma preventiva por si se produce algún problema, en cuyo caso podrían utilizarlos como prueba. No es de extrañar, por tanto, que el desarrollo de un sistema automático de verificación de firmas sea de gran interés para estos sectores [1][5][6][7].

Es poco probable que un sistema automático de verificación de firmas sobre documentos digitales tenga alguna vez validez legal debido a que puede ser imposible la ausencia de error que se le exigiría. Sin embargo, esto no quiere decir que tal sistema carezca de utilidad. En aquellos casos en los que el volumen de firmas a comprobar es elevado, con 2 ó 3 millones de documentos a revisar diariamente, un sistema de prevención de fraude sólo sería viable si existiesen sistemas automáticos que redujesen la carga de trabajo en 3 ó 4 ordenes de magnitud. Por ejemplo, es un hecho que un pequeño porcentaje de los cheques que se procesan en toda entidad bancaria están firmados por alguien que no goza del consentimiento de la persona pagadora del cheque. Como no es posible comprobar cada cheque sin introducir un coste elevado de operativa, se suelen pagar sin ninguna verificación los cheques cuyo valor no sobrepasa un importe umbral fijado por cada banco en cada momento y lugar. Dependiendo del volumen de cheques que se procesen, estas falsificaciones suponen una no tan pequeña cantidad de dinero que se pierde diariamente. Una vez pagado, si el cliente de la entidad descubre el robo, tras su airada protesta, el banco suele hacerse cargo del importe. Esta misma situación se da en el caso de pago fraudulento utilizando tarjetas de crédito.

Por otro lado, y aunque tampoco son infalibles, las decisiones de los peritos calígrafos sobre la autoría de una firma sí gozan de validez legal. Por ello, las técnicas automáticas que se desarrollen siempre podrían servir para construir herramientas de apoyo a la labor de estos expertos.

Finalmente, el procesamiento automático de firmas, en particular, y el desarrollo de un sistema de localización y comparación de figuras manuscritas, en general, es un problema interesante desde un punto de vista científico. ¿Qué mecanismos utilizan las personas para realizar estas tareas? ¿Cómo se puede crear un sistema automático que reproduzca, al menos parcialmente, esta habilidad humana? Son estas preguntas las que principalmente han motivado la realización de la presente Tesis.

## 1.2 Procesamiento automático de firmas manuscritas

El problema del procesamiento automático de firmas manuscritas involucra diferentes subproblemas. Para empezar, no sólo es preciso verificar la firma, primero es necesario encontrarla en el documento y separarla del ruido. Además, la elección del tipo de dispositivo de captura digital que se utilice puede cambiar la forma de enfocar el problema. La Figura 1.2 y las siguientes secciones ordenan y describen estos subproblemas así como algunos aspectos que son claves para poder entender las hipótesis y los objetivos de esta Tesis.

### 1.2.1 Captura de firmas

La primera etapa en cualquier sistema de Visión Artificial consiste en la *captura* o *digitalización* de las imágenes sobre las que se van a realizar las subsiguientes tareas de análisis y clasificación. En general, el proceso de captura consiste en la transformación de la información del dominio continuo del mundo físico al dominio discreto del mundo informático. En particular, el proceso de captura de firmas puede realizarse de dos formas diferentes: *on-line* y *off-line* [8][9][4].



**Figura 1.2.-** Áreas relativas al procesamiento automático de firmas manuscritas y principales aspectos inherentes.

En el modo *on-line* (o *dinámico*) la captura se realiza utilizando un dispositivo especial (un lápiz electrónico, una tableta digitalizadora [8] o una cámara [9]) que recoge información dinámica del escritor durante el acto de la firma. Esta información incluye, además del grafismo, la forma cómo se construye, mediante una secuencia ordenada de puntos. También incluye datos como la velocidad instantánea, la presión y la inclinación del útil de escritura, obtenidas en cada punto del trazado. Existen actualmente sistemas que realizan de forma precisa y eficiente la verificación utilizando esta abundante información dinámica [10].

Por otro lado, el método de captura *off-line* (o *estático*) se basa en el escaneo del documento que contiene la firma. Tanto el soporte donde se escribe, como el medio de escritura son elementos de uso cotidiano, típicamente un bolígrafo y un papel. Obsérvese que en el proceso de firmado *off-line*, a diferencia del proceso *on-line*, no hay ningún elemento electrónico involucrado, y que la etapa de escaneo se realiza necesariamente en un instante posterior indeterminado (segundos, días o años) [4]. El método *off-line* es menos intrusivo para las personas ya que la intervención del sistema no altera el procedimiento habitual de firma. Es decir, no hacen falta complejos medios tecnológicos al firmar, y la verificación siempre podría ser comprobada por una persona. Esto hace que un sistema *off-line* de verificación de firmas tenga muchas más aplicaciones potenciales que uno *on-line* [11]. Por contra, en el enfoque *off-line* la información capturada es mucho menor que en el caso *on-line*, pues no se recoge explícitamente la información relativa a la dinámica del proceso de firmado. Así, en el caso *off-line* si se usa una imagen bitonal sólo se dispone del trazo, su luminancia si se usan imágenes en niveles de gris, y su matiz si la imagen es en color [12].

### 1.2.2 Segmentación y reconocimiento de firmas

Según los manuales de Tratamiento Digital de Imágenes, la etapa de *segmentación* consiste en realizar una partición de la imagen digital de manera que cada píxel posea una etiqueta distintiva del objeto al que pertenece [13]. En este caso, la segmentación de firmas dentro de documentos digitalizados, consiste en identificar los píxeles de la imagen que corresponden a la firma y descartar el resto [14].

Cuando la captura es *on-line* no existe el problema de la segmentación de la firma. Esto se debe a que se usa un sistema informático para realizar la escritura, y por lo tanto se obtiene la firma y toda la información dinámica libre de ruido.

Por contra, en el caso *off-line* la segmentación dentro de un documento real es un problema complejo. Esta complejidad se debe a que se desconoce el tamaño y la posición exacta de la firma dentro del documento, y tampoco se sabe como distinguirla del resto de objetos presentes en la imagen que, a efectos prácticos, corresponden a ruido. El caso más sencillo ocurre cuando estos objetos son píxeles que aparecen de manera aleatoria, se habla entonces de la presencia de *ruido blanco*. Desafortunadamente, el caso habitual es más complicado y se da cuando estos objetos no son tan pequeños ni aparecen de forma aleatoria, sino que tienen estructura, en ese caso hablamos de *ruido con estructura* [12]. El ruido blanco suele deberse a artefactos introducidos por el escáner (problemas ópticos o mecánicos) y a la existencia de zonas coloreadas en el documento que no se consiguen eliminar totalmente. Mientras que el ruido con estructura puede tener diversos orígenes: sellos que se han superpuesto a la firma, tramas de logotipos sobre los que se ha firmado, texto adyacente a la zona de firmado, líneas o recuadros sobre los que se firma u otro texto manuscrito (como comentarios, vistos bueno, o incluso otras firmas).

Para minimizar el impacto de este ruido sobre el proceso de segmentación suelen aplicarse técnicas de filtrado. Sin embargo, debe tenerse en cuenta que la aplicación de tales técnicas suele aparejar una pérdida de información relativa a la propia imagen de la firma. El método que se utilice para eliminar ruido dependerá en gran medida de la resolución espacial y radiométrica que tenga la imagen. En general, la tarea se vuelve más sencilla cuanto mayor resolución tenga la imagen. El caso más habitual en entornos reales consiste en el uso de imágenes bitonales (blanco y negro) con 200 puntos por pulgada de resolución espacial, lo cual representa a un escenario difícil por insuficiente.

Unido al problema de la segmentación encontramos el problema del *reconocimiento de firmas* [14]. Consiste éste en decidir, ante una imagen desconocida, si es o no es una firma. En general es un problema complejo pues implica comprender la semántica del documento donde se encuentra el objeto.

### 1.2.3 Identificación y verificación de firmas

La *identificación de firmas* (*signature identification*) [5][15] es un proceso que ante una firma trata de decidir a qué individuo corresponde dentro de un conjunto más o menos amplio de individuos conocidos. Así, ante un individuo que se desea identificar, un sistema de identificación devuelve el individuo más parecido de los que conoce, o los candidatos más probables. El problema de identificación de firmas también recibe el nombre de *recuperación de firmas* (*signature retrieval*) [5][16] y en algunos casos también el de *reconocimiento de firmas* (*signature recognition*) [2], creándose en este caso cierta confusión con la definición del párrafo previo.

Por otro lado, el proceso de la *verificación* (*signature verification*) de una firma tiene por objeto determinar si una firma es auténtica o no [17][2]. En particular, consiste en determinar si una firma, que se afirma que ha sido realizada por una persona, realmente ha sido realizada por ella. Para ello

suele realizarse una comparación entre la firma cuestionada y otras muestras, típicamente otras firmas, de las que no se tiene duda.

Desde cierto punto de vista se puede decir que ambos procesos, la identificación y la verificación, son en realidad un mismo proceso con diferente cardinalidad. Así, la identificación tendría una formulación 1 a  $N$  y la verificación tendría una formulación más restringida de tipo 1 a 1.

A grandes rasgos, los falsificadores, que pueden tratar de engañar a un sistema de identificación o de verificación, pueden catalogarse en dos grupos: los *falsificadores hábiles (skilled)* y los *aleatorios (random)* [18]. Los falsificadores diestros conocen la firma de la persona a quien quieren suplantar, se entrenan en reproducir la firma y consiguen unas falsificaciones de gran calidad. Los aleatorios son aquéllos que no han visto nunca la firma a imitar e incluso desconocen el nombre de la persona a la que tratan de suplantar, por lo que su reproducción no suele guardar ningún parecido con la firma auténtica. Curiosamente, y en contra de lo que pudiese pensarse, a este segundo grupo de falsificadores se le atribuye el 99% del fraude que se produce en las entidades bancarias de todo el mundo [19][11].

#### 1.2.4 Requisitos de un sistema realista de verificación *off-line* de firmas

Un primer requisito para la construcción de un sistema *off-line* de verificación automática de firmas, utilizable en condiciones realistas, consiste en que no es viable solicitar muchas firmas a cada usuario del sistema [20][21][22]. Tanto es así que, en la práctica, sólo suele disponerse de una firma por persona. Sin embargo, sí es factible exigir que la firma de referencia utilizada para construir el sistema esté libre de ruido y en una posición concreta.

Otro requisito consiste en que, un sistema *off-line* realista de verificación de firmas, deberá resultar escalable con respecto al número de individuos que es capaz de verificar, y no deberá ser demasiado exigente en cuanto a las condiciones de resolución espacial y radiométrica.

Además, el sistema deberá funcionar de manera autónoma, sin supervisión, tanto en la etapa de aprendizaje de nuevos individuos como en la etapa operativa.

Un cuarto requisito consiste en que, una vez construido el sistema, el proceso de verificación automática de una nueva firma debe realizarse en un tiempo de respuesta razonable (unos pocos segundos). Es importante que el proceso utilizado tenga un tiempo de respuesta tan breve que posibilite el tratamiento de miles de documentos a la hora y, en general, a un coste inferior al perjuicio que introducen las falsificaciones.

Un sistema de verificación de firmas normalmente funcionaría dentro de instalaciones que manejan varios millares de documentos a la hora. Esta característica implica que las imágenes que se capturan deben almacenarse en un formato que minimice el tamaño y el tiempo empleado para su proceso, y que posibilite las tareas que se desean realizar sobre el documento. Este punto es importante, ya que el problema de verificación cambia dependiendo del dispositivo que se emplee para obtener las imágenes y la configuración del mismo. Como referencia se puede indicar que una persona es capaz de realizar el proceso de verificación con éxito, ante falsificaciones aleatorias, con imágenes bitonales (blanco y negro) de resoluciones inferiores a 100 puntos por pulgada [23].

Además, el cliente potencial de un sistema automático de verificación de firmas desea percibirlo como una mejora libre de molestias a su proceso actual. Por ello, en algunos ámbitos como el bancario es preferible que el sistema acepte una firma falsa (*falso positivo*) a que rechace una firma verdadera (*falso negativo*), siempre que el coste económico de la supuesta falsificación no supere un umbral previamente establecido. Esto se debe a que previamente a la introducción de un sistema de verificación automático no se analizaban las firmas, o sólo se hacía en casos de importes elevados, y

por lo tanto el coste del trabajo era cero. Elevar este coste no suele ser aceptado por un personal que no recibe un beneficio directo del sistema y que “antes no tenía ese problema” [24].

### 1.3 Hipótesis

La hipótesis central de esta Tesis Doctoral puede resumirse en el siguiente enunciado: “puede construirse un sistema automático de verificación de firmas en *condiciones realistas*”. Expresando con mayor detalle esta hipótesis: “se puede decidir de manera automática, en tiempo razonable y con efectividad si se debe aceptar como auténtico o rechazar como falso un documento real, el cual está firmado supuestamente por una persona concreta de la que se conoce un único ejemplar de su firma”.

Esta hipótesis se sustenta en varias subhipótesis de mayor detalle:

- La imagen digitalizada de una firma, realizada sobre un documento con un medio común de escritura, contiene información suficiente para plantear un problema de segmentación y verificación del firmante. Más concretamente, el uso de una imagen digitalizada con resolución radiométrica bitonal y resolución espacial inferior a 300 puntos por pulgada es suficiente para abordar el problema de la segmentación y verificación frente a falsificadores aleatorios de una firma realizada en un papel con un bolígrafo.
- Es posible encontrar un algoritmo que segmente una firma dentro de la digitalización de un documento de carácter general cuyo formato previamente se desconoce. Este algoritmo será robusto frente al ruido común en los documentos digitalizados.
- En el caso de las falsificaciones aleatorias es posible encontrar un algoritmo que permita identificar que una firma es similar a otra de referencia en ausencia de más ejemplares. Este método también debe ser robusto frente a la variabilidad inherente a las firmas (como pequeñas rotaciones, variaciones en las formas o cambios de grosor) y al ruido que pueda sobrevivir a los procesos de segmentación. Ello permitiría utilizar una única firma de referencia por individuo a verificar.
- La tarea de segmentación y verificación de una firma puede realizarse con medios computacionales modestos en un tiempo razonable que permita su uso efectivo en aplicaciones reales (por ejemplo en el entorno bancario). Así, utilizando un ordenador de sobremesa, un escáner para la captura y empleando un tiempo inferior a unos pocos segundos es posible realizar la tarea de segmentar y verificar la autenticidad de una firma sobre un documento digital.

Las siguientes secciones concretan más estas hipótesis.

#### 1.3.1 Hipótesis relativas a la segmentación *off-line* de firmas

La solución que se propone para la verificación de firmas se basa en una observación que se pasa por alto en todos los trabajos sobre segmentación revisados: en un problema de verificación se conoce la forma de la firma que se desea segmentar. En base a este hecho se plantean las siguientes hipótesis:

- Se puede eliminar la mayor parte del ruido existente en documentos capturados en condiciones reales utilizando técnicas específicas basadas en morfología matemática [25].
- Conociendo una firma modelo de un individuo y conociendo la posición aproximada a priori de dicha firma en un documento es posible concretar exactamente su posición.

### 1.3.2 Hipótesis relativas a la verificación *off-line* de firmas

Para verificar las firmas se plantean dos aproximaciones. Una basada en el uso de técnicas de *generación automática de muestra* combinadas con clasificadores neuronales, y otra basada en el uso de *modelos de Contornos Activos* [26], también conocidos como *snakes*.

Respecto al uso de generación automática de muestra sintética para el problema de la verificación se plantean las siguientes hipótesis:

- Utilidad del uso de un conjunto de firmas generado de manera sintética a partir de una única firma original para realizar los diferentes procesos de aprendizaje clásicos.
- Utilidad de una técnica, que llamamos de Cortes Posicionales, que utiliza una secuencia de imágenes, correspondientes a una división en ventanas verticales de la imagen de una firma, para clasificarla como auténtica o falsa. Esta técnica se basa en el uso de una Red Neuronal de compresión entrenada con la muestra sintética.

Por otro lado, el uso de modelos de Contornos Activos para el problema de la verificación de autoría de firmas se sustenta sobre las siguientes hipótesis:

- Posibilidad de adaptar un Contorno Activo de una firma modelo a la imagen de la presunta firma de un individuo, utilizando la técnica de *snakes* o una adaptación de la misma.
- Idoneidad del uso, como característica discriminante, de una medida del grado de adaptación del *snake* a la imagen de la firma a verificar para aceptarla como auténtica o rechazarla como falsa.
- Factibilidad de la generación automática del *snake* de una firma a partir de una imagen modelo.

## 1.4 Objetivos

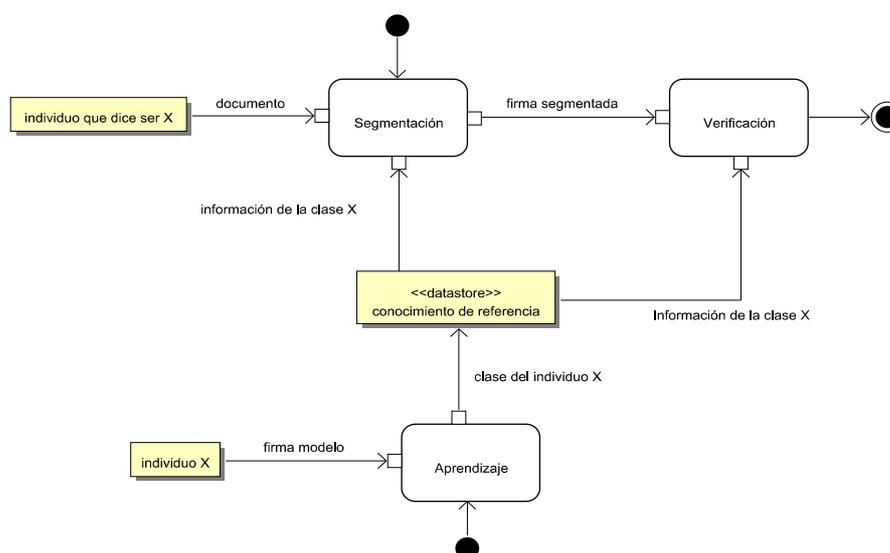
El principal objetivo de esta Tesis Doctoral es el desarrollo de algoritmos orientados a la creación de un sistema completamente automático y escalable de segmentación y verificación de firmas *off-line* en condiciones realistas. Con mayor precisión, se propone como objetivo el desarrollo de un conjunto de algoritmos orientados a la verificación de firmas, que sólo precisen de una firma por individuo para su construcción, que sean suficientemente inmunes al ruido que suele acompañar a las digitalizaciones de los documentos que las contienen, y que operen en condiciones realistas en cuanto a resolución, velocidad y capacidad de crecimiento. Este objetivo principal puede dividirse en los siguientes subobjetivos:

- Obtener una base de datos de firmas digitalizadas que se puedan utilizar para el desarrollo de algoritmos de segmentación y verificación de firmas. En concreto, se desea disponer de las firmas de un conjunto amplio de individuos, disponiendo para cada individuo de distintas firmas tomadas en diferentes momentos, con diferentes elementos de escritura y sobre diferentes soportes.
- Desarrollo de un algoritmo de segmentación que permita localizar la imagen de una firma dentro de una imagen más general correspondiente a un documento digitalizado. En particular, se experimentará una idea basada en el análisis de las diferentes combinaciones de objetos que podrían corresponder a la firma. Estos objetos se obtendrán tras la aplicación de una serie de filtros morfológicos que intentan preservar el texto manuscrito y eliminar la parte no manuscrita de un documento digitalizado.
- Desarrollo de un conjunto de algoritmos que permitan verificar si la imagen de una firma, una vez segmentada, corresponde a la de cierto individuo. Este último objetivo se cubrirá utilizando varios

enfoques con el fin de que se complementen y también de poder compararlos. Además todos los algoritmos desarrollados deberán ser aplicables en condiciones realistas, para ello, todas las propuestas que se desarrollen cumplirán las siguientes condiciones:

- Los algoritmos solo dispondrán de una única firma modelo de cada individuo.
- Los algoritmos deberán ser escalables en cuanto al número de individuos a procesar. Es decir, el coste computacional de un test de verificación deberá ser independiente del número de individuos que sea capaz de verificar el sistema. Además, el número potencial de individuos a verificar no estará limitado.
- Los algoritmos deberán tener una complejidad reducida para que sea posible utilizarlos en aplicaciones reales.
- Los algoritmos deberán utilizar imágenes digitales bitonales escaneadas con una resolución espacial igual o menor a 300 puntos por pulgada.
- Los algoritmos deberán ser inmunes al ruido de diferente naturaleza que habitualmente acompaña a los documentos digitales.

Estos objetivos vertebrarían un sistema de verificación en condiciones realistas como el descrito por el *Diagrama de Actividad UML*<sup>2</sup> de la Figura 1.3. En este diagrama se distinguen dos fases desconectadas: una de aprendizaje a partir de una firma modelo y otra de operación sobre documentos reales. Las diferentes propuestas que se presentarán en esta Tesis encajarán en este diagrama general.

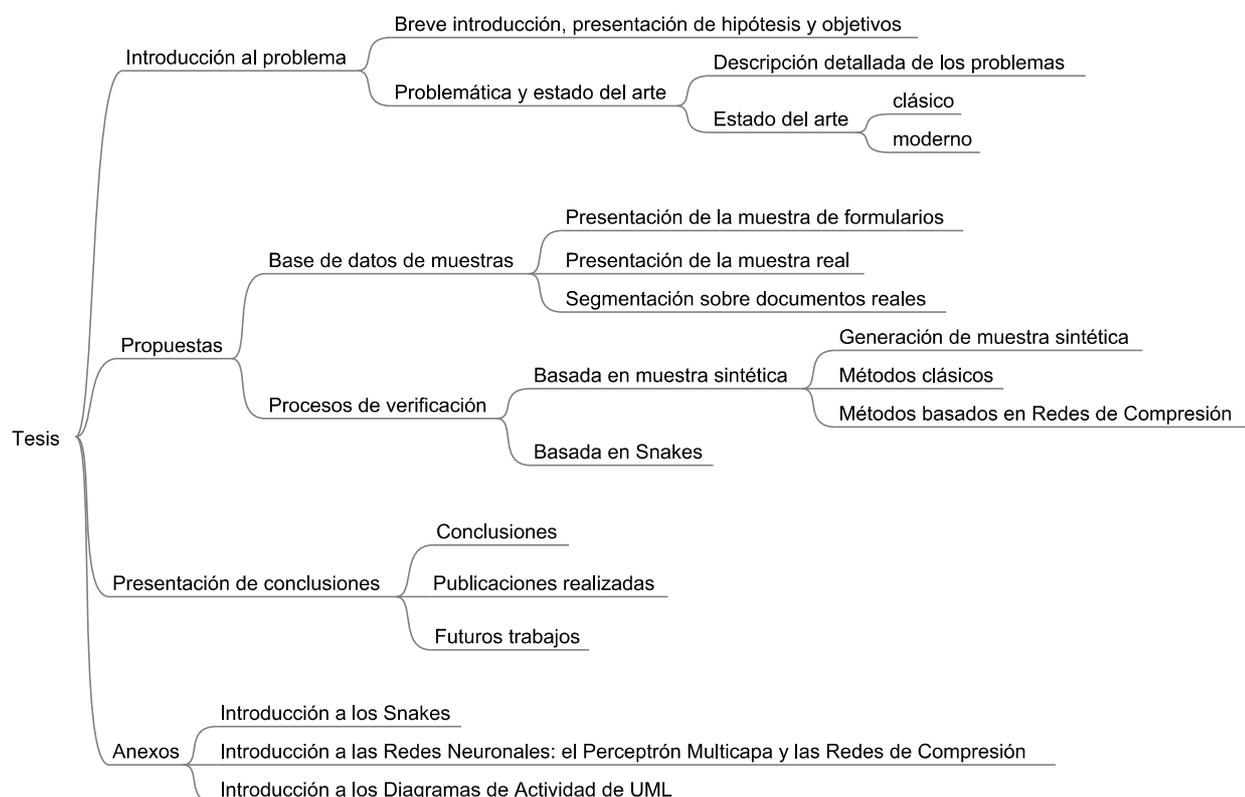


**Figura 1.3.-** Diagrama UML de Actividad que describe un sistema de verificación de firmas *off-line* en condiciones realistas.

2 El Lenguaje Unificado de Modelado (*Unified Modelling Language* o *UML*) designa al lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. *UML* es un lenguaje gráfico que facilita la especificación de los sistemas de software mediante una notación cargada de semántica. *UML* dispone de diferentes tipos de diagramas orientados a los diferentes aspectos de los sistemas que se modelan. En particular, los Diagramas de Actividad son uno de los diagramas *UML* más usados al describir el comportamiento de un software. En el anexo C se introducen brevemente este tipo de diagramas.

## 1.5 Estructura de la Tesis

La Figura 1.4 presenta un diagrama esquemático de la estructura de esta Tesis. A grandes rasgos se puede comprobar una división en una primera parte introductoria y de análisis, una segunda de preparación de muestra, una tercera de diseño de métodos y realización de experimentos, y una final de presentación de conclusiones.



**Figura 1.4.-** Representación en árbol de un esquema de esta Tesis.

De forma más detallada se puede comentar que en el Capítulo 2 se analiza la problemática de las firmas manuscritas y se expone el estado del arte de la disciplina. En este capítulo se presenta, en primer lugar, varios ejemplos que permiten entender las dificultades a las que se enfrenta un sistema de verificación de firmas. A continuación, se aborda el estado del arte desde dos perspectivas: por un lado desde un punto de vista clásico, previo a la introducción del ordenador, y por otro lado desde la perspectiva aportada por los numerosos trabajos aparecidos en los últimos años en el ámbito automático. Se considera que esta división es importante por varios motivos: por un lado sirve para presentar la nomenclatura clásica, y por otro sirve para apreciar la influencia de los enfoques de verificación clásicos sobre los modernos. Este Capítulo 2 es esencial para entender el resto de la Tesis, y tras él, serán comunes las referencias al mismo para justificar diversas decisiones de diseño.

Con el Capítulo 3 comienza el bloque de aportaciones metodológicas de esta Tesis. En él se presenta una base de datos de muestras de firmas que se ha construido a lo largo de varios años. Para empezar el capítulo se justifica la necesidad de la base de datos. Luego, se describe en detalle el proceso de captura seguido y se analizan los resultados obtenidos. Para finalizar el capítulo se aborda el problema de la segmentación mediante un enfoque novedoso. Éste se aborda en dos fases: en la primera se describen una serie de filtros cuya combinación permite eliminar gran parte del ruido presente en las imágenes reales, y en la segunda se propone el uso de técnicas de clasificación para elegir aquella combinación de elementos que parece corresponder a la firma de un individuo.

Finalmente, el algoritmo de segmentación se utiliza para segmentar parte de la muestra, siendo el resultado de esta segmentación un conjunto de prueba que se utilizará en experimentos posteriores.

En el Capítulo 4 se propone un algoritmo de generación de muestra sintética que, a partir de una única firma de un individuo, genera múltiples variaciones de la misma que intentan reproducir la variabilidad inherente al acto de firmar. A continuación, se proponen diversos métodos de verificación de firmas que utilizan esta muestra sintética como único elemento de aprendizaje. Así, se presentan dos trabajos que siguen una metodología clásica dentro del ámbito de la verificación de firmas: uno basado en usar la propia imagen de la firma como entrada a un clasificador, y otro basado en el cálculo de las principales características discriminantes citadas por la bibliografía. Estos trabajos aportan como principal característica novedosa el basarse en el sistema de generación de muestra sintética y por tanto sólo utilizar una firma en su etapa de aprendizaje. Además, estos métodos ofrecen una conexión imprescindible con trabajos de referencia en el área de la verificación de firmas, permitiendo contrastar los resultados de los métodos que se presentan posteriormente. Para finalizar el capítulo se presentan dos métodos originales de verificación de firmas basados en Redes de Compresión.

El Capítulo 5 presenta un método basado en *snakes* para la verificación de firmas. Este método, que también utiliza una única firma modelo, hace uso de técnicas de *matching* elástico para lograr la verificación. En el capítulo se describen las diferentes adaptaciones que se han realizado sobre el algoritmo original de *snakes* para obtener finalmente un sistema de verificación de firmas operativo.

En el Capítulo 6 se presentan las conclusiones, se enumeran las aportaciones, se mencionan las diversas publicaciones en las que se han publicado algunas de estas propuestas y se describen diversas líneas futuras de desarrollo.

La Tesis se complementa con tres anexos teóricos que complementan el contenido de los capítulos precedentes. En el primer anexo se introducen formalmente los *snakes* y una formulación basada en programación dinámica que posibilita su uso eficiente. En el segundo se introducen las Redes de Neuronas, el Perceptrón Multicapa y las Redes de Compresión. En el tercer anexo se introducen brevemente los Diagramas de Actividad UML. Finalmente, se presenta la bibliografía consultada y un índice de palabras.



## Capítulo 2 Problemática y estado del arte del tratamiento automático de firmas

La verificación de firmas manuscritas se engloba clásicamente dentro de las áreas forense y criminalista. Las técnicas utilizadas se remontan a finales del siglo XIX, cuando este tipo de métodos renovó las técnicas de investigación policial y de análisis judicial. Sus técnicas no han variado sustancialmente desde aquella época, y ni los modernos sistemas de microscopía, ni los progresos en computación han modificado en un grado perceptible sus procedimientos. En la primera sección de este capítulo se realiza un breve repaso de esta disciplina clásica con un doble objetivo: por un lado introducir la nomenclatura formal relativa a la verificación de firmas, y por otro lado presentar los métodos que normalmente utilizan los expertos para la verificación de firmas.

Por otro lado, la construcción de algoritmos para el tratamiento automático de firmas es un área mucho más reciente. Se inicia en la década de 1970 con los primeros Sistemas de Visión Artificial y desde entonces es un área de investigación muy activa. En una segunda sección de este capítulo se introducen los principales problemas relativos al tratamiento automático de firmas.

En la tercera sección de este capítulo se expone un análisis exhaustivo a este respecto de los trabajos presentados en los principales foros especializados. Como se comprobará, la actividad en esta área, motivada en gran medida por el interés que para la industria tienen sus resultados, queda patente por la gran cantidad de trabajos que aparecen. Tanto es así que en el año 2004 se celebró la primera competición internacional de verificación de firmas (*First International Signature Verification Competition*), la cual ha continuado realizándose periódicamente hasta la actualidad.

### 2.1 La escritura como medio clásico de autenticación

Según Villalain [27] la escritura en general y la firma en particular es una función realizada simultáneamente por la mente consciente y subconsciente, y en ella tienen reflejo factores relativos a los movimientos musculares y actos nerviosos del cuerpo. Su valor radica en que en su ejecución se plasman, de manera involuntaria y automática, una gran cantidad de características que permiten la identificación de un determinado escritor.

En el proceso de aprendizaje de la escritura, las personas aprenden formas o convencionalismos sobre cómo hacer las letras. Sin embargo, estas formas no se mantienen inalteradas por mucho tiempo sino que cambian debido, entre otros factores, a la influencia de la observación de la escritura de otras personas y al hábito motor del individuo. De esta manera, cada individuo dispone de una personalidad gráfica que es prácticamente irreproducible por otro escritor en forma espontánea y exacta.

Las formas o convencionalismos se denominan en criminalística *características de clase*. Así, un *factor caligráfico* es el nombre técnico que recibe una determinada característica individual hallada en la escritura.

Sin embargo, la escritura está sujeta a circunstancias externas o internas al ser humano que según su intensidad van a provocar variaciones que afectarán en mayor o menor grado la producción

espontánea de automatismos (por ejemplo: estados emocionales, enfermedades físicas, edad, o circunstancias extremas del clima).

### 2.1.1 El peritaje caligráfico

La figura del perito calígrafo tiene una larga tradición. Ya en el año 654 Recesvinto en el *Liber Indiciurum* (posteriormente traducido como Fuero Juzgo por Fernando III en 1241) incluía en el libro II un título llamado '*De los Escriptos Dubdosos*' (títulos V-XV) en el que dispone que el obispo de la zona y el juez examinen los documentos y determinen si los escritos son similares para declarar sobre la legitimidad de lo dubitado. Luego el código destaca que el perito debe ser un obispo porque en él se tiene confianza, tanto en honestidad como en capacidad.

La figura del perito calígrafo llega hasta nuestros días como la de un experto forense en escritura manuscrita y mecanografiada. En la literatura inglesa recibe diferentes nombres: *forensic document examiner*, *handwriting expert* y *handwriting analyst*. Este especialista aporta su capacidad principalmente en procesos judiciales y en análisis de documentos históricos. El campo en el que desarrolla su actividad un perito calígrafo es la *documentoscopia*. Dentro de este campo la rama en la que se encuadra la verificación de firmas es la *grafoscopia*.

### 2.1.2 Documentoscopia

El vocablo documentoscopia (*forensic document examination*), es una formación híbrida que proviene del latín *documentus* y del griego *copain* [28]. Se define como el campo que trata de establecer mediante una metodología propia la autenticidad de escritos y documentos, y determinar, cuando sea posible, la identidad de sus autores [29].

El área de estudio de la documentoscopia se centra en la verificación de falsificaciones entendiéndose con este término la duplicación o reproducción fraudulenta de documentos o escrituras, así como las posibles alteraciones en los impresos o en la escritura. La documentoscopia estudia tanto el escrito (mediante la grafoscopia) como el soporte o tipo de superficie (área física que contiene la información comunicada en el documento) que puede variar de forma y composición (por ejemplo: papel, cartón, muros, plásticos...).

Las herramientas que se utilizan en documentoscopia son básicamente dos:

- Lentes de distintas dioptrías (aumentos), bien en modo de lupas o microscopios (generalmente estereoscópicos).
- Fuentes de luz y filtros con los que se realizan observaciones cualitativas y cuantitativas.

### 2.1.3 Grafoscopia

La grafoscopia o *grafotecnia* (*handwriting examination*) es una rama criminalista que examina escrituras y firmas para identificar a su autor. Su principal procedimiento consiste en el análisis comparativo de la escritura y de la firma de una o varias personas. Sus objetivos son:

- Identificar al autor de manuscritos.
- Verificar la autenticidad de las firmas o descubrir falsificaciones.
- Identificar a los autores de escritos anónimos.
- Verificar la legitimidad de notas póstumas.
- Verificar la autenticidad de documentos dudosos, falsos, alterados y/o camuflados.

La grafoscopia apenas ha evolucionado desde finales del siglo XIX. Los medios técnicos que se utilizan siguen siendo los mismos, si bien el uso del ordenador ha permitido un aumento en la capacidad de almacenamiento de datos sobre la población en general por parte de las autoridades [30].

A lo largo de la historia de la grafoscopia se han desarrollado numerosas técnicas. A continuación se describen brevemente por orden cronológico las principales [31]:

- Método *morfológico* caligráfico o *gramatomórfico*.- Consiste en la comparación caligráfica de las formas extrínsecas de los grafismos. Sólo tiene en cuenta elementos estáticos que son fáciles de reproducir.
- Método *grafológico*.- Se basa en el estudio de cualidades subjetivas del grafismo y ahonda en las cualidades psicológicas atribuibles a la escritura.
- Método *sinalético* o *segnalético-descriptivo*.- Que se centra en los rasgos salientes y en las particularidades constantes que se encuentran en las formas estáticas de un escrito.
- Método *anastasiográfico* y *escopométrico*<sup>3</sup>.- Estos métodos, que se originan en el análisis de textos antiguos, se fundamentan en el análisis bajo luz ultravioleta que aporta peculiaridades reveladoras ante alteraciones de los documentos.
- Método *grafonómico*.- Es una evolución del método morfológico. Abandona toda interpretación psicológica y utiliza exclusivamente los elementos estructurales definidos por Crepieux-Jamin: orden, forma, tamaño, inclinación, dirección, velocidad, presión y cohesión. Aunque no posee un criterio objetivo de valoración del análisis es el método que actualmente goza de mayor aceptación entre los peritos calígrafos.
- Método *grafonómico*.- Básicamente reduce la escritura a un conjunto de cifras que luego compara mediante un proceso estadístico. Para descubrir la falsificación utiliza una medición precisa de los caracteres ideada por Humbert y un procedimiento estadístico diseñado por Langenbruck. Su idea fundamental es estudiar elementos que el falsificador no reproduce porque no son aparentes.

Según F. Álvarez [32] la técnica fue abandonada a principio del siglo XX debido a los trabajos de Schneickert y de Meyer [33] que demostraron su poca fiabilidad con fines identificativos. Sin embargo, en los años 40 el profesor Edmond Locard [34][35] recuperó la técnica.

Aunque el grafonómico es el único método dentro de la metodología clásica que permite obtener un resultado cuantitativo, la experimentación no ha ofrecido resultados positivos concluyentes. Unos lo achacan a la invalidez del método, mientras que otros a la gran cantidad de muestra que sería necesaria para que el resultado fuese estadísticamente significativo [32].

### **Factores caligráficos de la firma**

Para el análisis de la firma la grafoscopia distingue dos tipos de ejemplares: firmas legibles y firmas ilegibles.

Las firmas ilegibles son aquellas que no contienen escritura legible, y que básicamente corresponden a un garabato personal. En ellas las características estudiadas por el peritaje caligráfico son:

---

<sup>3</sup> El escopómetro es un aparato inventado a mitad del siglo XX para ayudar al estudio de textos mecanografiados y que toma su nombre del gabinete escopométrico argentino donde surge.

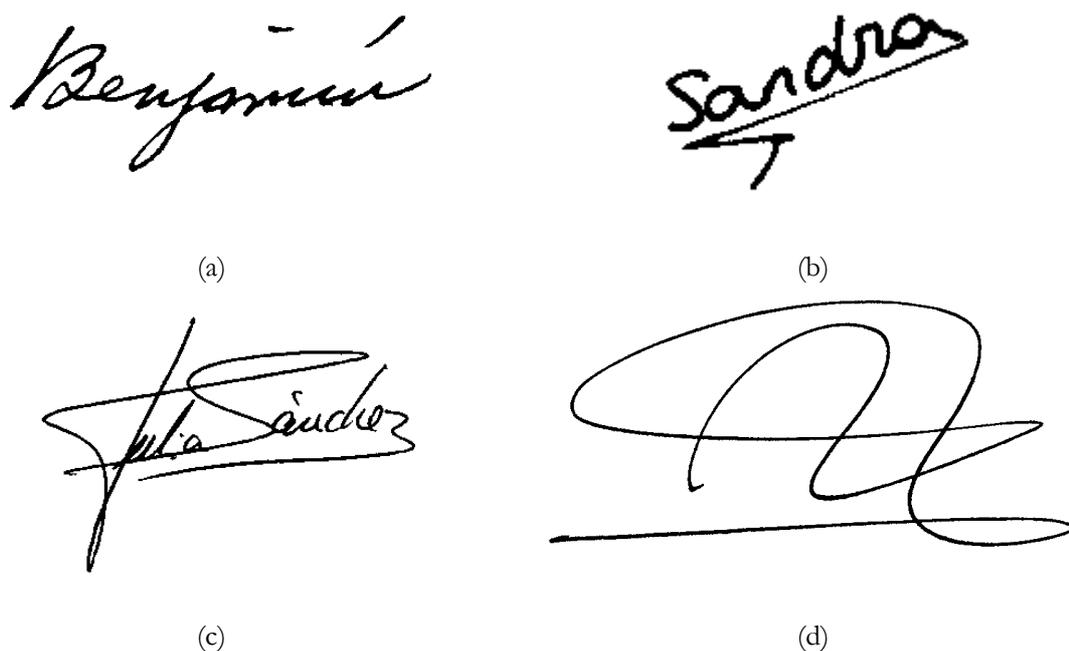
- La *trayectoria* que sigue el útil de escritura cuando se realiza la firma. En general es posible descubrir el orden en el que se han realizado los diferentes trazos que componen una firma, aunque en firmas complejas puede ser necesario recurrir al uso de microscopios sobre el documento original. Sin embargo, en ocasiones pueden encontrarse situaciones indecidibles.
- La *inclinación* de los caracteres respecto a la línea base que parece soportar la firma.
- El *tamaño* y las *proporciones* de la firma.
- La *velocidad* aparente de la escritura, deducida de características de los trazos como: el grosor, las interrupciones o depósitos de tinta o la deformación del papel.
- Relación de distancia establecida con respecto a los bordes del documento o zona donde se escribe.

Las firmas legibles son aquellas compuestas por caracteres legibles. En este caso las características estudiadas por el peritaje caligráfico son un subconjunto de las utilizadas en análisis de texto manuscrito:

- Forma particular de hacer los caracteres. Por ejemplo analizando los adornos o trazos agregados en letras, símbolos o números.
- Análisis de los trazos iniciales y finales en los caracteres de escritura. De estos se observará su orientación, forma y presión aparente ejercida en su construcción.
- El estudio de las acentuaciones, los signos de puntuación, las admiraciones y las interrogaciones, analizando sus posiciones, formas y tamaños.
- Distancias detectadas en la escritura. Específicamente la relación existente entre letras, sílabas, palabras, símbolos y números.
- Posición de la escritura sobre la línea base (imaginaria o real) que parece soportar los caracteres. Esta posición suele catalogarse como: ascendente, descendente, elevada, irregular o ajustada a la línea que le sirve de base.
- Calidad de línea estudiada mediante el análisis de los juegos de presión y velocidad aparentes manifestados a lo largo de la escritura.
- Inclinación de la escritura en relación con un eje perpendicular al plano (hacia la izquierda, a la derecha, irregular, o perpendicular).
- Revisión de la ortografía buscando omisiones espontáneas de reglas ortográficas.
- Análisis de las separaciones y los *enlodamientos* que se producen entre los caracteres. Por ejemplo cuando se unen habitualmente las terminaciones de letras como la “g”, o la “P” con la siguiente letra que aparezca.
- Estudio de los *adosamientos* entre los trazos, es decir de las uniones forzadas entre dos trazos cuando debería existir un trazo continuo.

Así, atendiendo a estos dos tipos de firmas, se decide estudiar unos u otros elementos dentro de las mismas. El estudio de la grafoscopía se sustenta en la hipótesis de que el imitador o falsificador de firmas no va a poder reproducir en su totalidad este conjunto de características, sino que pondrá énfasis en unas pero omitirá otras.

También se debe señalar que las firmas pueden ser mixtas, constando de una parte legible y otra parte ilegible (ver Figura 2.1) por lo que los análisis suelen involucrar simultáneamente ambos aspectos.



**Figura 2.1.-** Diferentes grados de legibilidad de las firmas. En (a) la firma es idéntica a la escritura del individuo y es totalmente legible; en (b) la firma sigue siendo legible pero empieza a contener elementos gráficos añadidos; en (c) la firma se hace más ilegible y más gráfica, aunque aún conserva elementos legibles; y en (d) la firma sólo es gráfica y no contiene elementos legibles.

Finalmente, cabe resaltar que la tipología de las firmas cambia considerablemente dependiendo de la cultura en las que se originan. Así, las firmas orientales, las árabes y las occidentales tienen pocos parecidos. Incluso, dentro del mundo occidental hay grandes variaciones. Por ejemplo, en los países anglosajones la firma siempre es legible, mientras que en los países latinos suele ser mixta o ilegible.

#### 2.1.4 Alteraciones de la escritura

La escritura en general y la firma en particular están sujetas a dos tipos de alteraciones [31] que son objeto de estudio de la grafoscopia:

- Alteraciones no fraudulentas debidas a cambios en los útiles de escritura, a elementos ambientales, a patologías, a estados emotivos o a la edad.
- Alteraciones fraudulentas realizadas con objeto no lícito, bien ocultando la propia identidad para eludir una responsabilidad, bien imitando o alterando las grafías ajenas para obtener un beneficio no legítimo.

Idealmente, un sistema de verificación de firmas debe obviar las primeras y detectar las segundas.

##### Alteraciones no fraudulentas de la firma

Las alteraciones no fraudulentas de la firma suponen variaciones en la escritura que se producen de manera involuntaria y sin ánimo de fraude. Distingue Antón [31] dos tipos principales de alteraciones: las ocasionales y las permanentes.

Las ocasionales se deben a circunstancias que rodean el acto de firmar. Entre ellas Antón destaca:

- La calidad del papel, que puede afectar a la aparición de grumos o rebabas en los trazos.
- Las características del útil de escritura (pluma, bolígrafo, rotulador...) pues pueden aportar información sobre la presión o pueden ayudar a discriminar al escritor si habitualmente utiliza el mismo útil.
- La fluidez del tipo de tinta, que permite valorar los *brisados* (espacios en blanco que resultan en roturas del trazo) y los *depósitos* (acumulaciones o borrones de tinta) en los trazos.
- Elementos ambientales como el motivo del escrito, el destinatario, el lugar donde se realiza o la temperatura.
- Algunas patologías transitorias y estados emotivos.

Las alteraciones permanentes están ligadas a situaciones irreversibles o de larga duración. Entre ellas se pueden citar todas las enfermedades que introduzcan problemas en las extremidades superiores. Finalmente, la edad también introduce cambios que podrían ser considerados como permanentes.

### **Alteraciones fraudulentas de la firma**

Las *alteraciones fraudulentas* son aquéllas que se realizan con intención de fraude. F. de Antón [31] las clasifica en tres grupos: desfiguración de la propia firma, imitación de la firma de otro y alteración de la firma de otra persona.

La alteración de la firma propia tiene por objeto dificultar su identificación. Para ello, se modifican los aspectos más visibles de su trazado, se utilizan letras estereotipadas, se cambia la postura natural de escritura, se inventan grafías nuevas... Señala Antón que el diagnóstico es difícil y precisa de abundante muestra de cotejo.

La alteración por imitación se da cuando un falsificador intenta reproducir lo más fielmente la firma de otra persona. Distingue Antón seis tipos de imitaciones: imitación servil, imitación por asimilación de grafías, falsificación por calco, falsificación por medio de punzón, falsificación mecánica y falsificación mediante composición. En los siguientes puntos se describen brevemente cada una de ellas:

- En la *imitación servil* el autor copia la firma auténtica teniéndola a la vista. En este caso, en los trazos imitados suelen encontrarse numerosas paradas, reenganches, enmiendas, torsiones, y temblores. La lupa y el microscopio son los útiles necesarios para detectar las anomalías descritas.
- La *imitación por asimilación* de grafías ocurre cuando el falsificador ensaya hasta conseguir una firma que tenga gran semejanza con la auténtica. Esta falsificación presenta grandes dificultades ya que la soltura y decisión que tienen recuerdan a la espontaneidad de las firmas auténticas. Su detección se basa en el cotejo de la firma con ejemplares auténticos, comparando la presión y la velocidad aparente en la ejecución de los trazos, así como los puntos de inicio y fin de trazo, los adornos y las distancias.
- La falsificación por *calco* se realiza habitualmente al trasluz y su resultado es de baja calidad apareciendo trazos inseguros, lentos, temblorosos y con adosamientos. Además, si se dispone del original que se utilizó para el calco, la concordancia exacta con éste es suficiente para asegurar el fraude.
- El uso de un *punzón* para emular la presión observada en los trazos de un original sufre los mismos problemas que la falsificación por calco.

- La *falsificación mediante sistemas mecánicos* se derivan de unas máquinas reproductoras de firmas que aparecieron en EE.UU. en la década de 1970. Estos aparatos similares a los *plotters* de planos reproducen de manera idéntica una firma modelo y además no reproducen la presión de la escritura humana, por lo que su detección es sencilla.
- Finalmente, la *falsificación por composición* utiliza medios como las fotocopias o procesos digitales para realizar composiciones gráficas fraudulentas. La perfección y definición pueden ser muy elevadas. El estudio de la presión, que en principio no es reproducido por estos sistemas, y un análisis fotográfico o microscópico permite demostrar el fraude.

Finalmente, Antón [31] precisa que no siempre es posible realizar un cotejo con un mínimo de fiabilidad. En particular, cuando las firmas son ilegibles y de sencilla realización se pierden muchos elementos técnicos necesarios para la labor.

### Mecanismos de seguridad del papel

La superficie del documento es otro elemento que se debe inspeccionar cuidadosamente cuando se sospecha que una firma puede ser una falsificación. En el caso del papel se busca la desorganización de las fibras que lo componen, el adelgazamiento en su espesor, restos o decoloraciones de la tinta, y repintes o tachaduras. Además, si el papel consta de mecanismos de seguridad se debe verificar la integridad de los mismos. Son los estados y las entidades bancarias los que principalmente introducen estos mecanismos en sus papeles timbrados y cheques. Se enumeran a continuación los más habituales:

- *Fondos de seguridad* impresos con tintas delezables para revelar cualquier intento de borrado o lavado químico.
- *Banda luminiscente*, por ejemplo sensible a la luz ultravioleta, que protege textos especialmente sensibles.
- *Películas plásticas* de alto poder adherente que se pegan encima de zonas sensibles para evitar alteraciones.
- *Papeles permeables* con entramado que permite a la tinta penetrar hasta capas profundas del papel, dificultando el borrado.
- Papeles que reaccionan ante ciertos productos para atestiguar su autenticidad (*respuesta química*).

### 2.1.5 Grafopatología, Grafología y Grafoterapia

Desde finales del XIX han surgido numerosas pseudociencias a la sombra del peritaje caligráfico como la *grafopatología*, la *grafología* (*graphology*) y la *grafoterapia* [36][37]. La grafopatología estudia el reflejo de las patologías en la escritura de la persona. Para la grafología la escritura de una persona representa una proyección de su personalidad y por tanto su estudio permite analizar su psique. Finalmente, la grafoterapia intenta cambiar hábitos en la escritura de un sujeto para tratar un problema psicológico. El valor de estas disciplinas, especialmente la grafología y la grafoterapia, está cuanto menos cuestionado desde hace tiempo por numerosos estudios científicos [38][39].

## 2.2 Problemática relativa al tratamiento automático de las firmas

A grandes rasgos, como ya se ha adelantado en el Capítulo 1, en el tratamiento automático de firmas *off-line* se pueden distinguir cuatro áreas:

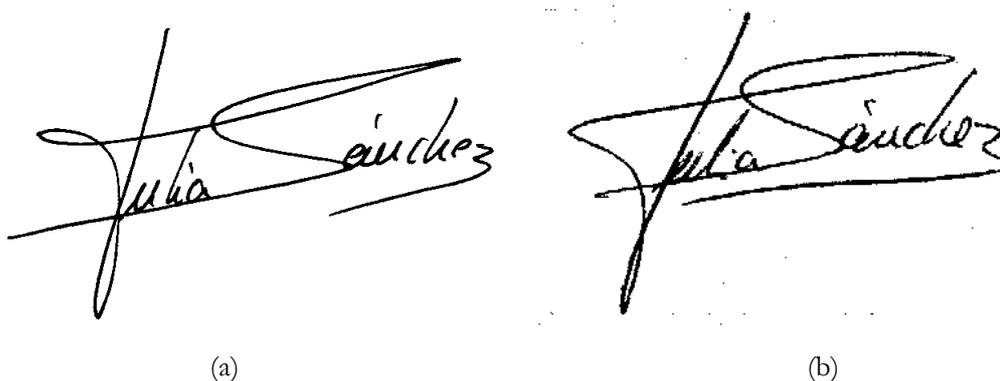
- Captura.- Trata de determinar cómo se debe digitalizar un documento que contiene una firma.
- Segmentación o reconocimiento.- Consiste en localizar dentro de una imagen digitalizada una o más firmas que en ella aparezcan.
- Verificación.- Dada la imagen de una firma y su pretendido autor, se trata de comprobar si la firma es auténtica o se trata de una falsificación.
- Identificación.- Dada la imagen de una firma se trata de averiguar cuál es su autor, dentro de un conjunto de candidatos.

En los siguientes apartados se analiza en detalle la problemática asociada a cada una de estas áreas. Por motivos de exposición se analiza en primer lugar la problemática relativa a la verificación y a la identificación de firmas, pues su análisis introduce un contexto imprescindible para concretar la problemática relativa a la captura y a la segmentación los cuales se abordan en último lugar.

### 2.2.1 Problemas en la verificación automática *off-line* de firmas

Ya se ha comentado que la verificación tiene por objeto determinar si la firma contenida en una imagen ha sido realizada por la misma persona que realizó otras firmas que se usan como modelo. Básicamente, el proceso de verificación consiste en medir el grado de similitud entre las diferentes firmas. Sin embargo, como se puede apreciar en el ejemplo de la Figura 2.2, el principal escollo al construir un sistema de verificación de firmas *off-line* consiste en que las personas nunca reproducen su firma de manera exacta. No obstante, se espera que las diferencias entre dos firmas de la misma persona sean pequeñas (o de diferente índole) comparadas con las diferencias entre dos firmas de personas diferentes. E. Justino [40] enuncia que: “en el problema de verificación de firmas se trata de maximizar las *diferencias interpersonales* y de minimizar las *diferencias intrapersonales*”.

Al margen del problema de medir el grado de similitud entre las formas, en la construcción y en la operativa de un sistema de verificación *off-line* en condiciones realistas suelen darse problemas de otra naturaleza como: la escasez de muestra, la escalabilidad del sistema, la automatización del proceso de aprendizaje y el tratamiento de los errores del sistema. Todos estos problemas se describen detalladamente en los siguientes puntos.



**Figura 2.2.-** Ejemplo de la variabilidad en la firma de una persona. La imagen (a) se capturó en el año 2001, la (b) en el 2004.

### Variabilidad intrapersonal

La mayoría de los modelos que explican la forma en que una persona escribe consideran el acto de firmar como un proceso de movimiento controlado que no necesita realimentación visual sobre la posición de los miembros [4][27]. Así, la firma constituye un proceso de escritura rápida cuyo patrón de movimiento ha sido previamente aprendido, lo que hace innecesaria la supervisión por el cerebro de los movimientos que la producen. Por ello se afirma que el proceso de firmar es un movimiento pseudo-reflejo o movimiento disparado, en contraste con los movimientos supervisados.

Como todas las acciones humanas, las firmas que realiza un individuo presentan cierta variabilidad. Dos firmas de un mismo individuo nunca son iguales [10]. Además, esta variabilidad se puede acentuar debido a diferentes factores ocasionales (como el tipo de útil de escritura, el papel y demás elementos descritos en 2.1.4) y permanentes (como enfermedades o la edad) [11][31]. Al conjunto de estas variaciones se las denominan variaciones intrapersonales. En general, un sistema de verificación de firmas no debería dejarse influir por estas variaciones intrapersonales, pues no se derivan de un cambio de firmante.

A modo de ejemplo, en la firma de la Figura 2.2 se aprecian varias diferencias intrapersonales: el trazo de la “z” final se alarga al doble de una firma a otra, la visera se acorta, una filigrana de la “j” inicial desaparece, etc. Como señala E. Justino, estas diferencias intrapersonales deben ser minimizadas y obviadas por un sistema de verificación.

### Variabilidad interpersonal

La variabilidad interpersonal se refiere a las diferencias en cuanto a morfología y a factores caligráficos entre los diferentes individuos. Por otro lado, esta variabilidad intenta ser disimulada por los falsificadores que desean cometer fraude imitando una firma ajena. Un sistema de verificación de firmas debería detectar las variaciones interpersonales entre firmas.

Señalan M. Ammar y otros [18] que los falsificadores, que pueden tratar de engañar a un sistema de verificación se catalogan en dos grandes grupos: los falsificadores hábiles (*skilled*) y los aleatorios (*random* o *casual*).

Los falsificadores hábiles pueden conocer la firma de la persona a quien quieren suplantar, y pueden entrenarse en reproducir la firma para conseguir unas falsificaciones de gran calidad. Los aleatorios no han visto nunca la firma de la persona a la que quieren suplantar, por lo que su reproducción no suele guardar ningún parecido con la firma auténtica. Como ya se ha señalado, un estudio, realizado en el entorno bancario brasileño, revela que el 99% del fraude corresponde a falsificadores aleatorios [19][11].

Algunos autores dividen a los falsificadores hábiles en diferentes grupos: falsificadores *simples*, *simulados*, *expertos* y *calcadores*. Los simples conocen el nombre de la persona a la que tratan de suplantar, pero no conocen su firma. Los simulados (*simulated* o *freehand*) han visto la firma brevemente e incluso pueden haberla ensayado. Los expertos (*experts*) han tenido tiempo para estudiar la firma y tienen experiencia en imitación de firmas. Finalmente, los calcadores han realizado la falsificación mediante el calcado de una firma original.

### Escasez de muestra

Uno de los problemas de la verificación de firmas en entornos reales consiste en la dificultad de obtener múltiples firmas de un mismo individuo para estudiar su variabilidad. Por ejemplo, cuando alguien abre una cuenta en un banco se le puede pedir que firme una o dos veces. Sin embargo, es muy probable que dicha persona se niegue si, con objeto de tener una muestra estadísticamente

significativa, se le pide firmar 10 ó 15 veces con diferentes bolígrafos a lo largo de diferentes semanas.

Sería posible idear un sistema que en principio no fuese capaz de verificar la firma de un individuo, pero que tras unos meses, cuando ya se hubiesen recogido varias muestras reales del mismo, fuese capaz de hacerlo. Desafortunadamente, este enfoque plantea varios problemas: por un lado, mientras se recogen las firmas el sistema no es capaz de verificar y esto puede significar meses hasta estar operativo para un individuo; por otro lado las firmas recogidas seguramente no estarán libres de ruido, al provenir de documentos reales, y no podrían utilizarse en ningún proceso de aprendizaje; finalmente, las firmas recogidas durante la vida operativa, la cuales se pretenden usar en el proceso de aprendizaje, podrían no ser auténticas, por lo que sería necesario un sistema de verificación previo que utilizase menos muestra de aprendizaje. Estos problemas crean la necesidad de un sistema que, con una sola firma de aprendizaje, permita verificar una firma manuscrita. Además, teniendo en cuenta que cualquier persona, con un mínimo grado de entrenamiento, es capaz de realizar la verificación de una firma con un alto grado de acierto, no parece descabellado proponer que esta tarea se realice de manera automática.

Se puede concluir que, para construir un sistema de verificación real no es viable solicitar muchas firmas a cada usuario. Es más, en general sólo suele disponerse de una firma por persona. Afortunadamente, es factible que la firma que se utilice para construir el sistema esté libre de ruido y en una posición conocida, ya que se captura en condiciones muy controladas (por ejemplo en una oficina bancaria y sobre un papel especialmente diseñado para tal efecto).

### **Escalabilidad y automatización del proceso de aprendizaje**

Un sistema real debería poder crecer en cuanto al número de individuos que es capaz de verificar. Este crecimiento debería realizarse de manera dinámica, sin que cambien los tiempos necesarios para incorporar un nuevo individuo o para verificar un ejemplar. Además, debería funcionar de manera autónoma, sin supervisión, tanto en la etapa de aprendizaje de nuevos individuos como, por supuesto, en la etapa de verificación.

Hay sistemas que, al añadir nuevos individuos para que puedan ser verificados, necesitan recalcular ciertos parámetros internos. Para ello, se debe realizar un nuevo proceso de aprendizaje que involucre a los individuos nuevos y a los antiguos. Además, estos procesos de aprendizaje suelen emplear más tiempo cuantos más individuos incluya el sistema. En situaciones en las que puede haber cientos de miles de potenciales usuarios, como ocurre en el entorno bancario, estos sistemas no son utilizables por razones obvias.

También hay sistemas que, para aprender a verificar una firma, precisan de un proceso de aprendizaje supervisado por una persona especializada. Tal puede ser el caso del entrenamiento de algunos tipos de Redes de Neuronas. Este proceso, que puede durar un tiempo variable, termina cuando a juicio del experto se alcanza la capacidad de generalización necesaria para interrumpir el aprendizaje. Estos sistemas tampoco son utilizables en entornos reales, a no ser que el proceso de aprendizaje se realice sólo una vez para todo el sistema.

### **Errores del sistema**

Señalan Leclerc y Plamondon [41] que el resultado de cualquier trabajo de investigación en el campo de la verificación de firmas debe presentar un análisis de dos tasas de error que se conocen como *Tasa de Falso Rechazo* (*False Rejection Rate* en adelante *FRR*) y *Tasa de Falsa Aceptación* (*False Acceptance Rate* en adelante *FAR*) [17]. La primera tasa, *FRR*, que la estadística clásica etiqueta como *error de tipo I*, mide la proporción de errores que se producen al rechazar firmas auténticas. La segunda tasa,

$FAR$ , que también se conoce como *error de tipo II*, mide la proporción de error que se comete al aceptar como auténticas firmas que son falsas.

Es habitual que los sistemas desarrollados permitan obtener diferentes valores para el  $FRR$  y el  $FAR$  en función de un parámetro. Por ello los resultados de los trabajos suelen presentarse mediante una gráfica como la de la Figura 2.3 (a). En esta gráfica se utiliza una curva para representar el porcentaje de error relativo al  $FRR$  y otra para el porcentaje de error relativo al  $FAR$ . Estas curvas se originan mediante la variación de un parámetro, que se presenta en el eje  $x$ , y que habitualmente se normaliza entre 0 y 1. El punto donde se cortan ambas curvas se denomina  $EER$  (*Equal Error Rate*) y suele tomarse como elemento de referencia al comparar dos sistemas de verificación.

Como presenta la Figura 2.3 (b), también es posible unir estas dos curvas en una, obviando el valor del parámetro. Esta representación se denomina curva  $ROC$  (*Receiver Operating Characteristic*), y en ellas, el punto  $EER$  se encuentra siempre en el corte con la diagonal que parte del origen, ya que en esta diagonal los valores de  $FRR$  y  $FAR$  son iguales. Obsérvese que, cuanto más cerca de los ejes de abscisa y ordenada esté la curva  $ROC$  mejores son sus resultados. Este hecho facilita la comparación de dos o más sistemas de verificación sin recurrir a la simplificación de comparar los puntos  $EER$ . Así, un sistema A es mejor que otro B si la curva  $ROC$  de A queda por debajo de la de B, es decir más próxima a los ejes cartesianos.

Aparte de estos gráficos, lo que el usuario de un sistema automático desea percibir es un sistema carente de errores. Por ejemplo, un cliente de un banco no desea que se rechacen sus cheques al no verificarse correctamente su firma. Por otro lado, el banco tampoco desea crear molestias innecesarias a sus clientes, rechazando firmas auténticas. Por ello, en algunos casos, puede ser preferible que el sistema no dé respuesta a que rechace una firma incorrectamente. Aún más, contrariamente a lo que pudiese pensarse, puede ser preferible que el sistema acepte una firma falsa a que rechace una firma verdadera (siempre que el coste económico del engaño no supere un umbral). Esto se debe a que previamente a la introducción del sistema no se analizaba la firma (o sólo se hacía en caso de importe elevado) y por lo tanto el coste de trabajo en este sentido era cero. Además, un sistema automático de verificación resuelve un trabajo que antes nadie resolvía, por tanto los problemas que origina el sistema crean “nuevas molestias”. Esto no suele ser aceptado por usuarios que “antes no tenían ese problema”. De todo esto se desprende que, en todo caso, debe optimizarse la relación entre el  $FRR$ , el coste que introduce su tratamiento y el coste derivado del fraude.

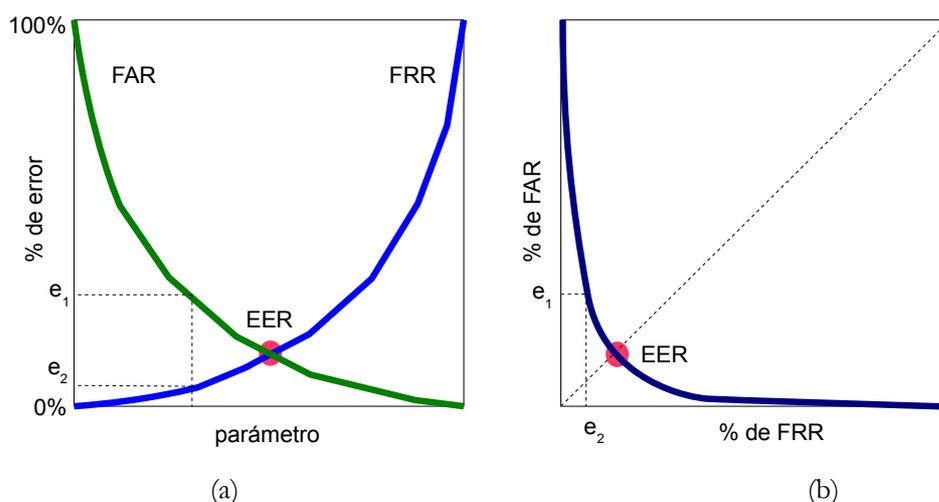


Figura 2.3.- Curvas utilizadas para presentar los resultados de un sistema de verificación de firmas.

### 2.2.2 Problemática de la identificación automática *off-line* de firmas

El problema de la identificación *off-line* de firmas trata de decidir a qué individuo pertenece una firma dentro de un conjunto prefijado de individuos conocidos. Así, ante una firma que se desea identificar, el sistema señala al individuo (o individuos) cuyas firmas son más parecidas de entre las que conoce.

Desde cierto punto de vista se puede decir que el problema de verificación y el de identificación corresponden al mismo problema, aunque la identificación tiene una formulación 1 a  $N$  y la verificación tiene una formulación más restringida de tipo 1 a 1. Sin embargo, los métodos utilizados para la verificación y los utilizados para la identificación son diferentes. Estas diferencias se basan en que en la verificación se sabe la identidad del individuo que dice ser el autor de la firma, y por tanto se dispone de más información que en la identificación, donde esa información no se conoce.

Desde el punto de vista de su aplicación industrial, ya ha quedado patente que la verificación de firmas es más importante que la identificación debido a la multitud de entidades financieras que precisan automatizar sus sistemas de autenticación de firmas. La identificación de firmas es un problema que encuentra una aplicación más restringida. A pesar de ello, se pueden plantear varios escenarios en los que es interesante recuperar de una base de datos aquellas firmas similares a una dada. Por ejemplo, un sistema de identificación, puede ser útil como parte de un sistema de verificación bancario, cuando una cuenta pueda tener varias firmas reconocidas. También puede ser útil en aplicaciones para la identificación automática de escritos históricos o en aplicaciones forenses.

Desde un punto de vista científico, no está del todo clara la dificultad relativa de estos dos problemas. Así, hay autores que consideran más difícil el problema de la identificación [16], mientras que otros consideran más difícil el problema de la verificación [2]. La mayor dificultad de los procesos de verificación se justifica porque en los procesos de identificación no hay que preocuparse por la variabilidad intrapersonal. Sólo hay que señalar las variaciones interpersonales que permiten asignar a una firma a un individuo en vez de a otro. Sin embargo, es cierto que en la identificación se dispone de menos información. Además, conforme crezca el tamaño de la base de datos de firmas de un problema de identificación, la tarea se complica al aumentar la probabilidad de parecido entre ejemplares de diferentes individuos. Incluso un sistema de identificación podría necesitar incluir un sistema de verificación si tuviese que protegerse frente a intentos de intrusión.

### 2.2.3 Problemática de la digitalización

Al digitalizar un documento se deben de tomar ciertas decisiones en cuanto a la configuración del dispositivo de captura. Básicamente, estas decisiones se centran en la elección de la *resolución espacial* y la *resolución radiométrica*. Dado que el tamaño de las imágenes generadas puede ser importante, también debe considerarse el algoritmo de compresión utilizado para almacenarlas de manera eficiente en disco o para transmitir las por cualquier medio. En los siguientes puntos se analizan en detalle estos aspectos.

#### Elección de la resolución espacial

La frecuencia a la que se realiza el *muestreo* sobre el documento físico, que se mide en número de muestras por unidad de espacio, da lugar al concepto de resolución espacial, que suele medirse en *puntos por pulgada* (*dots per inch*, o abreviadamente *DPIs*).

Es evidente que cuanto más alta sea la resolución espacial más alta será la calidad de la imagen obtenida. Por otro lado, una resolución espacial insuficiente puede hacer que los contornos de los objetos a reconocer sean imprecisos. La conjetura de muestreo de Nyquist (posteriormente

convertida en teorema por Shannon) establece la frecuencia mínima que es preciso aplicar para poder recuperar sin errores una señal [25]. El teorema establece que la frecuencia de muestreo debe ser al menos el doble de la máxima frecuencia con la que cambian los elementos que se quieren capturar en una señal. Por ejemplo, si se ha impreso una hoja de papel con una impresora configurada a 200 DPIs, el proceso de escaneo, necesario para obtener una representación fiel de los datos allí impresos, debe realizarse muestreando al menos a 400 DPIs.

Por otro lado, en los dispositivos de digitalización de sobremesa son habituales resoluciones de 200 y 300 DPIs<sup>4</sup>. Mientras, en los dispositivos de tipo fax la resolución suele disminuir a 150 ó 75 DPIs. Finalmente, deseamos llamar la atención del lector sobre la Figura 2.4. En ella se muestra la imagen de una firma capturada con diferentes resoluciones espaciales. Se puede apreciar en este ejemplo que la resolución espacial en los 3 casos parece suficiente para que una persona pudiese realizar las tareas de verificación o de identificación, como afirman Ramesh y otros [23].

### Elección de la resolución radiométrica

La *cuantización* de una imagen consiste en la discretización de los posibles valores que puede tomar cada uno de sus píxeles. Los niveles de cuantización suelen ser potencias de 2 para facilitar el almacenamiento y el procesamiento de las imágenes en un computador. El número de posibles niveles de cuantización define la resolución radiométrica.

Si sólo se permiten dos niveles de cuantización (normalmente blanco y negro) se habla de *imágenes bitonales* o imágenes binarias. Cuando las imágenes sólo tienen información sobre el brillo se habla de *imágenes en niveles de gris*. En este caso se suelen utilizar hasta 256 niveles para representar los tonos intermedios desde el negro (0) hasta el blanco (255). Para las imágenes a color suelen usarse 256 niveles para representar la intensidad de cada uno de tres matices primarios<sup>5</sup> o bandas de color. De esta forma se obtienen aproximadamente 16 millones de colores (256x256x256) y se habla de *imágenes en color real* o *imágenes RGB*. En algunos casos, en los que se necesita mayor resolución radiométrica, se usan 4096 niveles por banda de color o incluso más.

Obsérvese que una imagen en color real sin compresión, por ejemplo cuando se muestra en la pantalla de un ordenador o cuando se almacena en una variable para su tratamiento, ocupa 3 veces más memoria que una en niveles de gris y 24 veces más que una bitonal. Por su parte, una imagen en niveles de gris sin compresión ocupa 8 veces más que una bitonal.

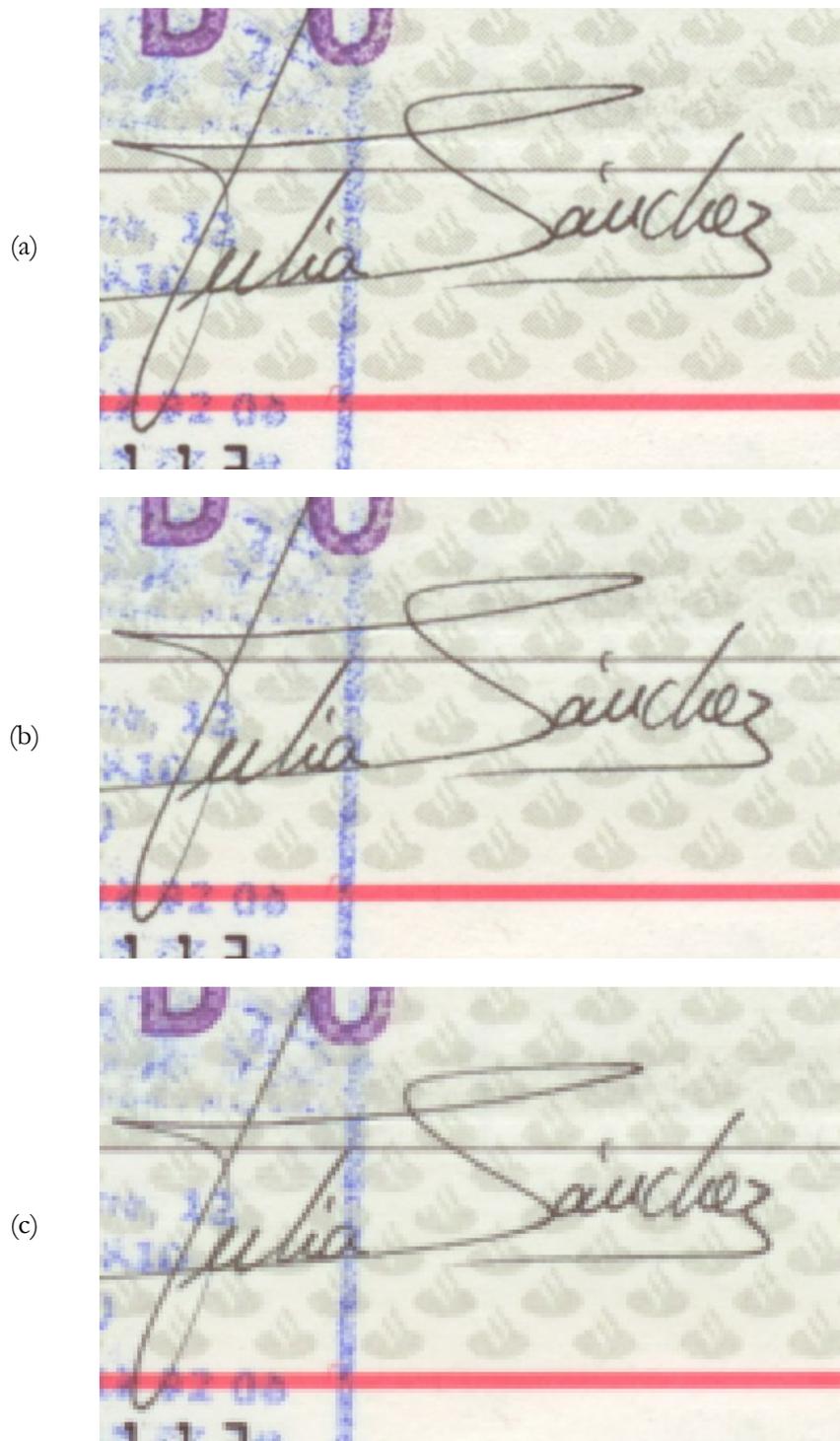
Evidentemente, la segmentación de una imagen es más sencilla cuando la información aportada por cada píxel es mayor, y por tanto cuando la resolución radiométrica es más elevada. Así, por ejemplo, parece más sencillo eliminar el sello en la imagen 2.5 (a) que en la imagen (b) debido a que se puede distinguir por la diferencia de color.

En la Figura 2.5 puede observarse cómo parece posible que una persona segmente y reconozca todos los elementos del cheque de la imagen en cualquiera de las tres elecciones de resolución radiométrica. Sin embargo, puede apreciarse que en la imagen bitonal se elimina información del trazo de la firma al eliminarse los diferentes grados de intensidad del mismo. Esta información, que se pierde, podría ser relevante en un proceso de verificación.

---

4 Aunque hay dispositivos que dicen ofrecer más de 300 DPIs, hay que señalar que en los más económicos su óptica y su *hardware* no permiten esa resolución, sino que utilizando técnicas de interpolación aumentan de manera artificial la resolución. Lógicamente, estos aumentos artificiales de resolución no aportan información nueva al sistema.

5 Estos matices se corresponden, de manera aproximada, a un matiz rojo, uno verde y uno azul, por lo que las imágenes obtenidas se suelen denominar imágenes RGB (*red, green, blue*). La teoría trístímulo del color establece estadísticamente que mediante la mezcla aditiva de tres linternas con estos matices primarios se obtiene un amplio rango de matices.



**Figura 2.4.-** Imagen de una firma capturada a diferentes resoluciones espaciales: (a) 300 DPIs, (b) 200 DPIs y (c) 100 DPIs.



Figura 2.5.- Imágenes de un cheque capturadas a diferentes resoluciones radiométricas. La imagen (a) está en color real (RGB 24 bits por píxel), la (b) en niveles de gris (8 bits por píxel) y la (c) es bitonal (1 bit por píxel).

La elección de la resolución radiométrica no sólo se limita a decidir el número de *bits* por píxel. En el caso bitonal puede existir la posibilidad de seleccionar el tipo de *binarización* que se realizará, ya que los sensores de los dispositivos suelen capturar la información en niveles de gris o en color. El

método de binarización más sencillo se denomina de *binarización global*, y consiste en fijar un valor umbral fijo para que, si un píxel lo supera, se etiquete como blanco y en caso contrario como negro. Sin embargo, señala J. R. Parker [42] que en el caso de documentos manuscritos o mecanografiados la mejor alternativa suele consistir en el uso de la *binarización adaptativa (moving averages thresholding)*. Este tipo de binarización se caracteriza por utilizar un umbral diferente para cada píxel, que se decide en cada momento en función del valor del píxel a binarizar y de los valores de los píxeles de su entorno.

Hay que señalar que, al escanear en color o en niveles de gris, los programas de captura suelen proporcionar funciones de transferencia sobre el histograma. Estas funciones tienen por objeto aumentar el contraste aparente (que no real) de las imágenes que se obtengan. Estas transformaciones se realizan por software y son irreversibles. Por ello, es preferible que no se apliquen, ya que implican una pérdida de información, a no ser que el único objetivo de la captura sea su visualización.

### Elección del formato de compresión de imágenes

Como se puede apreciar en las tablas 2.1 y 2.2 los tamaños en *bytes* de las imágenes correspondientes a las resoluciones altas son elevados incluso en los cheques, que son documentos con un tamaño físico bastante reducido. Estos tamaños tan elevados hacen que los algoritmos de tratamiento de imagen sean costosos, los tiempos de transmisión altos y que los tiempos de almacenamiento y recuperación desde una memoria secundaria sean elevados.

Para evitar estos inconvenientes se suelen utilizar algoritmos de compresión de imágenes. Los algoritmos de compresión se basan en el concepto de eliminación de la redundancia. González y Woods [25] clasifican la redundancia en tres tipos: redundancia en la codificación, redundancia en la representación y redundancia visual.

La eliminación de la *redundancia en la codificación* consiste alterar el alfabeto que se utiliza para almacenar la información, de manera que se asigne códigos más cortos a las palabras más probables. Quizás los algoritmos de eliminación de redundancia en la codificación más utilizados son ZIP y Huffman. Estos algoritmos no modifican la información sólo la comprimen. Además se pueden utilizar sobre cualquier tipo de datos, ya que no se apoyan en el tipo de información que se está comprimiendo.

DPIs	Píxeles Ancho	Píxeles Alto	Bits x Pixel	Tamaño (MB)
100	654	268	1	0,17
			8	1,4
			24	4,2
200	1307	535	1	0,7
			8	5,6
			24	16,8
300	1961	803	1	1,57
			8	12,6
			24	37,79

**Tabla 2.1.-** Tamaño en *bytes* obtenido al digitalizar un cheque de dimensiones 16.6 X 6.9 cms.

DPIs	Pixeles Ancho	Pixeles Alto	Bits x Pixel	Tamaño (MB)
100	827	1142	1	0,94
			8	7,55
			24	22,65
200	1654	2283	1	3,78
			8	30,21
			24	90,62
300	2480	3425	1	8,5
			8	67,96
			24	203,89

**Tabla 2.2.-** Tamaño en bytes obtenido al digitalizar un A4 (21 x 29 cms).

La eliminación de la *redundancia en la representación* se basa en el hecho de que el elemento a comprimir es una imagen. Los algoritmos de este tipo utilizan la alta correlación que existe entre los píxeles de una imagen para codificar la información de una manera menos redundante. Ejemplos de estos tipos de algoritmos de compresión son los basados en *rachas de píxeles (runlengths)* como CCITT<sup>6</sup> Grupo 3 y 4.

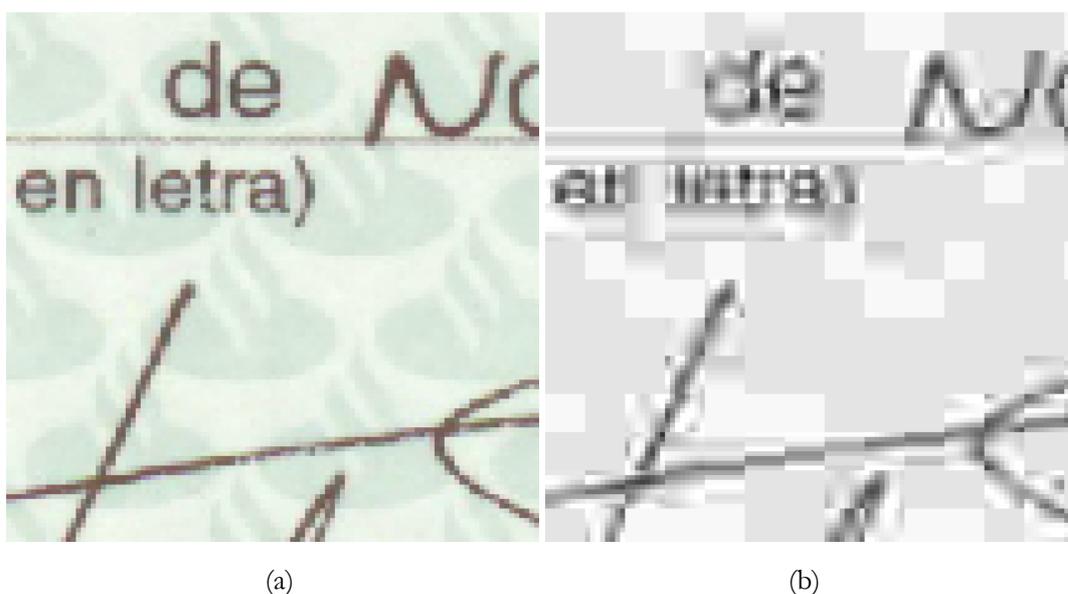
Idealmente, los métodos de eliminación de *redundancia visual* modifican la imagen de manera visualmente imperceptible con objeto de obtener una representación que permita mayor compresión. Esta modificación de la imagen supone una pérdida de información irreversible respecto a la imagen original, y por ello estos métodos se conocen como *compresores con pérdida* o con error. Esta pérdida constituye la diferencia fundamental con los métodos precedentes, ya que en aquéllos no se alteraba la información de la imagen, simplemente se reducía la cantidad de datos necesarios para representarla. El algoritmo de eliminación de redundancia visual más utilizado es en la actualidad JPEG<sup>7</sup>.

Se debe añadir que los algoritmos de compresión pueden tener como desventaja el aumento del tiempo de acceso a la imagen, al ser superior el tiempo empleado para descomprimir la imagen, que el tiempo utilizado para leer la imagen si se hubiese almacenado sin compresión.

Los algoritmos de compresión con pérdida permiten reducir el tamaño de los ficheros de las imágenes en ratios muy elevados. De hecho la transmisión de imágenes por Internet, como la conocemos hoy en día, no sería posible sin este tipo de algoritmos. Sin embargo, estos algoritmos introducen artefactos en las imágenes de documentos (ver Figura 2.6) que no afectan demasiado a los procesos de percepción humanos, pero sí lo hacen sobre los algoritmos de reconocimiento automático [43]. Esto ha motivado que se desarrollen algoritmos específicos de compresión con pérdida que minimicen estos efectos perniciosos (como JPEG2000 o *Djvu*). Desafortunadamente, estos formatos están protegidos por patentes que han restringido su difusión, o son computacionalmente demasiado costosos o no ofrecen el nivel de compresión requerido.

6 CCITT son las siglas de Comité Consultivo Internacional Telegráfico y Telefónico, antiguo nombre del Comité de Normalización de las Telecomunicaciones dentro de la UIT (Unión Internacional de Telecomunicaciones) ahora conocido como UIT-T. Este comité definió los estándares de compresión de imágenes G3 y G4 basados en *runlengths*. Estos algoritmos se utilizaron primero en la transmisión por fax y luego dentro de los archivos TIFF (*Tagged Image File Format*) para imágenes bitonales.

7 El acrónimo JPEG corresponde a *Joint Photographic Experts Group*, que a su vez es el nombre del comité que en 1986 se propuso crear un estándar para eliminación de redundancia visual. El algoritmo JPEG fue aprobado en 1994 como la norma ISO 10918-1.



**Figura 2.6.-** Ejemplo de artefactos introducidos por el formato de compresión JPEG. En (a) la imagen sin comprimir, en (b) la misma imagen utilizando un alto índice de compresión.

### La digitalización en un entorno de producción real

Como se ha comentado, la verificación automática *off-line* de firmas es un problema interesante en ambientes como el bancario donde sería preciso verificar millones de firmas diariamente. De hecho, ya existen instalaciones que procesan millones de documentos bancarios, con objeto de extraer información, como el importe o el pagador, de los mismos [44].

Estas instalaciones de alto rendimiento siguen un esquema similar al de la Figura 2.7. En ella se puede apreciar un flujo que comienza con la digitalización de los documentos en papel. Tras esto, los documentos, ya en formato digital, son procesados por los sistemas de reconocimiento automático. Si estos sistemas capturasen toda la información requerida de los documentos el proceso prácticamente habría finalizado. Sin embargo, los actuales sistemas de reconocimiento habitualmente no son capaces de extraer toda la información deseada de manera automática, por lo que se precisa una etapa de grabación manual. En esta etapa se presenta cada documento digital a uno o varios operadores especializados. Éstos extraen la información requerida del documento, mediante su inspección visual en una pantalla, y la teclean en un terminal. Finalmente, una vez que toda la información deseada ha sido capturada se exporta a bases de datos de explotación. El diagrama de la Figura 2.8 describe más formalmente el proceso realizado en estas instalaciones.

Las instalaciones de este tipo suelen tener ventanas de tiempo muy pequeñas para procesar ingentes cantidades de documentos. Por ejemplo, en una gran instalación, con varios sistemas de digitalización de alto rendimiento y con decenas de terminales para la grabación manual de datos, se pueden procesar 1 ó 2 millones de documentos en 5 ó 6 horas. En este contexto los tamaños de las imágenes y de los ficheros que las almacenan cobra vital importancia, ya que si tienen un tamaño demasiado grande podrían colapsar el tráfico de red o la capacidad de almacenamiento de los servidores.

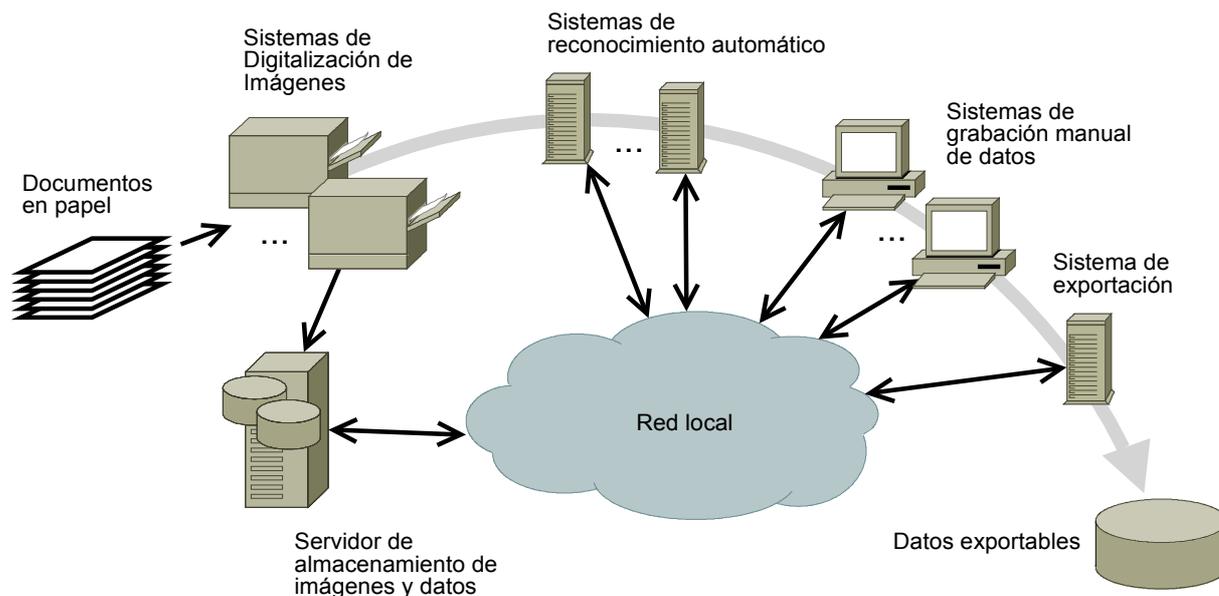


Figura 2.7.- Esquema de una instalación de captura masiva de información sobre documentos.

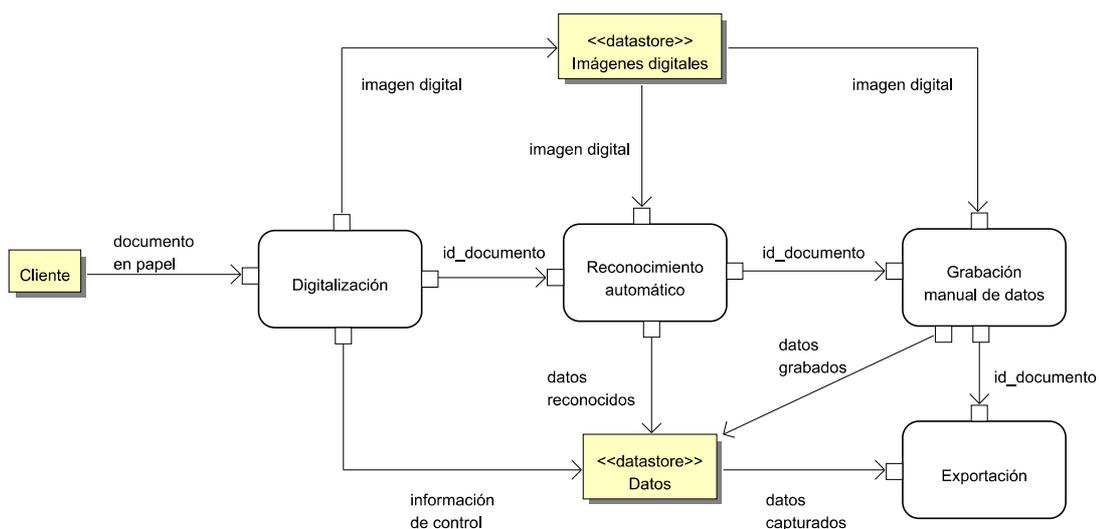


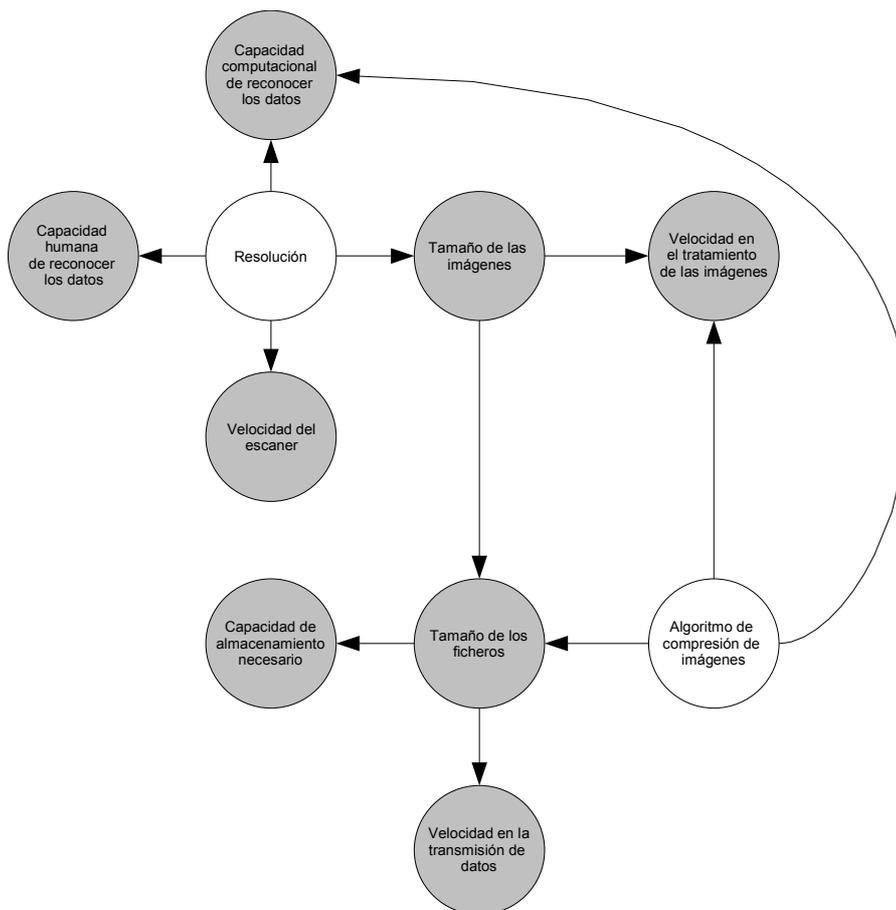
Figura 2.8.- Diagrama típico para el flujo de información en una instalación de captura masiva de datos sobre documentos.

A continuación, se resumen los diferentes factores (recogidos gráficamente en la Figura 2.9) que se tienen en cuenta a la hora de construir una instalación de captura masiva de documentos:

- El tamaño de las imágenes viene determinado por dos factores: la resolución espacial y la resolución radiométrica.
- La resolución espacial y la resolución radiométrica influyen en la capacidad de las personas y de las máquinas para reconocer los datos presentes en los documentos.

- El tamaño de las imágenes influye en el tiempo que emplea un programa en presentar la imagen a un operador humano. Esta influencia se debe principalmente a dos factores: la mayor cantidad de memoria necesaria y el mayor volumen de datos a manejar.
- El tamaño de los ficheros viene determinado por el tamaño de las imágenes y por el algoritmo de compresión que se elija para las mismas.
- El tamaño de los ficheros influye en la capacidad de los dispositivos de almacenamiento y en la velocidad de la red local que se utiliza para transferir las imágenes.
- El tipo de algoritmo de compresión que se elija también influye en el tiempo que emplea un programa para presentar la imagen a un operador humano, ya que si el algoritmo es complejo empleará mayor cantidad de tiempo.
- La imagen debe tener una calidad suficiente para que un operador humano pueda reconocer los elementos presentes en ella.
- La imagen debe tener una calidad suficiente para que los algoritmos de reconocimiento automático de imágenes puedan operar sobre ella con éxito.

En un compromiso entre rendimiento y calidad visual, a día de hoy, las principales instalaciones que operan en el mundo utilizan imágenes bitonales muestreadas a 200 DPIs, almacenadas con compresión sin pérdida y umbralizadas mediante algoritmos adaptativos [44].



**Figura 2.9.-** Factores que se tienen en cuenta al construir una instalación. En gris se representan las consecuencias, en blanco los puntos de control y las flechas indican influencia.

## 2.2.4 Problemas en la segmentación y el reconocimiento

La etapa de segmentación consiste en la localización de los píxeles correspondientes a la firma dentro de la imagen digitalizada de un documento.

La detección de la posición de la firma dentro del documento y la eliminación del ruido adyacente son los principales problemas que se abordan en esta etapa. El ruido se debe principalmente a elementos estructurales presentes en el documento que están próximos o se superponen a la firma. Estos inductores de ruido habitualmente corresponden a: dibujos o textos de fondo, logotipos, sellos, líneas de apoyo para la escritura, texto mecanografiado, sellos y anotaciones manuscritas. En los siguientes puntos se analizan en detalle cada uno de estos problemas.

### Posición de la firma

La firma en los documentos puede aparecer en posiciones diversas. Sin embargo, ya que la firma suele implicar la aceptación de unas condiciones que se han expuesto en párrafos previos dentro del documento, es común que aparezca en la parte inferior de los mismos.

### Existencia de líneas

En la mayoría de los casos, la firma suele realizarse dentro de un rectángulo o sobre una línea horizontal dispuesta a tal efecto dentro del documento (ver Figura 2.10). Estas líneas, cuya posición puede variar de un documento a otro, introducen estructuras que, al solaparse con la firma, pueden afectar al sistema de reconocimiento.

Como se puede apreciar en la Figura 2.10, sobre una imagen digitalizada puede resultar difícil distinguir una de estas líneas de apoyo de los trazos correspondientes a la firma. Si la segmentación se realiza incorrectamente, y entre los puntos de la firma se incluyen los puntos de estas líneas, el proceso de verificación posterior podría fallar al encontrarse con una línea que no pertenece a la firma.

Algunas características de estos elementos son:

- Suelen ser líneas perfectamente horizontales y no conectadas a ningún objeto preimpreso del resto del documento.
- También podrían ser rectángulos con la dimensión horizontal de mayor tamaño que la vertical.

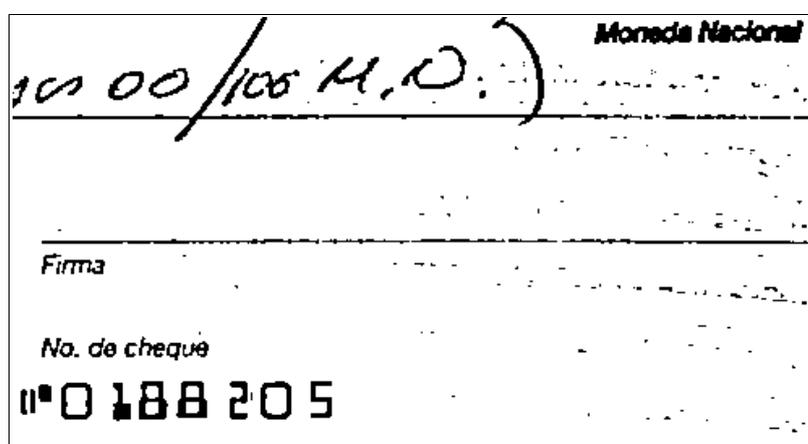


Figura 2.10.- Ejemplo de una línea sobre la que se requiere la firma.

Hay que añadir que algunas firmas contienen segmentos rectos como parte de las mismas. Por ello, la eliminación sin más de todos los segmentos rectos de una imagen no es una solución carente de problemas, pues el proceso podría estar destruyendo la firma que se desea verificar.

### **El fondo del documento**

Habitualmente, los documentos bancarios suelen tener fondos pictóricos. Otros tipos de documentos que también suelen tener estos fondos son los contratos, las escrituras y las facturas. Estos fondos pueden consistir en una única imagen (ver Figura 2.11) o en algún elemento que, repetido a forma de mosaico, tesela el documento (ver Figura 2.5).

Como se ha explicado en 2.1.4 el objetivo original de estos fondos pictóricos es el de dotar al documento de mayor seguridad, ya que dicho fondo permite que se aprecien las raspaduras o borrados que pudiesen realizarse al intentar cometer un fraude.

En el caso de repetición de una figura su complejidad es variable, pudiendo consistir en la repetición de un sencillo patrón geométrico (como las líneas de la Figura 2.12), de un logotipo (como el de la Figura 2.5) o incluso de un texto. Esta trama habitualmente se realiza utilizando tonos pasteles para facilitar la lectura del texto que se le superponga. Sin embargo, en imágenes bitonales, tras los procesos de umbralizado la trama suele persistir en forma de ruido que se mezcla con la firma y que dificulta su segmentación (ver Figura 2.12 (c)).

### **Existencia de sellos**

Los sellos de caucho son otra fuente habitual del ruido que suele acompañar a las firmas en los documentos. Las figuras producidas por estos sellos suelen incluir textos, rectángulos y círculos. Debido a que el sello se pone a mano, estos elementos aparecen en el documento con orientación diversa. Además, en muchos casos se fuerza la superposición a la firma para reivindicar autenticidad, ya que implica que la persona que ha sellado el documento ha visto la firma o viceversa.

Los sellos en imágenes en color pueden discriminarse más fácilmente debido a los cambios de color (ver Figura 2.13 (a)). Sin embargo, tras la umbralización, cuando las imágenes están en niveles de gris o son imágenes bitonales, la discriminación no es sencilla (ver Figura 2.13 (b)).

### **Existencia de texto preimpreso o mecanografiado**

Diverso texto de origen mecánico suele aparecer sobre o debajo de la firma (ver Figura 2.14). Esta superposición suele realizarse, al igual que en el caso del sello, para reivindicar la autenticidad de la parte del documento afectada. De hecho, un análisis minucioso del documento puede revelar, en el caso de que exista superposición, el orden en que se han realizado las operaciones de sellado, impresión y firmado.

De nuevo, cuando se trata de imágenes a color suele ser sencilla su eliminación, pues el color de la firma suele ser muy diferente al de estos elementos. Desgraciadamente, en el caso de imágenes bitonales sólo se puede utilizar la información relativa a la forma de los objetos, lo cual dificulta el proceso enormemente.



Figura 2.11.- Imagen de un cheque con un fondo pictórico.



Figura 2.12.- Trama de fondo de un documento en forma de líneas oblicuas. Obsérvese cómo la dificultad de separar la trama de la firma aumenta desde una imagen en color (a) a la misma imagen en grises (b) y en bitonal (c).



Figura 2.13.- Imágenes de firmas sobre las que se aplicado un sello: (a) imagen en color y (b) imagen bitonal.

### Existencia de texto manuscrito

Algún texto manuscrito también suele aparecer cerca de la firma e incluso sobre ella (ver Figura 2.15). Este texto puede tener su origen en la disposición de los datos dentro del documento, pero normalmente se debe a anotaciones posteriores realizadas por las personas que manipulan el documento.

En este caso, la distinción entre los trazos de la firma y el texto manuscrito parece imposible desde un punto de vista puramente morfológico, resultando necesario utilizar un enfoque que tenga en cuenta la semántica asociada a los textos.

### Residuo en forma de ruido aleatorio

En muchas ocasiones se puede observar dentro de la imagen digitalizada cierto ruido en forma de manchas o puntos con distribuciones aparentemente aleatorias. En el caso de imágenes en niveles de gris o de color este ruido se debe a problemas en el documento original o en el dispositivo de digitalización. Sin embargo, en el caso de imágenes bitonales puede deberse también a efectos colaterales al proceso de umbralizado. En la Figura 2.16 se puede ver un ejemplo en el que una figura de fondo, al ser umbralizada, deja residuos en forma de ruido aleatorio. Esta vez la eliminación del ruido mediante técnicas morfológicas parece más sencilla porque la firma y el ruido no guardan parecido.

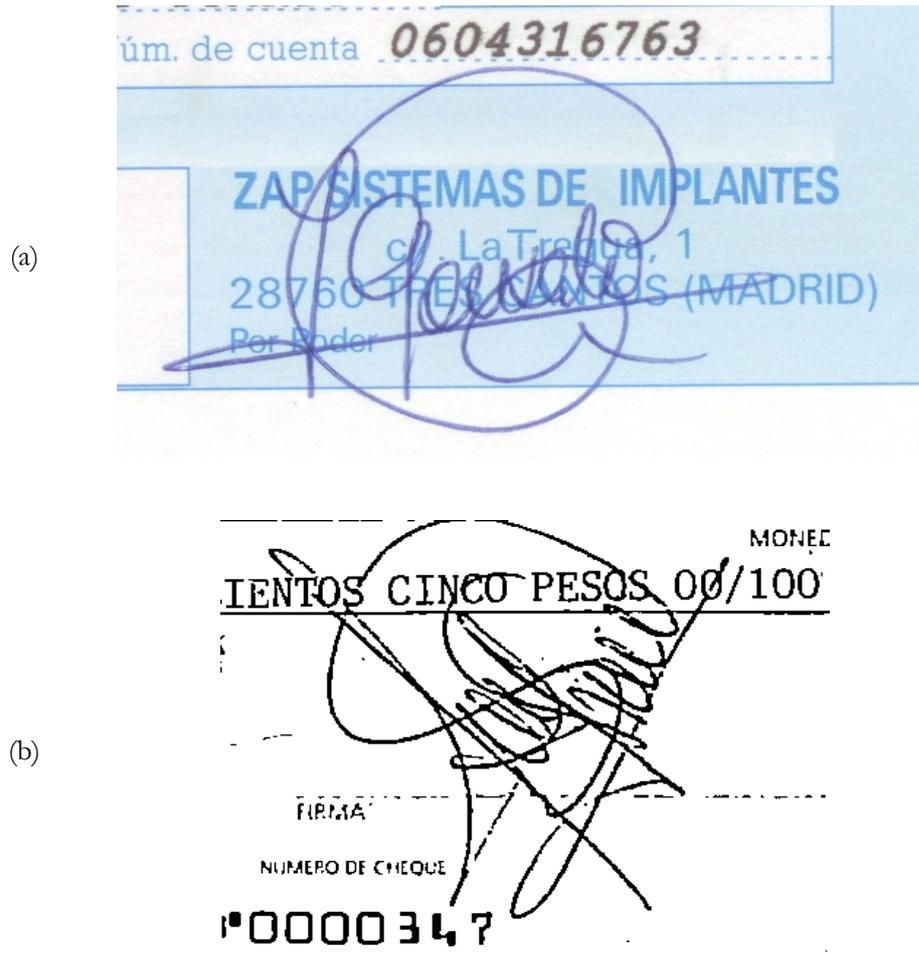


Figura 2.14.- Imagen de firmas sobre texto preimpreso y mecanografiado: (a) imagen en color y (b) imagen bitonal.



Figura 2.15.- Imagen en color de un conjunto de dígitos manuscritos adyacentes a una firma.



**Figura 2.16.-** Ruido como producto de la binarización. El fondo de la imagen en color (a) conserva sentido al convertirla a niveles de gris (b), pero lo pierde y parece ruido aleatorio al binarizarla (c).

### 2.3 Técnicas de verificación automática de firmas *off-line*

En esta sección se repasan los trabajos más relevantes que han aparecido en el campo de la segmentación y la verificación automática de firmas. Para ello se han realizado numerosas búsquedas en los principales foros especializados, entre los que podemos citar: el *International Symposium in Frontiers of Handwriting Recognition (IWFHR)*, la *International Conference on Pattern Recognition (ICPR)*, la *International Conference on Document Analysis and Recognition (ICDAR)*, así como las revistas: *International Journal of Pattern Recognition and Artificial Intelligence*, *Pattern Recognition Letters* y *Pattern Recognition*. El histograma de la Figura 2.17, que recoge los trabajos más importantes aparecidos en estos foros, permite comprobar la actualidad del problema en estudio.

El desarrollo de sistemas informáticos para la verificación *off-line* de firmas se inicia con los trabajos de Nagel y Rosenfeld en 1973 [45][46]. La gran mayoría de los trabajos se refieren a firmas occidentales, aunque también hay trabajos sobre firmas en el mundo árabe [2] y oriental [47][48][49]. Desde entonces, periódicamente, han ido apareciendo algunos trabajos recopilatorios sobre el estado del arte del tratamiento automático de firmas manuscritas, cuyos principales exponentes son: [1] [50] [51] [17] [52] [12] [53] [4].

### 2.3.1 Tipología de los trabajos sobre verificación *off-line* de firmas

En general los trabajos se enfocan desde una perspectiva grafonométrica [52], es decir se intenta realizar la verificación realizando diversas medidas sobre la imagen de la firma. Una diferencia con los trabajos anteriores a la aparición del ordenador estriba en el aumento de complejidad de las medidas que el ordenador permite realizar.

Los trabajos podemos englobarlos en dos categorías principales:

- Aquéllos que aportan alguna característica discriminante novedosa que puede utilizarse para el problema de la verificación. En estos trabajos se deja en segundo plano el tipo de clasificador utilizado. En los primeros trabajos las características discriminantes se asimilaron del enfoque clásico de la grafoscopia (ver subsección 2.1.3), pero con el tiempo han aparecido características novedosas generalmente dependientes de las posibilidades introducidas por el tratamiento informático de las imágenes y el incremento en capacidad de cómputo de los ordenadores.
- Aquéllos en los que el tipo de clasificador utilizado es la novedad. En estos casos suele justificarse que el clasificador es capaz de mejorar los resultados obtenidos por características discriminantes propuestas por otros autores, o bien que el clasificador en si mismo posibilita un análisis directo de la firma que no precisa de otras características.

Por supuesto, también encontramos trabajos mixtos en los que se combinan nuevas características con el uso de clasificadores novedosos.

#### Características discriminantes usadas para la verificación de firmas *off-line*

La extracción de características sobre la imagen de la firma es un enfoque clásico en la Visión Artificial [54]. En este caso, tras las etapas de captura, preproceso y segmentación se realiza una extracción de un conjunto de características discriminantes que se utilizan para clasificar el objeto (aquí para verificar la firma).

Diferentes autores suelen clasificar las características discriminantes utilizadas en el proceso de verificación de diferentes maneras. Inicialmente, en un trabajo de 1936, E. Locard [34] las clasificó en estáticas y pseudodinámicas, según utilizan información estática de la imagen o tratan de hallar la dinámica del proceso de firmado subyacente, esta clasificación es retomada por Weip y otros [11] en un trabajo del 2004. En 1991, S. Lee y J.C. Pan [8] hablan de tres tipos de características: las globales, que se basan en el estudio de cada píxel de la imagen por separado, las estadísticas, basadas en el estudio de las distribuciones de los píxeles de la firma, y las geométrico-topológicas, que intentan describir las formas interiores a una firma. Finalmente, en 2004, J. Fierrez y otros [4], clasifican las características discriminantes en dos grupos: características globales (*global* u *holistic*) que utilizan la imagen de la firma en conjunto y características locales (*local* o *grid*), que se basan en el estudio de zonas o partes específicas de la firma.

A continuación, se recogen las características globales más importantes que se repiten en la literatura para el caso de la verificación *off-line*. En general, se hace referencia a los principales trabajos en las que aparecen, aunque en algunos casos se hace referencia también a trabajos recopilatorios.

- Proporciones de la firma (*aspect ratio*) mediante la medida de la caja que contiene la firma (*bounding box*) [55][23][2].
- Centros de gravedad y otros momentos medidos sobre las proyecciones de la firma sobre los ejes horizontales y verticales [56][23][2].

1970-1979	Computer Screening Of Handwritten Signatures: A Heuristic Approach To Signature Verification [140]	Steps Towards Handwritten Signature Verification [140]	Experimental Investigation Of Automatic Signature Verification [100]	Computer Detection Of Forged Signatures [90]
1980-1989	A Heuristic Approach To Signature Verification [140]	Segment And Feature Extraction Of Handwritten Signatures [120]	Computer Methods Of Signature Verification [100]	A New Effective Approach For Automatic Offline Verification Of Signatures [102]
1990	Designing An Automatic Signature Verifier: From Description To Implementation [147]	Structural Description And Classification Of Signature Features [120]	Shape Histograms Detection Of Forged Signatures [111]	A Multiple Level Signature Verification System [91]
1991	Verification Of Fraudulent Handwritten Signatures: A Case Study [47]	Progress In Verification Of Handwritten Signatures [120]	Machine Verification Of Handwritten Signatures [111]	Signature Verification On Neural Networks [100]
1992	Offline Ident With Images: Survey And Perspective [144]	Offline Tracing And Recognition Of Signatures [127]	Elimination Of Skewed Handwritten Signatures: A Breakthrough [113]	Offline Signature Verification Using Neural Networks [100]
1993	An Extended Stroke-code Signature Verification System: Survey And Perspective [144]	An Artificial Neural Network Approach To Offline Signature Verification [129]	Stroke Signature Verification Using Neural Networks [113]	A NN Approach For Offline Signature Verification [101]
1994	Automatic Signature Verification For Off-line Signatures: The State Of The Art [130]	Algorithms For Signature Verification [129]	Special Issues Of Automatic Signature Verification [115]	Structural Interpretation Of Handwritten Signatures [102]
1995	Recognition Of Renewed And Modified Signatures: A Believable And Identifiable Approach [148]	Local Correspondence For Detecting Random Forgeries [129]	Shape Of Sign: Verif. Using A New Approach To Genetic Based Learning [142]	A Mathematical Approach To Computer Verification Of Handwritten Signatures [143]
1996	Algorithms For Automatic Signature Verification [171]	Offline Signature Verification Using An Extended Stroke-code [129]	Signature Verification Using Genetic Algorithms [102]	Offline Verification Of Handwritten Signatures: A Knowledge Of Class D, A New Approach [150]
1997	Check Background Based On A Systemic Criterion [131]	A Systematic Comparison Between On-line And Off-line Verification For Signature [130]	On-line Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
1998	Signature Classification By Hidden Markov Model [171]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
1999	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2000	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2001	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2002	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2003	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2004	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2005	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2006	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]
2007	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Extended Stroke-code [129]	Offline Signature Verification Using An Auto-associator [118]	Offline Signature Verification Using Genetic Algorithms [102]

Figura 2.17.- Histograma de referencias de los artículos aparecidos en los principales foros especializados que tratan específicamente el procesamiento *off-line* de firmas. El fondo celeste indica que el trabajo trata la verificación, el fondo gris indica que el trabajo es recopilatorio, el fondo naranja indica que el trabajo trata la segmentación, y el fondo verde que trata la verificación.

- Línea base global (*global base line*) [161][2] y los límites superiores e inferiores de la firma [2].
- Número de bucles, puntos de cruce y puntos extremos encontrados en la firma [71][106].
- Varias medidas sobre diferentes tipos de envolventes de la firma (*envelope*) [56][23].
- Estimaciones del ángulo de la línea base de la firma (*slope*) mediante los ejes de inercia [4][87].
- Estimación del ángulo de inclinación de los trazos presentes en la firma (*slant*) [55][161][2].
- Área de los píxeles activos [161][94][23], a menudo normalizados respecto al área de la caja que contiene la firma y utilizando el esqueleto para ser invariantes al grosor del elemento de escritura.
- El número de componentes de la firma [55][94] mediante el análisis de las componentes conexas de la imagen.
- Varias características basadas en cálculos de *wavelets* sobre la imagen de la firma [23][95].
- Una última característica global que se utiliza frecuentemente es la propia imagen de la firma, o una imagen escalada de ella a una resolución fija menor [106] [162] [140].

Por otro lado, entre las características locales que encontramos en la literatura abundan variantes de las globales, aplicadas sobre regiones acotadas de la firma, como ventanas (verticales u horizontales) o celdas (*grid*) [161]. También se encuentran características locales que no derivan de globales, entre ellas podemos destacar:

- Densidades por celdas o regiones de la imagen [161][151].
- Medidas locales de orientación de trazos [163][107].
- Ángulo predominante de situación de píxeles por ventanas [163].
- Análisis local de una retícula situada sobre la firma mediante unos artefactos matemáticos llamado *Distribuciones de Tamaño Granulométrico* (*Granulometric Size Distributions* o *GSD*) [83] que basados en morfología matemática permiten obtener una descripción vectorial de cada punto de la retícula.
- Características pseudodinámicas como la presión obtenida por equivalencia con la intensidad de la luminosidad del trazo en imágenes en niveles de gris [152], o una reconstrucción de la dinámica del trazado [107].

### **Clasificadores utilizados en el problema de la verificación automática de firmas *off-line***

Es un hecho que, al poco tiempo de la aparición de un nuevo tipo de clasificador, algún grupo de investigación lo prueba para el problema de verificación automática de firmas. Así se encuentran trabajos que utilizan las Redes Neuronales (*Neural Networks* o *NN*), las Máquinas de Vectores de Soporte (*Support Vector Machines* o *SVM*), los Métodos de Ajuste Elástico, los Modelos Ocultos de Markov (*Hidden Markov Models* o *HMM*), o los Clasificadores Borrosos, entre otros. En la Tabla 2.3 se muestran algunos trabajos que utilizan estos clasificadores.

Tipo de clasificador	Algunos trabajos que lo usan
Redes Neuronales	[128][56][106]
Máquina de Vectores de Soporte	[48][164]
Ajuste elástico	[21][20][141]
Modelos Ocultos de Markov	[40][122][164]
Clasificadores borrosos	[155]
Redes Bayesianas	[78]

**Tabla 2.3.-** Algunos tipos de clasificadores utilizados para la verificación de firmas y algunos trabajos que los usan.

### 2.3.2 Trabajos relevantes en el problema de verificación automática de firmas *off-line*

La Figura 2.18 presenta un grafo con los autores de trabajos más relevantes sobre verificación de firmas *off-line*. Obsérvese que se mantiene un orden, de manera que los autores que antes abordan el problema están arriba, mientras que los más modernos están abajo. La figura también recoge las relaciones entre autores que han dado lugar a publicaciones. Así, las cajas contienen a los autores o a grupos de autores, mientras que las publicaciones aparecen mediante una fecha en la línea o en un círculo que une a los coautores que la crearon. Como se puede apreciar, aparecen grupos más o menos estables de autores que colaboran en sucesivos artículos. Entre los más prolíficos podemos destacar a R. Plamondon, a R. Sabourin y a B. Fang.

Aunque han sido muchos los artículos revisados, en los siguientes puntos se describen brevemente aquéllos que, por el método que describen o por los problemas que abordan, han tenido más influencia en el desarrollo de los trabajos que se presentan en esta Tesis:

- M. Ammar y otros [55] proponen, en un trabajo de 1990, un análisis de la firma basado en características globales y en un análisis local. Este último análisis se concreta en una representación en árbol que describe la firma mediante un conjunto ordenado de elementos (*Global Descriptor String* o *GDS*). Estos elementos se obtienen al realizar diferentes medidas locales según se recorre la firma de izquierda a derecha, siguiendo una línea horizontal imaginaria. Desafortunadamente, en su trabajo no realizan pruebas experimentales de verificación.
- J.C. Pan y S. Lee [8], en 1991, proponen una representación de la firma basada en una serie de elementos que simulan el proceso humano de generación de trazos (*strokes*) para la verificación de firmas. En el trabajo se exponen 7 reglas heurísticas que se siguen a la hora de construir el trazado de la firma:
  - Regla 1.- Se debe comenzar el trazado por puntos extremos.
  - Regla 2.- El trazado sigue los píxeles utilizando el criterio de vecindad.

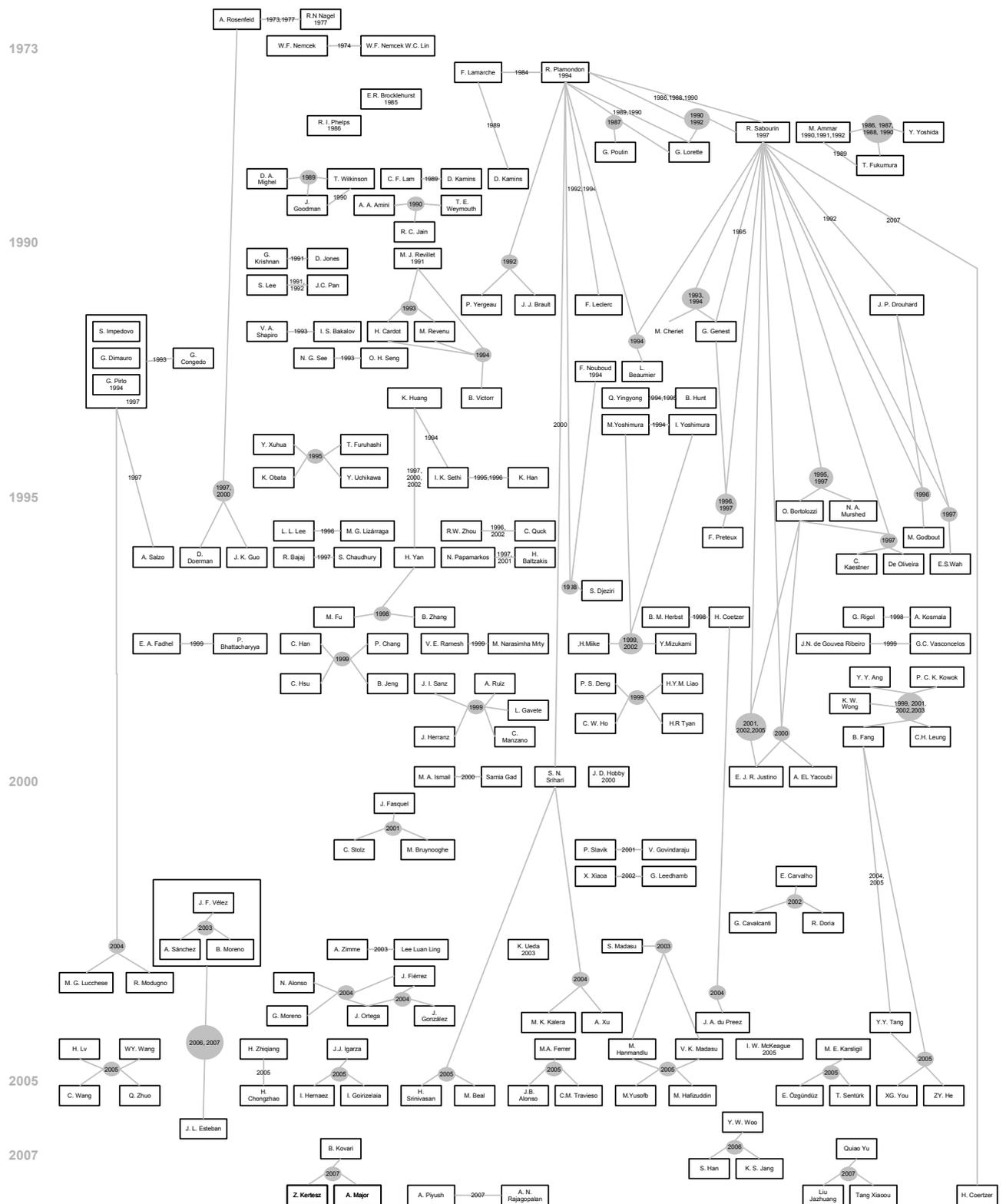


Figura 2.18.-Principales autores en verificación automática de firmas *off-line*. Las fechas datan las colaboraciones realizadas entre los autores unidos por las líneas. Los trabajos están ordenados de más antiguos (arriba) a más nuevos (abajo).

- Regla 3.- En las intersecciones se toma el camino que maximice la suavidad.
- Regla 4.- Algoritmo para detectar y catalogar los finales de trazo.
- Regla 5.- Algoritmo para realizar el retroceso necesario al trazar caracteres como la 't'.
- Regla 6.- Regla de ordenación del orden de los píxeles de cada trazo.
- Regla 7.- Regla de ordenación del orden en que se realizan los diferentes trazos.

En este trabajo tampoco se ofrecen resultados experimentales.

- Yingyoung Qi [161], en 1994, compara un conjunto de características geométricas con el estudio de las características obtenidas al superponer una rejilla (*grid*) a la firma y realizar el análisis los bordes de cada celda de la rejilla para obtener un código binario descriptor. Debido a que el uso directo de la distancia euclídea entre los patrones así obtenidos no da resultados satisfactorios, se utiliza un proceso de ajuste previo entre los patrones de características basado en técnicas de programación dinámica. Los porcentajes de error que postula son muy bajos, pero en los experimentos no se aprecia una clara separación entre las etapas de aprendizaje y de test.
- En 1997, R. Bajaj y S. Chaudhury [56] construyen un sistema basado en dos tipos de características discriminantes: los momentos y las envolventes superior e inferior. Luego utiliza varias Redes de Neuronas de tipo *feed-forward*. Usando una base de datos compuesta de 10 individuos, con 15 firmas por sujeto, y usando 5 muestras para el aprendizaje de cada individuo, obtiene un *FRR* del 1% y un *FAR* del 3% para falsificaciones aleatorias.
- En 1998, G. Rigoll y A. Kosmala [151], y después E. Justino, Bortolozzi y R. Sabourin [40] [131][163] han estudiado el problema utilizando: una segmentación de la firma en celdas, la obtención de una cadena de símbolos y el posterior uso de *HMM* como clasificador. Las características que se utilizan para cada celda en estos trabajos son: la densidad de los píxeles negros [151], el *ESC* (*Extended Shadow Code*) de R. Sabourin [135] y el ángulo predominante [163].
- B. Fang y su equipo [76] abordan, en 1999, el problema de los falsificadores habilidosos utilizando una aproximación basada en conocimiento de expertos humanos según la cual las falsificaciones son menos suaves y naturales que las firmas genuinas, construyendo un índice de suavidad para su estudio. En este trabajo se justifica no poder ofrecer una distinción entre *FRR* y *FAR* debido al reducido tamaño de la muestra y sólo se ofrecen unos resultados del 17.4% de error medio en la verificación.
- También en 1999, V. E. Ramesh y otros [23] construyen un sistema que funciona con firmas escaneadas a 72 DPIs. El enfoque se basa en características discriminantes globales, en características de celda y en características obtenidas mediante un análisis de *wavelets*. En este trabajo, los autores ensayan diferentes tipos de clasificadores. Utilizando el mejor de ellos, 15 firmas de cada individuo para entrenar y falsificaciones de las firmas para entrenar los rechazos, obtienen un 10% de *FRR*, un 2% de *FAR* para las falsificaciones simples y un 30% para las falsificaciones habilidosas.
- En 2002, Y. Mizukami [22] que aborda la problemática de utilizar pocos ejemplares, utilizando uno para el aprendizaje y otro para determinar un umbral de rechazo. Su enfoque se basa en la comparación de imágenes utilizando funciones de desplazamiento. Los resultados que describe el autor, en torno al 24% de error medio, son extremadamente buenos teniendo en cuenta que trata la problemática de los falsificadores habilidosos. Desgraciadamente sus pruebas se realizan sobre firmas japonesas, que al ser tan diferentes a

las firmas occidentales hace que sus resultados no se puedan comparar con los que se presentan en esta Tesis.

- En 2002, K. Huang y H. Yan [107] estudian la firma descomponiéndola en los trazos que aparentemente la componen (mediante fronteras direccionales) y ordenando los trazos en una posible secuencia temporal. Posteriormente, para verificar una firma, estudian la correspondencia entre los modelos de firma que han obtenido mediante un procedimiento que denominan “ajuste por relajación” (*relaxation matching*).
- En 2002, X. Xiao y G. Leedhamb [78] proponen el uso de una Red Bayesiana para tratar el problema de la no certeza o la desaparición de las características discriminantes que ocurre al comparar las envolventes de la firma como método de verificación. Para la etapa experimental crea una muestra de 8 individuos con entre 10 y 20 firmas por cada uno. Utilizando el 60% de la muestra para el aprendizaje se obtienen unos resultados para el *FRR* del 20% y para el *FAR* (frente a falsificaciones aleatorias) del 14%.
- También en 2002 B. Fang y otros [20] abordan la problemática de la dificultad de obtener múltiples firmas por cada individuo. En este caso, intentan utilizar técnicas de ajuste elástico (*elastic-matching*) para generar nueva muestra modificando la existente. Además, utilizan ventanas verticales y horizontales para obtener ciertas medidas en las transiciones de píxeles blancos a negros que usan como características discriminantes. Este trabajo encuentra precedente en otro de Oliveira y otros [117] aunque en aquél se centran en la generación de la muestra y no se realiza ningún experimento de verificación. Utilizando 23 firmas de cada individuo para generar 529 muestras de aprendizaje por individuo se consiguen unos resultados del 14% de error medio (de nuevo, justificando que no se puede obtener una distinción entre *FRR* y *FAR*).
- Nuevamente, en 2003, Fang y otros [138] abordan el problema de los falsificadores habilidosos desde dos perspectivas diferentes. Por un lado, el estudio de la posición relativa de los trazos principales de la firma y por otro los cambios observados en las proyecciones de la firma sobre los ejes ordenados. Utilizando de nuevo 23 firmas por individuo durante el aprendizaje, esta vez se ofrece un resultado final del 23% de *EER* para el mejor de los dos enfoques propuestos.
- En 2005 X. G. You y B. Fang [21] utilizan la distancia entre los puntos localizados en dos modelos elásticos realizados sobre las firmas a comparar, atacando el problema de la imposibilidad de utilizar varias firmas y el de los falsificadores habilidosos a la vez. En este trabajo utilizan sólo 4 muestras por individuo durante la fase de aprendizaje, una para el modelo y 3 para determinar los umbrales de rechazo. Sus resultados fueron de un 18.6% de *EER*.
- En 2005, Justino y otros [164] comparan, en un trabajo que destaca por impecable proceso de recolección de muestra y experimentación, el uso de SVM y HMM para diferentes características extraídas sobre celdas. Es importante señalar que éste es uno de los pocos trabajos en los que se señala que utilizan una muestra para diseñar el sistema y otra diferente para realizar el test. Además, se realizan diferentes pruebas con un número creciente de ejemplares para el aprendizaje. El número más bajo de muestras de aprendizaje que utilizan es 5, y en este caso obtienen un *FRR* del 25% para los *HMM* y del 30% para los *SVM*. Para este mismo caso, el *FAR* que obtiene lo cataloga entre los diferentes tipos de falsificadores. Así, para los aleatorios obtiene un 27% con el *HMM* y un 0% con el *SVM*. Para los falsificadores simples obtiene un 38% con el *HMM* y otro 0% con el *SVM*. Finalmente, para los habilidosos obtiene un 34% para el *HMM* y un 2% para el *SVM*.

- En un trabajo de 2007, A. Piyush y A. N. Rajagopalan [141] verifican firmas mediante la comparación de las proyecciones verticales utilizando una variación de la técnica *Dynamic Time Warping (DTW)* que se basa en programación dinámica, por lo que también se le conoce como *Dynamic Programming Matching*. Esta técnica se usa ampliamente para alinear secuencias de manera óptima, en el sentido de minimizar la distancia entre dos secuencias de características discriminantes de diferente longitud. Durante la fase de aprendizaje utilizan 10 firmas de cada individuo para obtener secuencias con las que luego comparan las firmas a verificar. Los resultados que reportan son de un *FRR* del 2% y un *FAR* del 0% para falsificadores aleatorios cuando sólo trata de distinguir entre ambos. Cuando se consideran las falsificaciones habilidosas reportan un *FRR* del 25%, y un *FAR* del 0% para los aleatorios y de un 20% para los habilidosos.
- En otro trabajo de 2007, Yu Quia y otros [133] proponen un enfoque basado en la combinación de modelos *on-line* para el aprendizaje y de verificación *off-line*. En su trabajo se utiliza la técnica de *Conditional Random Fields (CRT)* para encontrar la correspondencia entre la firma *on-line* y la imagen *on-line*. Tras esto se obtiene una trayectoria que una vez alineada utilizando *DTW* se compara con las trayectorias *on-line* almacenadas para ese individuo. En sus experimentos utilizan la base de datos de la competición de verificación de firmas del año 2004 (*SVC 2004*), la cual consta de firmas inglesas y chinas. Emplean 10 firmas *on-line* de cada individuo para el aprendizaje, otras 10 *on-line* se convierten a *off-line* y se utilizan para el test y también se hace lo mismo con 20 falsificaciones simuladas. Sus resultados arrojan un 7.3% de ERR, aunque no detallan qué porcentaje corresponde a firmas chinas y cuál a firmas inglesas.

La tabla 2.4 resume y compara los resultados de estos trabajos respecto a las falsificaciones aleatorias, y la tabla 2.5 lo hace respecto a las falsificaciones hábiles.

Se puede resumir que el *FAR* obtenido para las falsificaciones aleatorias ronda el 0% y el *FRR* varía entre el 0% y el 25%. Sin embargo, debe tenerse en cuenta que no se ha encontrado ningún trabajo en el cual sólo se utilice una única muestra durante la fase de aprendizaje. Además, cuando el número de ejemplares utilizados en el aprendizaje se reduce, el error crece de forma importante.

En cuanto al *FAR* frente a falsificadores habilidosos debe señalarse que normalmente es muy alto (superior al 30%). En aquellos casos en que este error es bajo se debe a que se entrenan los sistemas con falsificaciones de la misma mano que luego, en la fase de test, intenta engañar al sistema, cosa que en un sistema real no sería posible.

Autor y trabajo	Año	FRR	FAR	Ejemplares utilizados en el aprendizaje	Descripción general
Bajaj y Chaudhuri [56]	1997	1%	3%	5	Características discriminantes
Ramesh y otros [23]	1999	10%	2%	15	Características discriminantes globales + <i>grid</i> + <i>wavelets</i> + 72 DPIs
Xiao y Leedhamb [78]	2002	20%	14%	6-12	Redes Bayesianas
Justino y otros [164]	2005	25%	27%	5	HMM
Justino y otros [164]	2005	30%	0%	5	SVM
Piyush y Rajagopalan [141]	2007	25%	0%	10	<i>Dynamic Time Warping</i>

**Tabla 2.4.-** Resumen de los trabajos que tratan el problema de las falsificaciones aleatorias.

Autor	Año	FRR	FAR	Ejemplares utilizados en el aprendizaje	Descripción general
Ramesh y otros [23]	1999	10%	30%	15	Características discriminantes globales + <i>grid</i> + <i>wavelets</i> + 72 <i>DPIs</i>
Fang y otros [138]	2003	23%	23%	23	Posiciones relativas de los trazos
You y Fang [21]	2005	18.6%	18.6%	4	Modelos elásticos
Justino y otros [164]	2005	25%	34%	5	HMM
Justino y otros [164]	2005	30%	2%	5	SVM
Piyush y Rajagopalan [141]	2007	25%	20%	10	<i>Dynamic Time Warping</i>

**Tabla 2.5.-** Resumen de los trabajos que tratan el problema de las falsificaciones habilidosas.

Por otro lado, respecto a la metodología observada en estos trabajos, se debe señalar que pocos separan la muestra que se utiliza para diseñar el sistema de la muestra que se utiliza para probarlo. Esto puede restar validez a los resultados ya que los algoritmos que se diseñan podrían estar adaptados a las particularidades de la muestra de test que se utiliza.

Finalmente, respecto a los conjuntos de muestra utilizados se debe señalar la completa falta de homogeneidad. Cada autor utiliza una muestra propia que luego no hace pública. Además, la muestra que utilizan casi siempre se concentra en tomas de una sola sesión para cada individuo. En ninguno de los casos se realizan capturas de muestra a un mismo individuo a lo largo de diferentes años.

## 2.4 Técnicas de identificación automática de firmas *off-line*

El número de trabajos existentes en el área de la identificación de firmas es mucho menor que el encontrado en el área de verificación [16]. Esto seguramente se debe a su menor aplicabilidad respecto al problema de la verificación.

Al igual que en el caso de la verificación de firmas, en la identificación de firmas encontramos la dicotomía entre trabajos en los que la novedad está en la introducción de alguna característica discriminante y aquéllos en los que el tipo de clasificador utilizado es la novedad.

Hay autores que utilizan las mismas características discriminantes en el problema de la identificación de firmas y en el de la verificación [87]. Aunque también hay autores que encuentran diferencias entre unas y otras [2], debido principalmente a la mayor especialización que requiere la discriminación de falsificadores habilidosos que no se considera en el problema de la identificación.

### 2.4.1 Trabajos relevantes en el problema de la identificación de firmas

En la Figura 2.17 se presentan los autores más relevantes así como la relación que hay entre ellos que ha dado lugar a una publicación. Como se puede observar en la Figura 2.17 el número de trabajos es muy inferior al que se observa para el problema de la verificación de firmas. A continuación comentamos brevemente varios artículos cuyas han tenido alguna relevancia en el desarrollo de esta Tesis:

- Pavlidis y otros [16] utilizan el contorno de la firma (una especie de cierre convexo) que obtienen mediante la técnica de *snakes* para comparar las firmas a clasificar.

- Entre los trabajos en los que el clasificador es la novedad encontramos a J. L. Camino y otros, que utilizan *HMM* combinados con el código de cadena de la firma [137]. En su enfoque Camino y su equipo tratan de encontrar una representación pseudodinámica a partir de la imagen *off-line*.
- M. K. Kalera y otros [87] relacionan el problema de la identificación de firmas con el del reconocimiento de autor basado en texto manuscrito *off-line*, y por ello utilizan características comunes a este problema.

## 2.5 Técnicas de segmentación automática de firmas off-line

Como se ha señalado anteriormente, el problema de la segmentación de la firma se simplifica utilizando imágenes en color o incluso en niveles de gris. Quizás por ello, el problema de la segmentación de firmas *off-line* ha sido constantemente subestimado. Así, aunque encontramos trabajos que se remontan a la década de 1980 [125][18] en general no se ha tratado en profundidad el aspecto de la segmentación y los autores suelen suponer que la firma a verificar está limpia de ruido y perfectamente localizada [12]. Como consecuencia de lo anterior, el número de trabajos sobre segmentación de firmas es inferior al número de ellos sobre identificación y, por supuesto, que a los de verificación (véase la distribución de colores de la Figura 2.17).

Sin embargo, la realidad del problema de la segmentación es más compleja y, en general, no está resuelta totalmente. Para empezar ya se ha comentado que los procesos industriales de captura de documentos exigen el uso de imágenes bitonales, por lo que cualquier solución basada en el uso del color o de niveles de gris queda invalidada. Además, es difícil justificar su uso cuando una persona no necesita de estos requisitos, como puede comprobarse en las pruebas realizadas con personas por Ramesh y Fang y sus equipos respectivos [23][20]. Por otro lado, la problemática del ruido con estructura tampoco ha sido abordada más que en fechas recientes. En particular, Larrea y su equipo [165], utilizando muestra suministrada por el autor de esta Tesis, han realizado un estudio sobre la segmentación de firmas sobre cheques de los que debe disponerse de un modelo previo y han construido un sistema supervisado que, a partir de un documento bancario, permite limpiar y segmentar una firma en unos pocos segundos a un operador previamente instruido.

### 2.5.1 Trabajos relevantes en el problema de la segmentación de firmas

Entre los pocos trabajos que tratan el aspecto de la segmentación de la firma en un documento digital cabe destacar:

- Un trabajo de L. Lee y otros [19], de 1997, que estudia la segmentación de firmas y otros elementos existentes en cheques bancarios reales. El tipo de cheque que procesa el sistema es único, por lo que utiliza un procedimiento consistente en la sustracción de un cheque modelo. Dicho método proporciona unos resultados cercanos al 100%.
- Un trabajo de S. Djeziri y otros [136], de 1998, donde se utiliza un etiquetado de los elementos que parecen trazos mediante morfología matemática, simulando la capacidad del sistema visual humano para detectar líneas y que se prueba también sobre documentos bancarios en niveles de gris. El trabajo ofrece unos resultados cualitativamente etiquetados como buenos o muy buenos en el 80% de los casos y del 20% etiquetados como malos o muy malos.
- Un trabajo de Q. Ye y otros [166], del 2000, basado en las características de las formas y la estructura (*layout*) del documento, para extraer el texto manuscrito que aparece sobre documentos de tipo formulario. Los autores realizan una prueba subjetiva según la cual el

97% de los caracteres manuscritos se recuperan correctamente. También realizan una prueba objetiva según la cual un sistema de OCR posterior a la segmentación es capaz de reconocer correctamente el 95% de los caracteres.

- También en el año 2000, Hobby [14] propone el cálculo de características discriminantes sobre cada segmento de un texto de carácter general. Los segmentos se obtienen mediante el esqueletizado y un proceso posterior de ajuste de segmentos curvos. Luego, con la ayuda de un clasificador, separa el *layout* del documento, de las firmas, del texto y de los gráficos. En sus experimentos utiliza imágenes bitonales a 200 DPIs, un clasificador del tipo Vecino Más Próximo (*Nearest Neighbourb*) y obtiene un 19% de *FRR* y un 14.9% de *FAR*.
- Ueda y otros [167] proponen, en 2004, un método de segmentación de firmas japonesas basada en el análisis de los valores de matiz, saturación y brillo (Hue Saturation Value HSV) de los píxeles del cheque en color. Las pruebas realizadas sobre 19 cheques ofrecen resultados visualmente satisfactorios en todos los casos.
- En 2005, Madasu y Lovel [168] proponen el uso de una ventana deslizante que mide la entropía en torno a cada píxel para decidir si pertenece o no a la firma. Los resultados que ofrecen son del 97.78% de segmentaciones correctas, aunque los cheques que se muestra en el trabajo parecen libres de ruido y están en niveles de gris.
- Un trabajo de Larrea y otros [165], en 2006, permite asistir a una persona facilitándole la segmentación de firmas sobre documentos bancarios. Este trabajo ha utilizado muestra suministrada por el autor de la Tesis.

Puede observarse que pocos de los trabajos tratan el problema de la segmentación en las condiciones que se plantean en este trabajo, esto es a 200 *DPIs* y sobre imágenes bitonales. Sin embargo, una persona es capaz de realizar esta tarea sobre este tipo de imágenes sin problemas, como se desprende del trabajo de Larrea y otros [165].

## 2.6 Resumen del capítulo

En este capítulo se ha realizado un análisis de los problemas relacionados con la verificación automática de firmas *off-line*. Para ello, primeramente se ha presentado la nomenclatura y los enfoques utilizados por el peritaje caligráfico clásico. A continuación, se han detallado los principales problemas que plantea un sistema realista de verificación automática de firmas. Finalmente, se ha presentado el estado del arte respecto al problema de la verificación automática de firmas *off-line* en su conjunto.



## Capítulo 3 Base de datos de firmas: captura y segmentación

Un sistema de Visión Artificial opera distinguiendo entre imágenes que pertenecen a un amplio conjunto que se denomina *universo de trabajo*. La construcción de tales sistemas suele precisar de varias imágenes que a modo de modelos permitan calcular las funciones discriminantes de clasificación. Estas imágenes modelo forman el *conjunto de aprendizaje*, el cual, para ser útil, debe estar constituido por un subconjunto representativo del universo de trabajo.

Cuando la muestra es abundante suele reservarse una parte del conjunto de aprendizaje para probar la bondad de las funciones discriminantes que se calculen, que se denomina *conjunto de test*. Es importante que el conjunto de aprendizaje y el de test sean independientes, ya que esto permite cierta confianza empírica en que el clasificador desarrollado posee la deseable propiedad de generalización. Esta propiedad trata de garantizar que un sistema clasifica correctamente patrones que no se han utilizado durante el proceso de cálculo de funciones discriminantes. Como norma general, en el caso de un universo de trabajo grande, la independencia queda asegurada si el conjunto de aprendizaje y el test no tienen elementos en común.

Si una vez construidos ambos conjuntos, los clasificadores se prueban usando el conjunto de test y se obtienen unos resultados deficientes, deben descartarse ambos conjuntos y se debe volver a comenzar con un nuevo clasificador y nuevos conjuntos. Dicho de otro modo, si para mejorar los resultados de test, se volviesen a diseñar nuevas funciones discriminantes, se estaría usando el conjunto de test de manera indirecta en el proceso de aprendizaje, con lo que las pruebas posteriores no serían válidas. Para evitar esto, cuando hay suficiente muestra, suele crearse un tercer conjunto, independiente de los otros dos, llamado conjunto de validación. Este conjunto se utiliza para probar el sistema mientras se está construyendo. Éste tercer conjunto se utiliza durante el proceso de diseño de las funciones discriminantes para probar el sistema que se está construyendo con objeto de impedir que el conjunto de test se use más de una vez.

Se puede concluir que para posibilitar la realización de los experimentos que se plantean en esta Tesis se precisa una base de datos de imágenes de firmas. En el presente capítulo se justifica la creación de nuestra propia base de datos y se describe el proceso de diseño y captura de las imágenes de firmas que se ha realizado.

### 3.1 Base de datos de firmas

En el año 1993 el NIST<sup>8</sup> puso a disposición de la comunidad científica una gran base de datos de imágenes correspondientes a texto manuscrito. Desde entonces dicha base de datos ha sido utilizada por grupos de investigación de todo el mundo para realizar experimentos. Quizás, la principal consecuencia de esta aportación ha consistido en posibilitar la comparación directa entre los resultados de las diferentes investigaciones.

---

8 Instituto Nacional de Estándares y Tecnología (*National Institute of Standards and Technology*). Fundado en 1901, el NIST es una Agencia Federal del Departamento de Comercio de los EEUU. Su misión consiste en promover la innovación y la competitividad industrial mediante la creación de estándares, sistemas de medida y tecnología.

Desafortunadamente, la muestra del NIST no incluía imágenes de firmas manuscritas. Aunque existen diferentes bases de datos de imágenes de firmas manuscritas que podrían ser adoptadas por la comunidad científica, ninguna reúne las características que en general los diferentes grupos de investigación necesitan. Es por ello por lo que aún no existe un equivalente a la muestra del NIST para la problemática del reconocimiento de firmas manuscritas *off-line* [6].

En particular, esta Tesis precisaba dos características que no hemos encontrado en otras bases de datos. Por un lado, se requerían imágenes de firmas sin segmentar, acompañadas del ruido que habitualmente las rodea, para abordar la problemática de las firmas en su totalidad y no sólo en una irreal etapa final que parte de firmas “limpias”. Por otro lado, era un requisito disponer de varias imágenes de la firma de un individuo con diferentes útiles de escritura, simulando una realidad en la que no siempre se utiliza el mismo instrumento. La ausencia de estos requisitos motivó la creación de una base de datos propia.

Posteriormente, los requisitos de la base de datos de firmas se han ido ampliando. Finalmente, con objeto de que esta base de datos satisficiera las expectativas de un rango amplio de los grupos de investigación en este campo, hemos propuesto cumplir los siguientes requisitos:

- Variabilidad respecto al útil de escritura.
- Firmas sobre fondos pictóricos que permitan estudiar el problema de la segmentación bajo condiciones reales.
- Variabilidad en el tipo de sujeto firmante, incluyendo diferentes niveles sociales.
- Variabilidad temporal, con diferencia de meses o años, entre las diferentes firmas capturadas a cada individuo.
- Libertad total de acceso y uso por parte de la comunidad científica.

Para construir la base de datos se han utilizado dos fuentes:

- Por un lado se ha capturado un conjunto de firmas a partir de unos formularios que han sido rellenados por personas que voluntariamente han cedido su firma.
- Por otro lado se ha tenido acceso a un conjunto de documentos reales de tipo cheque y recibo, cortesía de la empresa española Investigación y Programas S.A.

En las siguientes secciones se describen los detalles de cada uno de estos conjuntos de imágenes.

## 3.2 Obtención de firmas mediante formularios

La obtención de las firmas se ha prolongado durante un periodo de 6 años, realizando una primera captura de muestra en el año 2002, una segunda en el año 2004 y una tercera en el 2007. Ello ha permitido obtener diferentes muestras de la firma de algunos individuos con varios años de diferencia. Además, a lo largo de estos años, el aumento de la comprensión del problema ha motivado que se varíe significativamente el material y el proceso utilizado para la captura de las firmas. En los siguientes puntos se detallan cada una de las fases de este proceso de recogida de firmas.

### **Diseño del primer formulario de captura de firmas**

En el año 2002 se realizó una primera captura de firmas. Se utilizó un formulario de tamaño A4, como el de la Figura 3.1, para la recogida de firmas. En esta primera ocasión se solicitaron 4 ejemplares de su firma a 29 personas.

Cada hoja constaba de una tabla de color amarillo (*RGB* 230,255,0 o *HSV* 58,100,100). El tono exacto de amarillo se eligió con objeto de que fuese fácil de eliminar tanto si la imagen se escaneaba en color como si se escaneaba en niveles de gris. Esta tabla, compuesta de 2 columnas y 8 filas, dio lugar a 16 rectángulos con unas dimensiones interiores de 8.8 cms de ancho y 4 cms de alto cada uno. Cada rectángulo se utilizó para la recogida de una firma.

Analizando la muestra recogida con estos formularios se observa que el ancho medio de sus firmas es de 4.33 cms, mientras que su alto es de 1.98 cms. Esto es, las firmas en general son más anchas que altas, algo asumido por el propio formulario y que se puede comprobar revisando la mayoría de los espacios reservados para firmar en otros documentos: cheques, declaraciones de renta, contratos... Todos tienen una relación de aspecto (ancho/alto) de aproximadamente 3 a 1.

### **Diseño del segundo formulario de captura de firmas**

En el año 2004 se realizó una segunda captura de firmas. En esta segunda ocasión se utilizó el mismo tipo de formulario de recogida de firmas. Esta vez se solicitan 6 firmas a 25 individuos diferentes a los de la primera encuesta. Además, se consiguió localizar a 10 de los 29 individuos de la primera encuesta para solicitarles 2 firmas adicionales. De esta forma se consigue completar a 6 el número de firmas de algunos individuos de la primera captura, obteniendo además para estos casos ejemplares con 2 años de diferencia (ver Tabla 3.1 para más detalles).

Debido a que el número de firmas solicitadas a cada individuo (6) no es divisor exacto del número de firmas que contiene cada formulario de recogida (16) en ocasiones se han dejado casillas vacías dentro de un formulario, siempre evitando que un firmante utilizase más de una hoja.

### **Diseño del tercer formulario de captura de firmas**

En el año 2007 se realiza una tercera recogida de firmas. Para esta tercera ocasión se creó un nuevo formato de formulario (ver Figura 3.2). Esta vez se utilizó una hoja independiente para cada firmante, que permitió recoger 9 firmas de cada individuo. Estas 9 firmas se dividen en diferentes categorías: 6 en un formato igual al de la primera y segunda recolecciones, 2 dentro de un pequeño recibo y 1 bajo los términos de aceptación de las condiciones de uso de sus firmas.

En este proceso de captura se han recogido firmas de 38 individuos. De nuevo se intentó localizar a los firmantes que participaron en los procesos de captura anteriores. En esta caso se localizaron 10 del primer proceso y 7 del segundo (ver Tabla 3.1 para más detalles). Los 21 firmantes restantes son nuevos.

El objetivo del cambio de formato fue el de obtener una muestra más diversa en cuanto a condiciones del documento sobre el que se firma. Así, en este caso, algunas de las firmas se encuentran rodeadas de texto, líneas y figuras de fondo. De esta forma, parte de las firmas pueden ser utilizadas como muestra útil para realizar experimentos de segmentación de firmas.

El análisis de la muestra correspondiente a los tickets es interesante porque limita en gran medida el área de firmado. El primer ticket deja un área para realizar la firma de 7.58x2.43 cms, mientras que el segundo ticket deja un área menor de 6.31x1.6 cms. Esta diferencia influye en el firmante de tal manera que realiza la firma con un área mayor en el primer ticket que en el segundo.

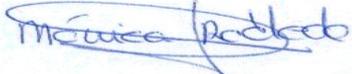
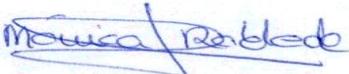
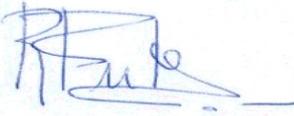
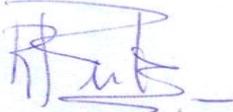
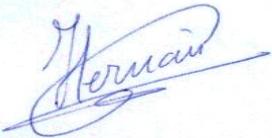
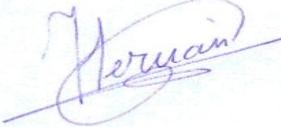
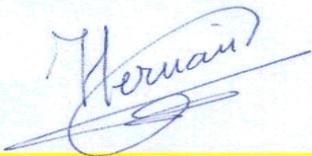
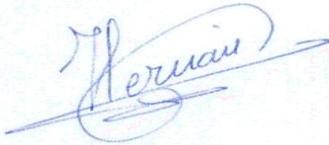
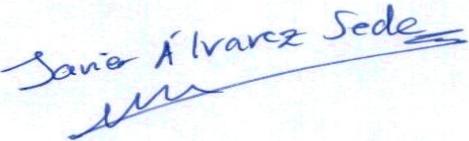
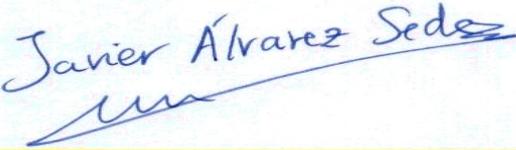
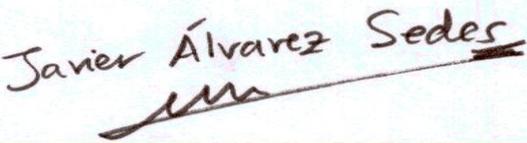
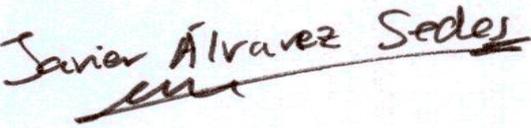
	
	
	
	
	
	
	
	

Figura 3.1.- Ejemplo de la primera hoja de muestras, utilizada en las recolecciones del 2002 y del 2004.



Figura 3.2.- Ejemplo de la segunda hoja de muestras, utilizada en las capturas de firmas del 2007.

Sin embargo, se puede observar que los anchos de las firmas correspondientes al segundo ticket son mayores que los anchos de las firmas del primer ticket. Así, las imágenes recortadas de las firmas del primer ticket en media miden  $3.47 \times 1.95 \text{ cms}^2$ , mientras que las del segundo ticket miden  $3.64 \times 1.8 \text{ cms}^2$ . Deducimos que esto se debe de nuevo a la influencia del área de trazado. Obsérvese que el área del segundo ticket es más alargada con una relación ancho/alto de 3.94 frente a un 3.12 del área del primer ticket. Esto parece influir de manera subconsciente en los firmantes que tienden a alargar su firma, y por tanto hacen más grande la firma cuando la caja es menor.

Concluimos que, en la muestra, el tamaño del área de trazado influye en el tamaño de la firma que se realiza. Ésta puede ser la causa de que en muchos trabajos se considere oportuno realizar una normalización de las firmas previas a su procesamiento. Es decir, que el tamaño de la firma no se considere como una característica discriminante.

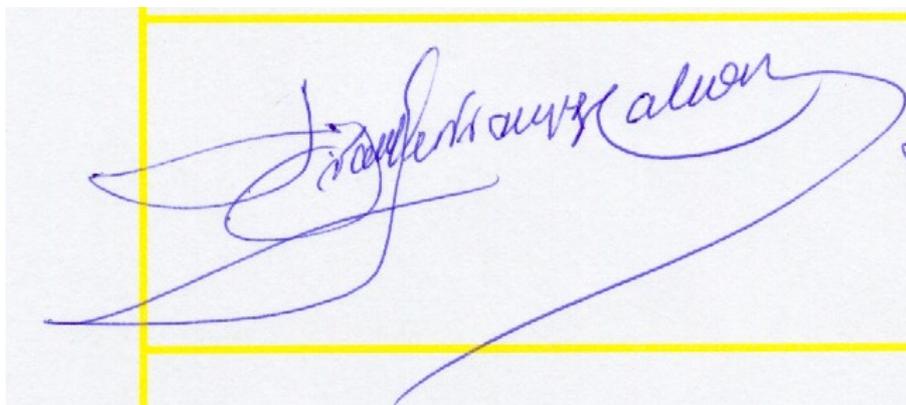
### 3.2.1 Descripción del proceso de recogida de firmas

Durante el proceso de rellenado de la hoja de firmas, al firmante se le pide que cambie de bolígrafo en varias ocasiones. Esta operación tiene por objeto forzar que la muestra sea independiente del tipo de útil de escritura y favorecer el cambio de posturas. En ocasiones los útiles de escritura eran propios del firmante y en otras ocasiones fueron suministrados por el encuestador. Los instrumentos más habituales fueron: bolígrafos de tinta espesa azul o negro (tipo Bic cristal normal), rotulador negro (tipo Edding 1200) y bolígrafos de tinta fluida (tipo Pilot) azul o negro.

El grupo de firmantes siempre ha estado formado por personas accesibles al autor de la Tesis (familiares, amigos, compañeros de trabajo y alumnos) o al entorno de los alumnos que han colaborado, debido a que es difícil que un extraño acceda a realizar las múltiples firmas requeridas. Ello ha motivado, entre otras cosas, que todas las personas sean españolas.

El rango de edades abarca a personas con edades entre 20 y 90 años. A este respecto no se observa diferencias entre los diferentes ejemplares, excepto en los ejemplares (ver Figura 3.3) de una persona sana de avanzada edad (90 años) en los que se aprecian ciertos temblores en los trazos cortos de su firma y adosamientos en algunos trazos largos.

El nivel de estudios de las personas que han prestado su firma comprende a personas con estudios primarios y con estudios universitarios. De nuevo, el entorno del proceso ha motivado que no se hayan encontrado casos de personas sin estudios primarios.



**Figura 3.3.-** Firma correspondiente a un sujeto de 90 años. A simple vista se aprecian ciertos factores caligráficos como adosamientos y temblores en varios puntos del trazado.

Las profesiones de los firmantes son diversas: amas de casa, estudiantes, ferroviarios, informáticos, topógrafos, administrativos, maestros infantiles, matemáticos, sociólogos... Tampoco se aprecia ninguna diferencia entre los ejemplares obtenidos a este respecto.

### **Reticencias a suministrar la firma**

Pese a que el encuestador ha sido siempre una persona de confianza de los firmantes, en general, las personas no relacionadas con este tipo de proyectos de investigación han ofrecido cierta reticencia a suministrar sus firmas. Las dos principales razones argumentadas han sido:

- Cansancio al tener que repetir tantas veces su firma.
- Cómo saber que su firma no se utilizará con fines distintos a los expuestos por el encuestador.

Algunos, tres en realidad, han planteado incluso la posibilidad de realizar una alteración de su firma, cosa que se ha rechazado. Otros incluso han denegado la firma, pese a la confianza existente en la persona que realiza la encuesta.

### **3.2.2 Descripción del proceso de digitalización y preproceso**

En el proceso de digitalización se han tenido en cuenta los siguientes aspectos:

- Resolución espacial: las firmas se han capturado a 300 puntos por pulgada utilizando un escáner de sobremesa que alcanza esa resolución en modo óptico (sin interpolación).
- Resolución radiométrica: las imágenes se han capturado en color real (RGB - 24 bits por píxel).
- Formato de almacenamiento: las imágenes se han almacenado en formato BMP, debido a que este formato no aplica algoritmos de compresión con pérdida y a que su tratamiento es inmediato en el sistema operativo *Windows*.

Tras su digitalización la muestra no puede ser utilizada directamente por los algoritmos de segmentación, reconocimiento y verificación que se presentan a lo largo de esta Tesis. Por ejemplo, la verificación precisa de imágenes en las que sólo aparezca la firma, mientras que la segmentación precisa documentos completos. Además, para simular unas condiciones reales de proceso, las imágenes deben estar en formato bitonal. Los siguientes puntos describen estos procesos.

### **Preparación de la muestra para su uso en algoritmos de reconocimiento y verificación**

Con objeto de contar con muestra de aprendizaje, validación y test para alimentar los diferentes sistemas de clasificación de patrones que se describen en esta Tesis, se debe disponer cada imagen de una firma en un fichero separado. Debido a que los sistemas de procesamiento de documentos reales utilizan imágenes bitonales, las imágenes deben encontrarse en este formato.

El primer objetivo se ha conseguido mediante el recorte manual de las imágenes digitales a partir de los formularios escaneados. Este recorte manual se ha realizado utilizando el programa GIMP de licencia *GPL*<sup>9</sup>. Debe observarse que en ocasiones la firma de una persona sobresale del rectángulo en el que se le ha pedido que firme (ver Figura 3.3). Por ello, como el recorte automático no es trivial, se ha optado por un recorte a mano alzada.

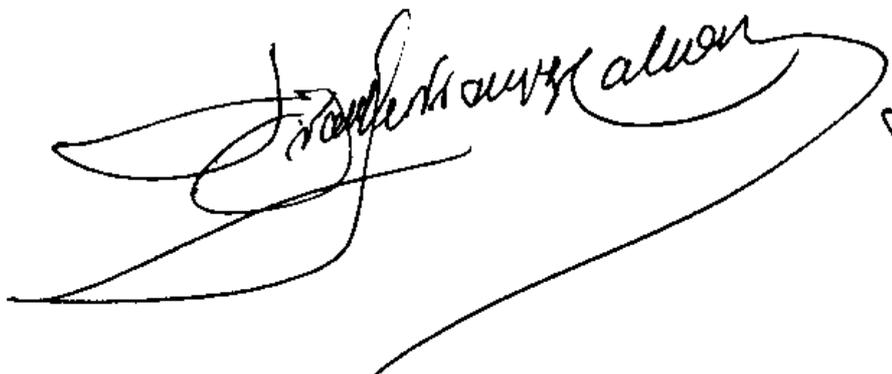
El segundo objetivo, la binarización de las imágenes, ha precisado de dos pasos. En un primer paso la imagen se ha pasado a niveles de gris utilizando de nuevo el programa GIMP. En el paso a niveles de gris, el color amarillo de la tabla (RGB 230,255,0) se ha convertido en un color gris claro

---

<sup>9</sup> Licencia Pública General de GNU (GNU *General Public License*). Es una licencia creada por la *Free Software Foundation* orientada principalmente a proteger la libre distribución, modificación y uso del software.

(aproximadamente, con valor 219). Posteriormente, cada imagen en niveles de gris, ha sido transformada a bitonal utilizando un algoritmo de umbralizado global [25] con valor umbral igual a 212.

Finalmente, las imágenes de las firmas se han almacenado en ficheros *TIFF*<sup>10</sup> utilizando el estándar de compresión sin pérdida CCITT Grupo 4.



**Figura 3.4.-** Resultado de la umbralización de la Figura 3.3.

### **Preparación de la muestra para su uso en algoritmos de segmentación**

En el caso de los algoritmos de segmentación interesa comprobar cómo distinguen los píxeles correspondientes a la firma de los píxeles correspondientes al fondo. Para ello se desea disponer de imágenes que contengan tanto la firma como el fondo. La última muestra recolectada (punto 3.2) proporciona ejemplares de este tipo de firmas en forma de tickets de uso de una tarjeta de crédito. Con objeto de utilizar esta muestra se ha segmentado en un fichero separado la imagen correspondiente a cada uno de estos tickets. Posteriormente, cada imagen se ha transformado a bitonal utilizando un proceso análogo al del punto anterior.

### **3.2.3 Resultado del proceso de captura sobre formularios**

Con objeto de disponer de conjuntos homogéneos para las pruebas, la muestra correspondiente a los formularios se ha dividido en 4 subconjuntos disjuntos. La Tabla 3.1 describe estos grupos. En las columnas de la tabla se muestran los diferentes procesos de recolección que se han realizado. En las cabeceras de las filas se muestran los individuos que han participado en cada proceso de recolección. Los conjuntos que se han creado son:

- *Muestra A.-* Conjunto capturado entre 2002 y 2004, compuesto de entre 4 y 6 ejemplares por cada uno de 27 individuos diferentes.
- *Muestra B.-* Conjunto capturado entre 2002 y 2004, compuesto de 6 ejemplares por cada uno de 27 individuos diferentes.
- *Muestra C.-* Conjunto capturado en 2007, compuesto por 6 ejemplares de 7 individuos que ya prestaron su firma para B.

---

<sup>10</sup> El *Tagged Image Format File (TIFF)* es un formato de fichero digital que sirve de contenedor para imágenes digitales en formatos que vienen especificados en ciertas etiquetas (*tags*) que las acompañan.

- *Muestra D.*- Conjunto capturado en 2007, formado por imágenes correspondientes a reproducciones de tickets firmadas por 38 individuos diferentes.

Estos conjuntos se utilizan de forma indisoluble en el resto de la Tesis, es decir o se utiliza todo el conjunto o no se utiliza ningún individuo de dicho conjunto. Sobre la Tabla 3.1 se puede apreciar que, para la parte experimental de esta tesis, se han desechado algunos ejemplares de firmas. Esto se ha hecho con objeto de homogeneizar los conjuntos de muestras en cuanto a número de imágenes por individuo.

		Descripción				
		2002 4 firmas aisladas	2004 2 firmas aisladas	2004 6 firmas aisladas	2007 6 firmas aisladas	2007 3 firmas en tickets y contratos
Individuos	1 al 17 (17)	X				
	18 al 19 (2)	X			X	X
	20 al 27 (8)	X	X		X	X
	28 al 29 (2)	X	X			
	30 al 47 (18)			X		
	48 al 54 (7)			X	X	X
	55 al 75 (21)				X	X

- Conjunto A
- Conjunto C
- Conjunto B
- Conjunto D

**Tabla 3.1.-** Resumen de la descripción de la muestra recogida. Una cruz en una celda significa que se han recogido las firmas descritas en la cabeza de su columna para los individuos indicados en la cabeza de su fila. Se representan con colores los diferentes conjuntos disjuntos de imágenes de firmas que se han formado.

### 3.3 Muestra procedente de documentos reales

Además de la muestra correspondiente a los formularios descritos en los puntos anteriores, también se ha recolectado diversa muestra correspondiente documentos reales. Esta muestra será de utilidad para comprobar el funcionamiento de un sistema completo que segmente y verifique una firma en condiciones reales.

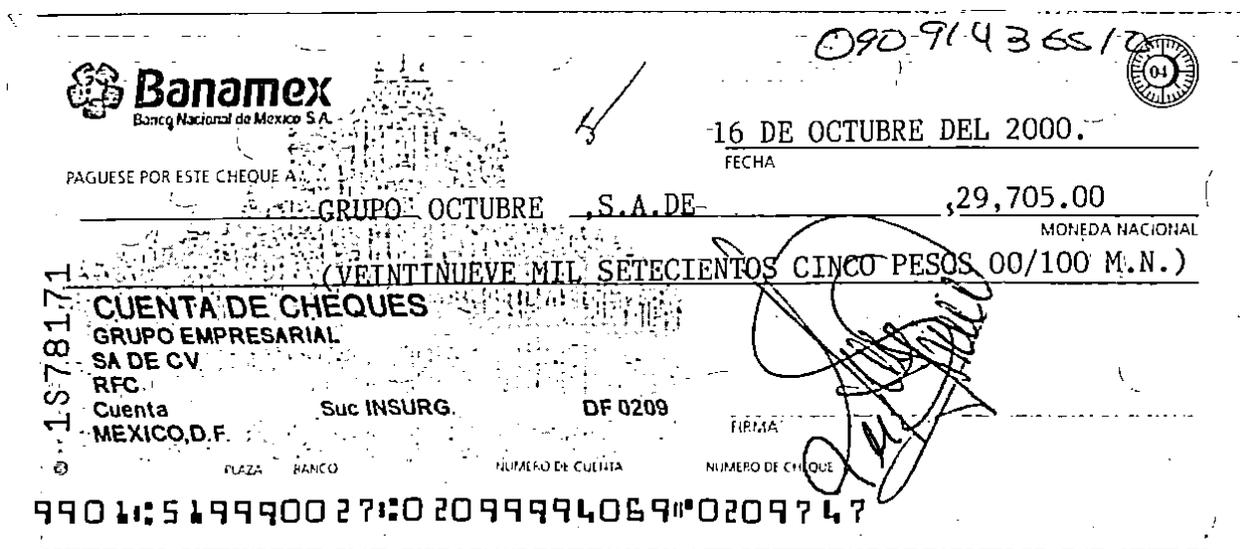
#### Documentos bancarios

La empresa Investigación y Programas S.A, ha suministrado 800 ejemplares reales de tipo cheque de origen hispanoamericano y 295 recibos españoles. Esta muestra, debido a su carácter confidencial, no puede ser facilitada a terceros. La Figura 3.5 presenta un ejemplar de dicha muestra, debidamente alterado para preservar la confidencialidad.

De este amplio conjunto de documentos se ha seleccionado un subconjunto mucho más reducido, compuesto de 120 imágenes correspondientes a 37 individuos diferentes. En dicha selección se tuvieron en cuenta principalmente dos aspectos:

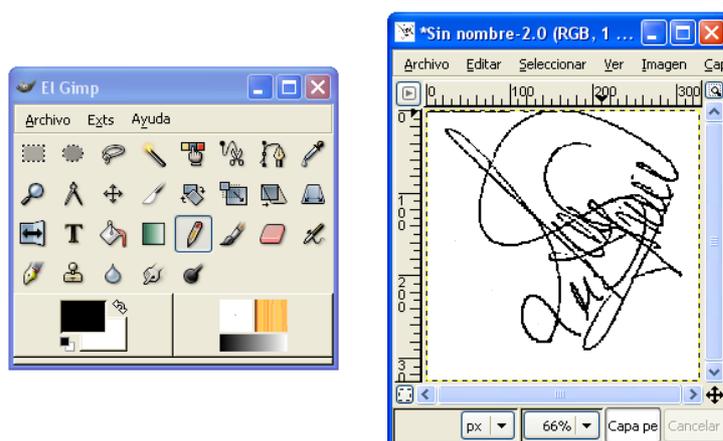
- que existiesen al menos 4 ejemplares de cada firmante, y
- que la tipología de los ejemplares fuese lo más variada posible.

Todas las imágenes han sido escaneadas a 200 DPIs. Su resolución radiométrica es bitonal y es el resultado de la aplicación de un umbralizado adaptativo mediante ventana deslizante proporcionado por el *hardware* del escáner utilizado (una lectoclasificadora NCR 3770 de alta productividad). Esta muestra se etiqueta como *muestra E*.



**Figura 3.5.-** Ejemplo de documento proporcionado por la empresa Investigación y Programas. Dado su carácter privado el documento ha sido alterado digitalmente para posibilitar su publicación.

Sobre esta muestra ha sido necesario realizar un delicado proceso de limpieza de una firma por cada firmante detectado. Este limpieza ha tenido como objetivo el disponer de un ejemplar de referencia por cada firmante. Este proceso se ha realizado manualmente utilizando la herramienta de retoque GIMP (ver Figura 3.6) y su herramienta de brocha.



**Figura 3.6.-** Detalle del proceso de limpieza sobre documentos reales.

## Documentos generales

Además de la muestra correspondiente a firmas se ha recogido muestra de 10 documentos manuscritos y 10 mecanografiados de carácter general. Esta muestra, que se etiqueta como *muestra F*, se utiliza para el análisis del contexto en el que puede aparecer una firma.

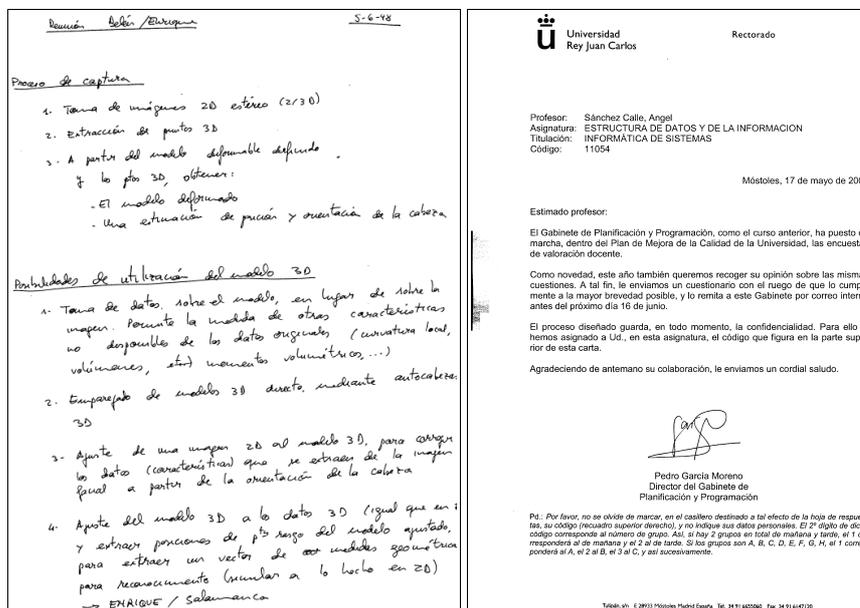


Figura 3.7.- Ejemplo de un documento manuscrito y otro mecanografiado de la muestra F.

## 3.4 Segmentación automática de la muestra real

En esta sección se presenta un método heurístico de segmentación que se ha diseñado para separar las firmas de la muestra E del fondo que las acompaña. En general, cualquier proceso de segmentación de firmas *off-line* consiste decidir para cada píxel activo de una imagen si pertenece o no a la firma de un individuo cuya identidad se desea verificar. El método que se presenta es totalmente automático, por lo que formaría parte de una solución completa al problema de verificación en condiciones realistas.

La solución que se propone para el problema de la segmentación de la firma dentro de un documento obtenido en condiciones realistas se apoya en el conocimiento de la firma que se desea segmentar. Este enfoque no es general, en el sentido de que no se puede utilizar para segmentar firmas, si se desconoce la firma que se busca. Sin embargo, es un enfoque practicable en el caso de la verificación pues la firma que se busca es conocida.

El proceso de segmentación que se ha diseñado se debe acotar a la zona aproximada del documento donde se sabe que se encontrará la firma (por ejemplo, en un cheque se aplicará sobre el cuarto derecho inferior del mismo) y consta de dos fases. En la primera se realiza un filtrado morfológico que elimina parte del ruido presente en la imagen. En la segunda fase se realiza la segmentación de la firma. La descripción detallada de estas dos fases vertebrará esta sección.

### 3.4.1 Etapa de filtrado morfológico

Antes de proceder a una discriminación sutil de aquellos elementos de la imagen que parecen pertenecer a la firma y de aquéllos que no, conviene realizar un filtrado de “grano grueso” que

elimine los elementos que por sus características morfológicas es seguro que no pertenecen a la firma.

La etapa de filtrado consta a su vez de dos fases. En la primera, mediante un filtrado de erosión, se eliminan las líneas horizontales y verticales. En la segunda fase se elimina el ruido blanco que pueda encontrarse en el documento (ver 2.2.4). En los siguientes puntos se describen en detalle estas dos fases.

### **Líneas en los documentos**

Habitualmente, en los documentos en los que se requiere una firma se dispone de una línea horizontal sobre la que se solicita su realización (ver subsección 2.2.4). Esta línea debe ser eliminada por el proceso de segmentación. Dada su especial geometría es un elemento fácil de detectar y por ello su eliminación se aborda en primer lugar.

Clásicamente, la detección de las líneas rectas dentro una imagen se realiza mediante la transformada de Hough [169]. Esta transformada convierte los puntos de una imagen del dominio espacial al dominio de los parámetros de las rectas que los contienen. En este nuevo dominio las rectas se detectan fácilmente porque se corresponden a las intersecciones de las curvas que se generan en la transformación.

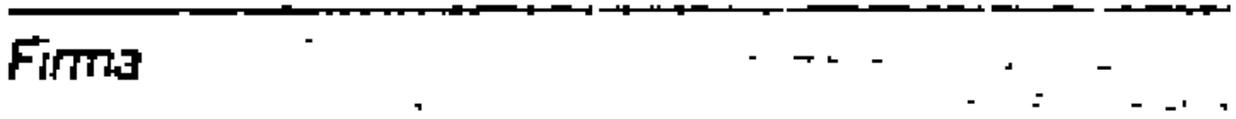
Sin embargo, en el contexto de las imágenes de documentos, la transformada de Hough no resuelve problemas como la detección del punto de inicio y fin de los segmentos, o la detección de líneas de grosor variable como una sola línea. Además, la transformada sólo contempla la detección de las líneas, y no contempla ni la eliminación, ni la reconstrucción de los objetos atravesados por ellas.

Para solventar estos problemas S. H. Kim y otros [170] proponen un método de detección de líneas basado en 4 pasos: preproceso, detección, eliminación y reconstrucción. La idea fundamental de este método consiste en el análisis de las rachas de píxeles activos presentes en la imagen. Este método soluciona las objeciones realizadas al enfoque basado en la transformada de Hough.

El análisis que realizan Kim y su grupo se centra fundamentalmente en la eliminación de las rectas de cualquier tamaño y de cualquier inclinación. Sin embargo, en el problema de la verificación de firmas *off-line* hemos visto que las firmas pueden contener segmentos rectos, y su eliminación arbitraria puede destruir la firma que se pretende verificar. También debe tenerse en cuenta que los documentos sobre los que se requiere realizar la verificación de firmas suelen haber sido preprocesados para eliminar la inclinación de la imagen resultante de una alimentación no horizontal del escáner. Por ello no suelen contener líneas con orientaciones diversas, sino sólo líneas casi perfectamente horizontales y verticales. Además, el aspecto de las líneas horizontales tiene un característico efecto de “almenas de muralla” (ver Figura 3.8). Se observa que las líneas horizontales y verticales oscilan levemente un píxel arriba o abajo a lo largo de su trayectoria. Este efecto se debe a la naturaleza discreta del algoritmo de giro aplicado para la corrección de la inclinación del documento. Estos problemas han motivado que se desarrolle un método específico para la eliminación de estas líneas. La siguiente sección describe este algoritmo.

### **Eliminación de líneas horizontales y verticales**

El procedimiento que se ha diseñado para la eliminación de las líneas horizontales y verticales sigue la secuencia de pasos propuesta por S. H. Kim y otros [170]. Sin embargo, se han añadido diversas heurísticas para adaptar el método a la problemática de la segmentación de firmas. En primer lugar se tratará el problema de las líneas horizontales, para luego tratar el particular a las verticales.



**Figura 3.8.-** Ejemplo de oscilación en una línea horizontal, resultado de la aplicación del algoritmo de giro utilizado para corregir la inclinación de la imagen del documento.

### Eliminación de líneas horizontales

La eliminación de líneas horizontales se basa en el concepto de racha. Se denomina racha a una secuencia horizontal de píxeles de ancho variable y un píxel de alto. Estas rachas se pueden representar mediante una tripleta compuesta de las coordenadas  $(x,y)$  de inicio y el tamaño horizontal de la racha.

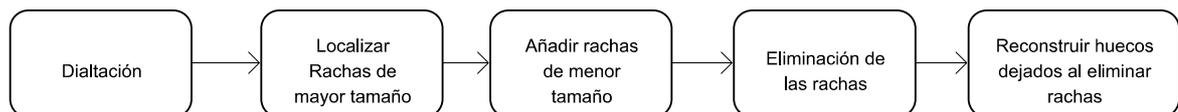
El algoritmo que se propone (ver Figura 3.9) consta de 5 fases. En la primera fase se realiza una dilatación, utilizando un elemento estructurante horizontal, que trata de eliminar las discontinuidades observadas en la Figura 3.8. En la segunda se localizan las rachas de mayor tamaño utilizando un umbral  $R_1$ . Estas rachas se supone que se corresponden con las bases de las líneas que se desea eliminar. En el tercer paso se itera un bucle que, utilizando un umbral  $R_2$ , añade las rachas de menor entidad a la base de rachas existentes. Estas rachas de menor entidad se corresponden a los trozos con forma de almenas de la Figura 3.8. En un cuarto paso, se eliminan las rachas detectadas de la imagen. Finalmente, en una quinta fase se intenta reconstruir los huecos que la eliminación de rachas pueda haber causado sobre la firma manuscrita. Para ello se dilata la imagen resultante del proceso anterior con un elemento estructurante vertical y se hace *AND* del resultado con la imagen original. El Algoritmo 3.1 describe formalmente el proceso.

Obsérvese que el umbral  $R_1$  deberá ser relativo a la zona prevista de firmado. Además, este umbral determina que las líneas que se eliminarán no sólo serán las horizontales sino también aquellas con ángulo  $a$  dado por (3.1):

$$\alpha = \arctg\left(\frac{t}{R_1}\right) \quad (3.1)$$

donde  $t$  el grosor del trazo de la línea en píxeles.

Así, suponiendo que un cheque está escaneado a 200 *DPIs* (por tanto el trazo de las líneas horizontales presentes en el mismo es de 3 píxeles) y suponiendo que  $R_1$  se fija a 40 píxeles, el procedimiento descrito detectaría las líneas con ángulos entre  $-4.28^\circ$  y  $4.28^\circ$ . Por ello, cuanto mayor sea el valor de  $R_1$  más seguridad se tendrá en eliminar únicamente las líneas horizontales.



**Figura 3.9.-** Diagrama *UML* que presenta las 5 fases del algoritmo de eliminación de líneas.

La elección de  $R_2$  debe realizarse cuidadosamente. Un valor alto hará que no se elimine ninguna racha, mientras que un valor bajo podría resultar en la eliminación de trazos de la firma que intersecten la línea horizontal que se desea eliminar. Por ello, los valores de este parámetro se dan en función del grosor del trazo esperado para esa resolución (por ejemplo, el triple del grosor del trazo).

### Eliminación de líneas verticales

La eliminación de líneas verticales auxiliares podría seguir un camino análogo al descrito en la sección anterior. Sin embargo, deben observarse varias peculiaridades respecto a este tipo de líneas:

- Las líneas verticales aparecen en el documento en un número menor de ocasiones, no siendo habitual firmar sobre una línea vertical. Si aparecen, suelen hacerlo a los lados de la zona de firma para no intersectarla.
- Además, estas líneas suelen tener un tamaño pequeño. Esto puede relacionarse con el hecho, observado en la muestra A, de que el ancho medio de las firmas es mayor que el alto.
- No es habitual firmar sobre una línea vertical.

Estas 3 peculiaridades influyen en las siguientes modificaciones respecto al Algoritmo 3.1:

- El umbral  $R_1$  debe reducirse.
- Si la racha a eliminar supera un tercer valor umbral  $R_3$  no se eliminará.

#### Algoritmo 3.1 de eliminación de líneas horizontales

**Nombre:** EliminarLineasHorizontales

**Entradas:**  $I_o$  : Imagen original  
 $D_1$  y  $D_2$  : Elementos estructurantes horizontal y vertical  
 $R_1$  y  $R_2$  : Umbral para rachas grandes y para rachas pequeñas

**Salida:**  $I_s$  : Imagen resultado

```

1 -  $I_d = I_o \oplus D_1$ 
2 -  $L = \text{BuscarRachas}(I_d, R_1)$ 
3 -  $L_{aux} = \text{BuscarRachas}(I_d, R_2)$ 
4 - do
5 -      $L_2 = L$ 
6 -     for each  $l$  in  $L_{aux}$ 
7 -         if  $\text{Adyacente}(l, L)$ 
8 -              $L = l + L$ 
9 -     while  $(L \neq L_2)$ 
10 -  $I_s = (\text{Eliminar}(I_d, L) \oplus D_2) \& I_o$ 

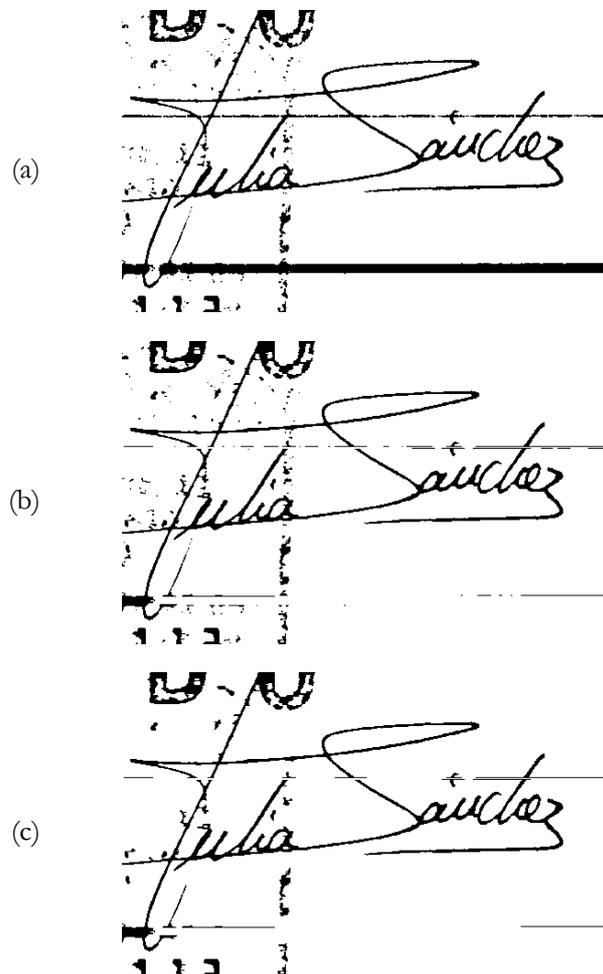
```

### Eliminación morfológica de ruido blanco

Se ha comentado en la subsección 2.2.4 que las firmas suelen estar salpicadas de ruido, en forma de objetos conexos comparativamente de pequeño tamaño, que en una imagen bitonal parece aleatorio. Además, tras la aplicación del filtrado de eliminación de líneas puede quedar algún residuo que también parece ruido aleatorio. Es por ello, que la aplicación de un filtrado que elimine este tipo de ruido resulta recomendable.

La eliminación del ruido blanco se realiza mediante dos métodos:

- Se eliminan las componentes conexas de la imagen con un tamaño en píxeles inferior a un umbral  $W$  que debe elegirse en función de la resolución de la imagen.
- Se eliminan las componentes conexas cuyo tamaño en número de píxeles es superior a  $2t$  veces el tamaño en píxeles del esqueleto del objeto, siendo  $t$  el tamaño esperado para los trazos a esa resolución.



**Figura 3.10.-** Ejemplo de eliminación de líneas sobre el cuarto inferior de un cheque: (a) presenta una parte de un cheque salpicado de líneas horizontales, (b) muestra la imagen tras el proceso de eliminación de líneas horizontales y (c) el resultado de eliminación del ruido.

### 3.4.2 Etapa de selección y descarte de objetos conexos

Es probable que las componentes conexas que se han mantenido tras los filtrados realizados en las secciones precedentes correspondan a la firma que se desea verificar. Sin embargo, también podrían corresponder a restos de sellos, líneas, otros caracteres adyacentes a la firma o a ruido en general.

En las siguientes secciones se propone un método heurístico que determina las componentes conexas que pertenecen a la firma a verificar y las que deben desecharse. Este método se basa en la generación de un conjunto de propuestas, creadas mediante combinatoria a partir de las componentes involucradas, y su descarte basado en la comparación contra un modelo de la firma a verificar.

#### Análisis de las componentes conexas

El recuento del número de componentes conexas presentes en las firmas de la muestra A permite obtener el gráfico de barras de la Figura 3.11. En éste se puede apreciar que no hay ninguna imagen que tenga más de 8 componentes conexas, siendo lo más probable que sólo tenga una o dos componentes.

Analizando el tamaño de los objetos según el número de componentes conexas en la misma muestra A se confecciona el gráfico de la Figura 3.12. Este gráfico se ha construido analizando los rectángulos horizontales mínimos que contienen a las diferentes componentes conexas (*bounding box*) de las firmas de la muestra A. El gráfico relaciona el área de los rectángulos, con el número componentes, y con el número de firmas. Por ejemplo, la casilla enmarcada en un círculo rojo indica que hay 1 firma (debido a su tono azul) con 6 componentes conexas de gran tamaño (250000 píxeles  $\approx$  500x500). En este gráfico se aprecia que la mayor densidad, la zona de colores cálidos, se encuentra para las componentes conexas de pequeño tamaño, en el margen izquierdo del diagrama. De este gráfico se desprende que las firmas casi siempre constan de una componente conexas de gran tamaño y, en algunas ocasiones, de otras componentes conexas adyacentes de menor tamaño.

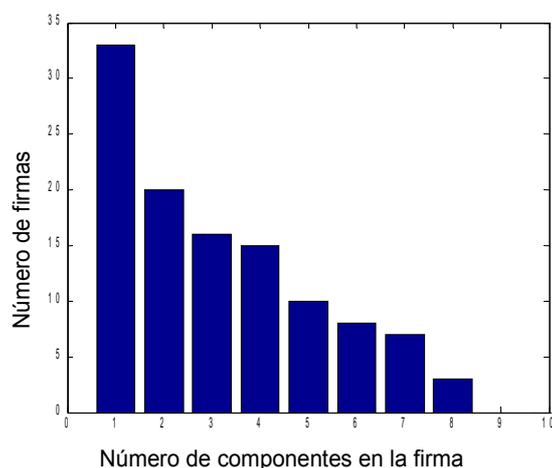
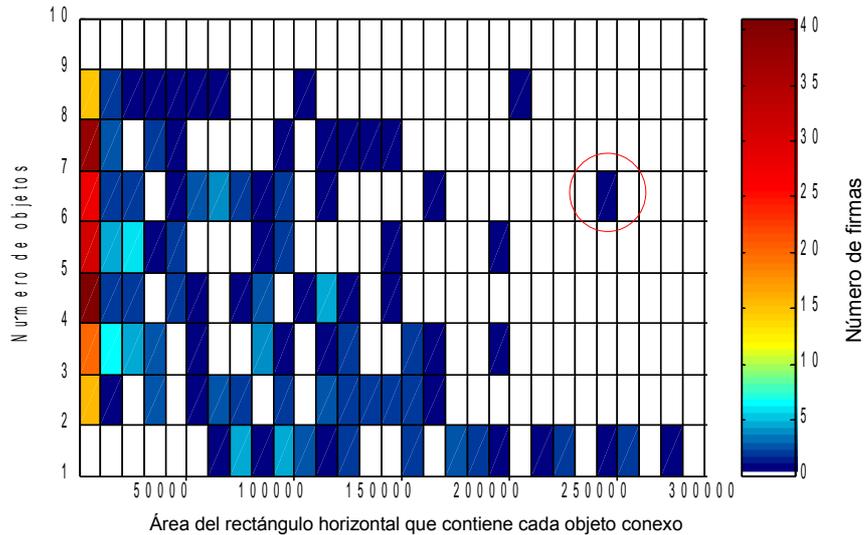


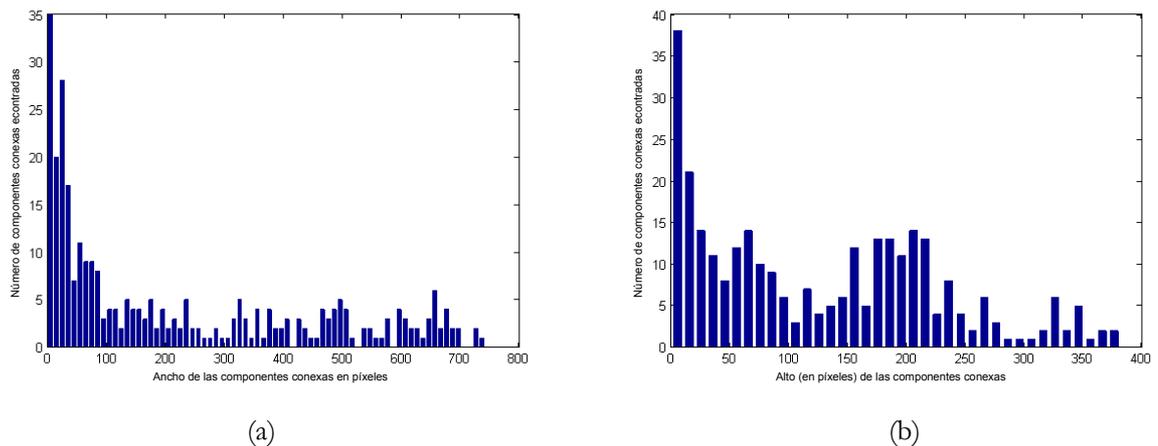
Figura 3.11.- Número de componentes conexas por firma.

El análisis de los documentos mecanografiados de la muestra F (ver Figura 3.13) ha revelado que los objetos conexos que en ellos aparecen, en media, son pequeños comparados con los tamaños de las firmas. Esto se debe a que en los documentos mecanografiados las letras aparecen como objetos conexos separados, y las letras tienen un tamaño pequeño (por ejemplo la letra “M” mayúscula con fuente Times 12 a 300 DPIs tiene 40x40 píxeles por objeto). Sólo los objetos gráficos y las firmas

corresponden a objetos de área grande (para las firmas de la muestra A se ha encontrado una media de 200x150 píxeles). Por otro lado, el análisis de los documentos con texto manuscrito de la muestra F revela que la distancia, entre el tamaño de las componentes conexas del texto y el de las firmas, se reduce pero sigue siendo significativa.



**Figura 3.12.-** Relación entre el área de los rectángulos horizontales que contienen a los objetos conexas de una firma, con el número de tales objetos, y con el número de firmas. El color azul de la casilla enmarcada en un círculo indica que hay 1 firma con 6 componentes conexas de tamaño grande.



**Figura 3.13.-** Tamaño de las componentes conexas encontradas sobre los documento de la muestra F. El eje  $x$  de los histogramas (a) y (b) presentan respectivamente el ancho y el alto de las componentes conexas. El eje  $y$  de ambos diagramas presenta el número de apariciones medio de tales componentes encontrados en los documentos de dicha muestra.

Estos hechos conducen a una hipótesis: dentro de un documento, las componentes conexas de la firma se corresponden habitualmente con las de mayor tamaño. Otra hipótesis que se plantea consiste en que la posición de la firma dentro de un documento suele estar localizada en una zona reducida. Por ejemplo, dentro de un cheque suele estar situada en la parte inferior derecha, y dentro de un documento de texto mecanografiado suele estar tras el texto, normalmente justificada a la derecha o centrada.

Sin embargo, estas dos hipótesis son insuficientes para descartar otros objetos conexos de gran tamaño que pueden estar cercanos a la firma (como sellos, otras palabras manuscritas u otra firma). Para salvar este obstáculo podemos considerar un tercer hecho: como el objetivo es verificar una firma conocida, se puede aprovechar la información que se tiene sobre el número de componentes conexas que debe tener y las posiciones relativas de las mismas.

Así, partiendo de estas 3 hipótesis:

- componentes conexas de gran tamaño,
- posición aproximada de la firma en el documento,
- y conocimiento sobre el aspecto y la posición relativa de las componentes conexas de la firma modelo,

se ha construido un método heurístico que permite segmentar la firma de un documento de carácter general con vistas a su verificación.

### Idea general del método de evaluación de selección de componentes conexas

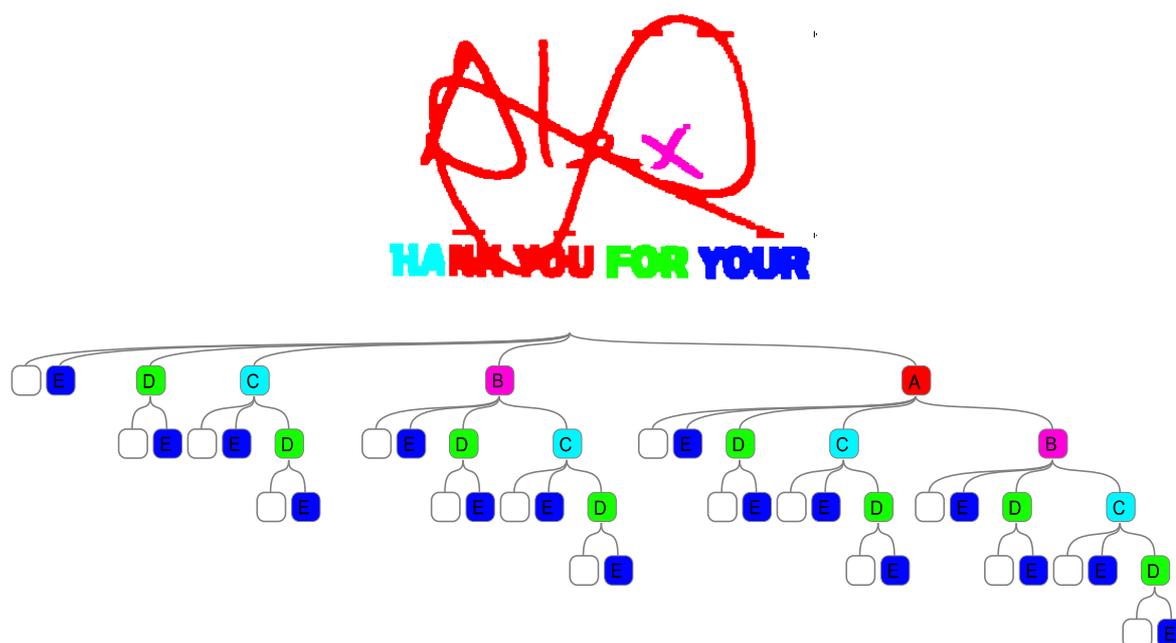
Como ya se ha señalado, tras los filtrados, se dispone de un conjunto de componentes conexas. Algunas de estas componentes corresponderán a la firma, mientras que otras corresponderán a elementos adyacentes a la firma que se considera ruido. Dado el escaso número de componentes es posible utilizar un algoritmo de fuerza bruta que genere imágenes con las diferentes combinaciones posibles de componentes conexas y luego las evalúe en busca de aquella que mejor se aproxime a la firma buscada. La evaluación de esta similitud se realizaría utilizando la imagen modelo de la firma a segmentar.

### Generación de combinaciones de componentes conexas

El algoritmo que genera diferentes imágenes mediante la combinación de las diferentes componentes conexas es trivial. La Figura 3.14 presenta un ejemplo de la aplicación de tal algoritmo para una imagen que consta de 5 componentes conexas. En la figura se denominan A, B, C, D y E a estas componentes y se presentan en diferentes colores para facilitar su visualización. El árbol de la Figura 3.14 presenta las posibles combinaciones de estas componentes. Entre estas combinaciones, posteriormente, se deberá seleccionar una de ellas, utilizando como criterio el parecido con la forma conocida de la firma a verificar. En la figura, la combinación de componentes que corresponde a la firma y descarta el ruido es la que está compuesta por la componente A (en rojo) y B (en rosa).

Así, para  $N$  componentes conexas, el número de posibilidades viene determinado por la suma de las posibles combinaciones de esos  $N$  elementos tomados de uno en uno, de dos en dos, etc.

$$combinaciones = \sum_{n=0}^N \binom{N}{N-n} \quad (3.2)$$



**Figura 3.14.-** Árbol que muestra las combinaciones posibles de las componentes conexas de la imagen de una firma. Arriba, la imagen de una firma junto con su entorno ruidoso constituida por 5 componentes conexas.

Para completar el proceso de segmentación, falta concretar cuál de las combinaciones generadas corresponde a la segmentación correcta de la firma.

### Comparación de las propuestas

En esta sección se propone un algoritmo que, de manera rápida y robusta, permite elegir una de las combinaciones como resultado de la segmentación.

Este problema encuentra ciertas similitudes con el problema de identificación ya que se trata de determinar cuál es la firma más parecida a una dada de un conjunto de firmas predeterminado. Sin embargo, encontramos varias diferencias importantes:

- De nuevo, sólo se dispone de una única firma modelo para comparar las múltiples combinaciones. Esto contrasta con las principales aproximaciones a este problema en las que se utilizan múltiples firmas de cada individuo.
- El problema es similar al de identificación en cuanto a la cardinalidad de los conjuntos involucrados. Sin embargo, en este caso, todas las firmas con las que se puede comparar la firma modelo son muy parecidas, ya que son el resultado de un proceso combinatorio de las mismas componentes conexas. Además, algunas de las imágenes a comparar pueden no ser firmas, sino objetos de otra naturaleza que hayan sobrevivido a los filtrados previos.
- Por último, si se desea que este método de segmentación se pueda emplear en la práctica, dado el número de combinaciones que puede haber resultado de la aplicación del algoritmo anterior, el método de identificación que se use debe ser computacionalmente poco costoso.

Estas peculiaridades motivan que no todas las características discriminantes utilizadas para la identificación de firmas puedan utilizarse para esta comparación. Se ha comprobado

experimentalmente que muchas de las características que habitualmente se utilizan en verificación de firmas no aportan información discriminante cuando se usan sobre la muestra resultante de los procesos de filtrado previos. Esto se debe a la varianza introducida sobre estas características por el ruido y por tales procesos de filtrado. Por ello, se han desechado las características de “grano fino” y se ha optado por utilizar una única característica de “grano grueso”: la comparación de las formas de las envolventes conexas de la firma modelo con las envolventes conexas de las diferentes posibilidades de segmentación. La envolvente conexas como única característica ha sido estudiada por otros autores [171]. En general, su uso sobre conjuntos grandes de firmas podría dar lugar a múltiples falsos positivos, ya que es probable que la envolvente de dos firmas coincida, pero su interior no. Sin embargo, para este caso, en el que el ruido adyacente hace cambiar la forma de la firma y no nos interesa tanto su interior, esto no es un inconveniente sino una ventaja.

### Comparación de firmas basada en sus envolventes conexas

Como paso previo a la comparación de envolventes conexas se procede a una normalización de las imágenes involucradas. Esta normalización se realiza en dos sentidos: en área y en ángulo. La normalización en área consiste en aumentar el tamaño del polígono con menor área para que iguale el área del otro polígono. Este aumento se realiza mediante un reescalado que mantenga las proporciones (ancho-alto) del polígono. Mientras, la normalización en ángulo se realiza girando uno de los polígonos para que su eje mayor coincida con el del otro.

Se pueden idear diferentes métodos de comparación para las áreas de las envolventes conexas [58]. Aquí se usa una idea simple, que consiste en verificar el solapamiento que se produce entre los dos polígonos resultantes. Así, tras la normalización, se procede a la evaluación del solapamiento que se realiza superponiendo ambas envolventes, utilizando sus centros de masas para alinearlas, y situando una retícula sobre ambas. Cada casilla de tal retícula se puede considerar coincidente o no coincidente según se solapen o no ambos polígonos sobre ella (ver Figura 3.15). Así, la comparación consiste en un proceso discreto de contabilización de coincidencias entre las casillas rellenas de la retícula. Como tamaño de retícula se utiliza el pixelado de la imagen menor. Así, si  $r_{eval}$  se corresponde con la imagen binaria de la retícula de la firma a evaluar y  $r_{mod}$  con la imagen binaria de la retícula de la firma modelo, el porcentaje de coincidencia viene dada por la expresión:

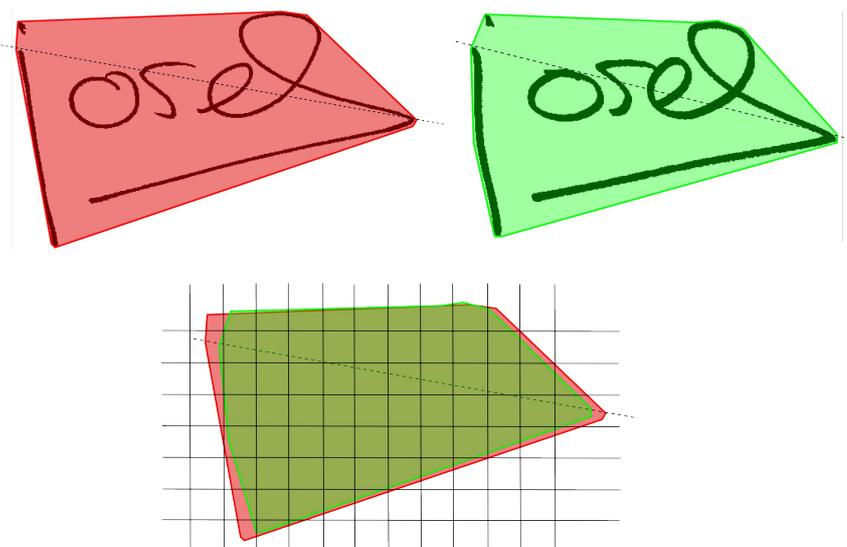
$$coincidencia = \frac{\sum_{x=1}^{ancho} \sum_{y=1}^{alto} \min(r_{eval}(x, y), r_{mod}(x, y))}{\sum_{x=1}^{ancho} \sum_{y=1}^{alto} \max(r_{eval}(x, y), r_{mod}(x, y))} \quad (3.3)$$

donde ancho y alto se refieren al tamaño del mínimo rectángulo horizontal que contiene las envolventes conexas.

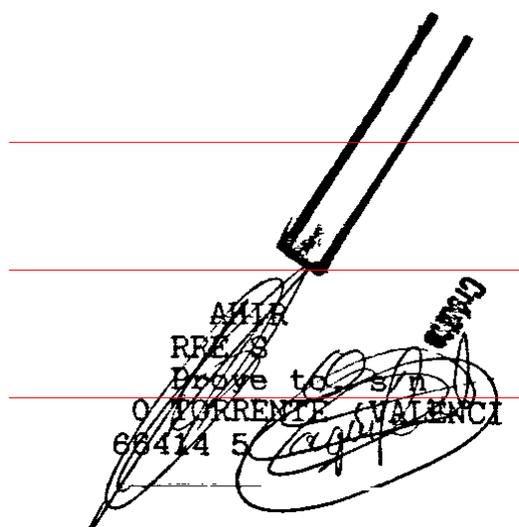
Así, de entre todas las envolventes conexas, resultantes del proceso combinatorio sobre las componentes conexas, se elige como mejor aproximación a la segmentación aquella que mayor valor ofrezca para el porcentaje de coincidencia (definida en 3.3).

### Eliminación de sellos y otros objetos de gran tamaño mediante sobresegmentación

Los procedimientos detallados hasta el momento proporcionan buenos resultados siempre que las componentes conexas que forman la firma no se solapen con componentes conexas de gran tamaño pertenecientes a sellos u otros elementos adyacentes. En algunos de estos casos, como en el ejemplo de la Figura 3.16, el algoritmo propuesto no es capaz de eliminar estos elementos y el resultado de la eliminación es deficiente.



**Figura 3.15.-** Comparación entre las envolventes conexas. La del modelo (rojo) y las componentes a segmentar (verde) se comparan sobre una rejilla. Previamente se normalizan en ángulo y área ambas envolventes.



**Figura 3.16.-**Ejemplo de ruido que no ha sido eliminado en la segmentación. Las líneas rojas señalan la sobresegmentación vertical que se realiza.

Para solventar estos casos se ha propuesto una solución basada en la sobresegmentación previa de las componentes conexas de gran tamaño. Así, si alguna de las componentes conexas de la firma tiene una dimensión superior a un valor umbral  $H$ , se segmenta esa dimensión en dos mitades. Este valor umbral  $H$  deberá fijarse en cada caso función del tamaño del área aproximada de firmado.

### Resumen del algoritmo de segmentación

El Algoritmo 3.2 agrupa y ordena el conjunto de acciones que se han presentado en las secciones precedentes, conformando el método de segmentación de firmas que propone esta Tesis.

**Algoritmo 3.2 de segmentación de firmas manuscritas****Nombre:** SegmentarFirma

**Entradas:**  $I_E$  : Imagen de un documento  
 $A$  : Ancho del documentos en centímetros  
 $H$  : Alto del documento en centímetros  
 $I_M$  : Imagen modelo de la firma a segmentar

**Salida:**  $I_S$  : Imagen segmentada de la firma

```

1 -       $I_{LH}$  = EliminarLineasHorizontales( $I_E, A, H$ )
2 -       $I_{LV}$  = EliminarLineasVerticales( $I_{LH}, A, H$ )
3 -       $I_{LR}$  = EliminarRuidoBlanco( $I_{LV}, A, H$ )
4 -       $L$  = ObtenerComponentesConexas( $I_{LR}$ )
5 -       $L_S$  = SobresegmentarComponentes( $L, H, V$ )
6 -       $L_C$  = GenerarCombinaciones( $L_S$ )
7 -       $E_M$  = CalcularEnvolventeConvexa( $I_M$ )
8 -      mejor_coincidencia = 0
9 -      candidato = []
10 -     for each  $c$  in  $L_C$  //Cada  $c$  es un conjunto de componentes conexas
11 -          $E_C$  = CalcularEnvolventeConvexa( $c$ )
12 -         comparacion_actual = Comparar( $E_M, E_C$ )
13 -         if (comparacion_actual > mejor_coincidencia)
14 -             mejor_coincidencia = comparacion_actual
15 -             candidato =  $c$ 
16 -      $I_S$  = Recortar( $I_E, c$ )

```

**3.4.3 Resultados de los experimentos de segmentación sobre muestra real**

Para este experimento se ha utilizado la muestra E. Como se ha descrito en la sección 3.2.3 esta muestra corresponde a 120 documentos reales, con firmas de 37 individuos que se repiten en los diferentes documentos. Esta muestra es ideal para realizar experimentos de segmentación ya que presentan a la vez toda la problemática descrita en el segundo capítulo. Dado que el formato de los documentos era tipo cheque el umbral  $H$  de sobresegmentación se ha fijado en 2 centímetros.

Las imágenes resultantes de este proceso de segmentación sobre documentos reales se han estudiado desde dos puntos de vista. Por un lado, como se comprobará en los dos capítulos posteriores, el resultado de esta segmentación se ha utilizado en diversos procesos de verificación con resultados positivos. Por otro lado, se ha medido manualmente la relación señal/ruido de los documentos y la relación señal/ruido del resultado de la segmentación (ver Figura 3.17). Los resultados medios se exponen en las Tablas 3.2 y 3.3. En ellos se puede destacar que el ruido se ha disminuido de un 91% a un 22% aproximadamente, mientras que en media sólo se pierde el 2% de los píxeles de la firma. Además, en la mayoría de los casos la firma queda perfectamente localizada.



**Figura 3.17.-** Ejemplo del etiquetado manual realizado sobre el resultado de la segmentación. En negro los píxeles que realmente corresponden a la firma, en rojo píxeles correspondiente a ruido que no se ha filtrado correctamente.

	Porcentaje de píxeles correspondientes a la firma	Porcentaje de píxeles correspondientes a otros elementos
En documentos completos	8.79 %	91.21 %
En las firmas segmentadas	77.57 %	22.43 %

**Tabla 3.2.-** Características de las imágenes de la muestra E utilizadas para las pruebas de segmentación.

	Porcentaje
Píxeles erróneamente etiquetados como pertenecientes a la firma ( <i>FAR</i> ) respecto al número de píxeles del documento	2.70 %
Píxeles erróneamente etiquetados como ruido ( <i>FRR</i> ) respecto al número de píxeles del documento	1.99%

**Tabla 3.3.-** Resultados de la segmentación sobre documentos reales.

### 3.5 Resumen del capítulo

En este capítulo se ha justificado la creación de una base de datos de firmas manuscritas, y se han detallado los procesos de captura y segmentación de firmas que se han realizado.

Como resultado se han obtenido 6 conjuntos de imágenes (que se han nombrado utilizando las 6 primeras letras del alfabeto). La muestra A, compuesta de 6 firmas aisladas por cada uno de 28 individuos diferentes que se utilizará para el diseño de algoritmos de verificación y para las pruebas de validación. La muestra B, compuesta por 6 firmas aisladas de 27 individuos diferentes que se utilizará para el test de los algoritmos de verificación. La muestra C, compuesta por 6 ejemplares de 7 individuos que ya prestaron su firma para la muestra B. La muestra D, compuesta de imágenes de tickets y de contratos que se ha utilizado para diseñar y probar los algoritmos de segmentación. La muestra E, correspondiente a documentos bancarios reales, compuesta de varios ejemplares por cada uno de 37 individuos, que se ha utilizado para probar los algoritmos de segmentación sobre

muestra real. Por último, la muestra F, compuesta de 20 documentos de diverso origen, que también se ha utilizado para el diseño de los algoritmos de segmentación.

Finalmente, se ha presentado un algoritmo de segmentación de firmas que se ha diseñado utilizando la muestra C y que se ha aplicado sobre la muestra E. Como resultado de esta segmentación se ha obtenido un conjunto de imágenes que se utilizará en las pruebas de verificación, que se realizarán en los capítulos posteriores, como si de la muestra E se tratase.

## Capítulo 4 Verificación *off-line* de firmas utilizando muestra sintética

En este capítulo se presentan diferentes métodos orientados a la verificación *off-line* de firmas manuscritas basados en el uso de muestra de aprendizaje sintética generada a partir de una única muestra modelo. Es por ello que el capítulo comienza con una exposición del método propuesto para la generación automática de la muestra sintética.

A continuación, se presenta un conjunto de propuestas para la verificación basadas en esta muestra sintética. Este capítulo describe diferentes desarrollos que se han realizado en los últimos 5 años. Debido a la amplitud de este periodo de tiempo los métodos están sujetos a una evolución que, partiendo de unos enfoques cercanos a la literatura clásica, se dirigen a enfoques más novedosos que los complementan.

Esta evolución se refleja en la exposición del capítulo. En primer lugar se experimenta utilizando directamente la imagen de la firma sobre una Red de Neuronas de tipo Perceptrón Multicapa (*Multilayer Perceptrón* o abreviadamente *MLP*). Después, se realizan experimentos utilizando un enfoque clásico basado en la extracción de características discriminantes. Finalmente, se exponen dos propuestas originales basadas en Redes de Compresión.

En general todos los trabajos siguen el esquema representado por el Diagrama de Actividad de la Figura 1.3. Este esquema de alto nivel permite apreciar las 3 actividades necesarias en un sistema autónomo de verificación *off-line* de firmas: el aprendizaje, la segmentación y la verificación. En este capítulo se presentan diversos métodos que, unidos al método de segmentación propuesto en el capítulo anterior, cubren estos tres procesos desde diferentes perspectivas.

### 4.1 Generación automática de muestra sintética

Ciertos tipos de sistemas de clasificación, como los clasificadores estadísticos y las Redes de Neuronas, precisan varios ejemplares de muestra para realizar un proceso de entrenamiento que ajuste sus parámetros. Una alternativa, desarrollada por varios autores [117][20], para evitar el requisito de disponer de un elevado número de firmas de aprendizaje por cada individuo, consiste en generar de manera automática varias firmas sintéticas utilizando una muestra reducida capturada de cada individuo.

En esta sección se detalla una propuesta de algoritmo de generación de muestra sintética y la base experimental sobre la que se sustenta. Cabe destacar que el método que se presenta es realista en cuanto a que sólo precisa de una firma original para su operación. Gracias este proceso de generación de muestra sintética, los algoritmos de verificación, que se expondrán posteriormente en este capítulo, podrán disponer de la muestra necesaria para sus procesos de aprendizaje.

La generación de firmas que se realiza en este trabajo se basa en diversas transformaciones que tratan de emular la variabilidad intrapersonal inherente a cada firmante. Mediante el análisis de la muestra A se han detectado diferentes factores de variabilidad: pequeños giros, alteraciones en los extremos de los trazos o en los extremos de la firma y cambios en el grosor del trazo.

### 4.1.1 Factores de variabilidad y transformaciones equivalentes

En los siguientes puntos se analizan diferentes variaciones, observadas sobre la muestra A, y se proponen diversos algoritmos que tratan de simularlas.

#### Giros

Como se puede apreciar en la Figura 4.1, el ángulo que forma una firma respecto al soporte en el que se firma suele cambiar ligeramente. Esta oscilación puede tener su origen en la posición del soporte o en la postura del firmante.

Utilizando el eje de mínima inercia como dirección de referencia se ha medido la inclinación de las firmas de la muestra A. Los resultados obtenidos, que se presentan en la Figura 4.2, arrojan una variación media en valor absoluto de  $2.84^\circ$  por firmante, con una desviación típica de  $1.54^\circ$ . Esta variabilidad se puede generalizar a cambios entre 0 y 10 grados, lo que ha motivado que no se consideren giros más allá de  $\pm 5^\circ$ . Así, para simular esta oscilación, se realizan transformaciones de  $-5^\circ$ ,  $-2.5^\circ$ ,  $2.5^\circ$  y  $5^\circ$  sobre cada firma modelo.

#### Modificación de los extremos

Se ha observado en la muestra A que, de una firma a otra de un mismo individuo, los extremos de los trazos se alargan o se acortan (ver ejemplo de la Figura 4.3). Con objeto de simular esta variabilidad, se ha diseñado un proceso que determina los puntos extremos de una firma y su dirección aparente, apoyándose en el esqueletizado de la imagen, para después prolongar o acortar sus trazos extremos.

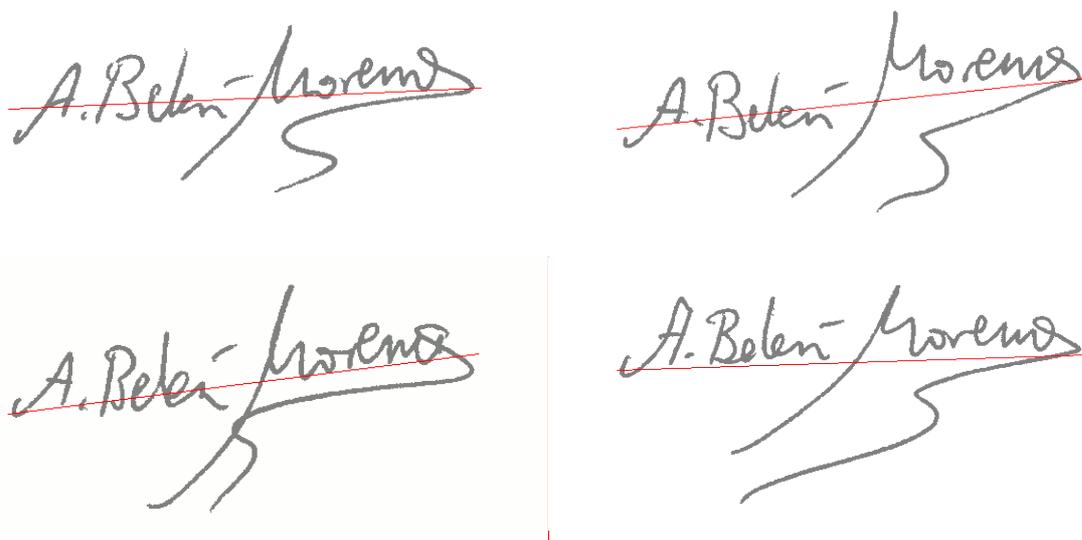


Figura 4.1.- Posición del eje de mínima inercia sobre diferentes firmas de un mismo individuo.

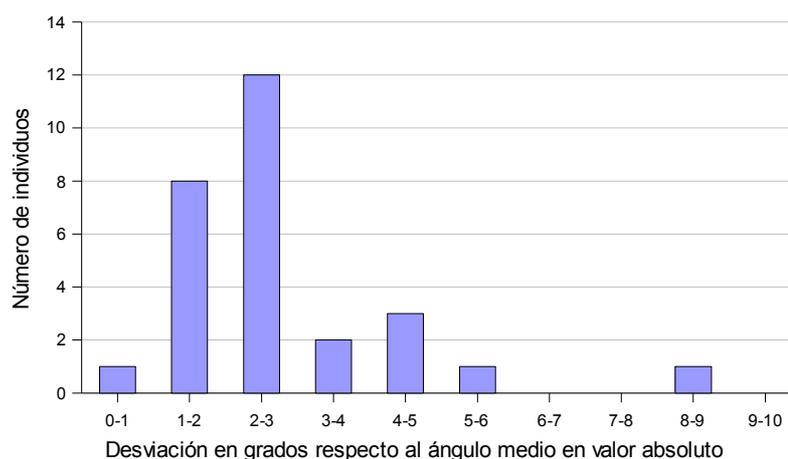


Figura 4.2.- Histograma de variación del ángulo para la muestra A.

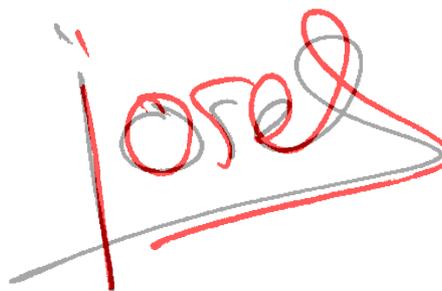


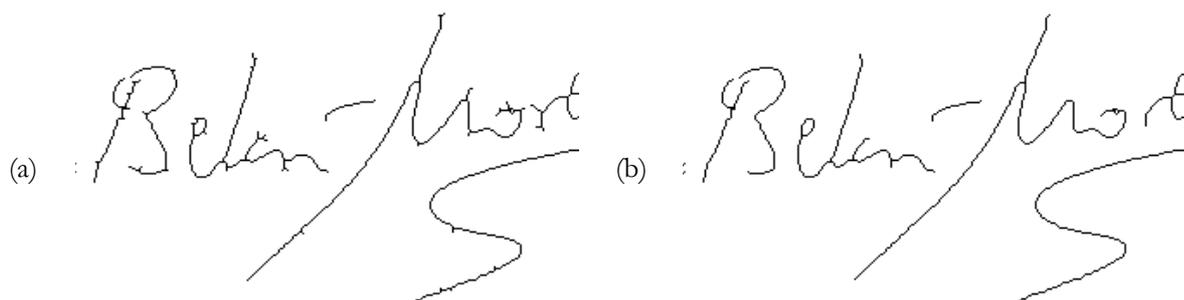
Figura 4.3.- Ejemplo que muestra la variación en la longitud de los extremos de los trazos. Se comparan dos firmas de un individuo de la muestra A.

Respecto al proceso de *esqueletizado* cabe decir que se ha realizado una adaptación que, sobre imágenes de firmas, mejora los resultados de los algoritmos de Zhang y Suen [172]. Esta modificación se apoya en el hecho de que la imagen a esqueletizar corresponde a una firma. Este hecho permite estimar el *grosor medio del trazo* mediante el cociente entre el número de píxeles de la imagen y el número de píxeles de la imagen esqueletizada.

$$G = \frac{P(I)}{P(Esq(I))} \quad (4.1)$$

donde  $G$  es el grosor estimado,  $Esq$  es la función de esqueletizado,  $P$  es una función que cuenta el número de píxeles, e  $I$  es la imagen de una firma.

Esta estimación del grosor del trazo se utiliza como umbral para eliminar los pequeños segmentos que suelen aparecer adosados al esqueleto según el Algoritmo 4.1. El resultado, como se puede apreciar en la Figura 4.4, es un esqueletizado de mayor calidad.



**Figura 4.4.-** Diferencias en los algoritmos de esqueletizado: (a) utilizando el esqueletizado de Zhang y Suen, y (b) utilizando la modificación propuesta.

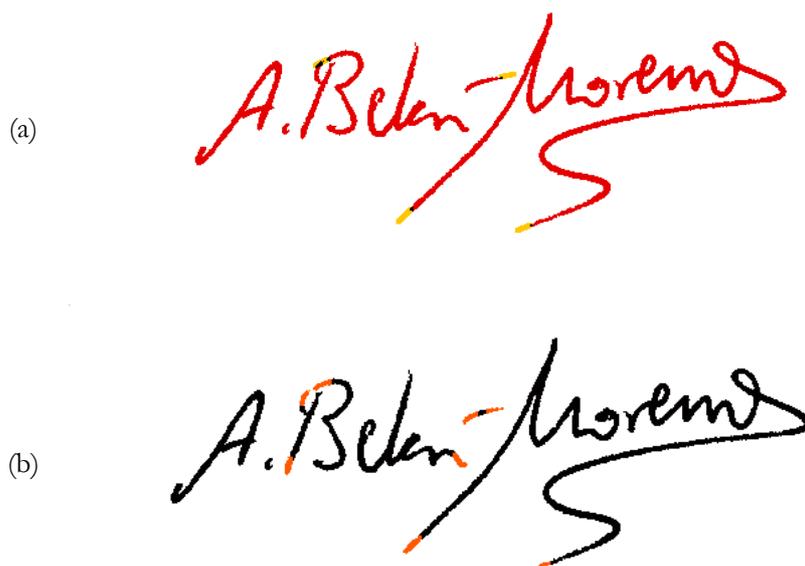
<b>Algoritmo 4.1 de mejora al proceso de esqueletizado</b>	
<b>Nombre:</b>	EsqueletizarFirma
<b>Entradas:</b>	$I_s$ : Imagen esqueletizada con pequeños adosados G : Grosor estimado para el trazo
<b>Salida:</b>	$I_L$ : Imagen esqueletizada limpia de tales segmentos
1 -	$P = \text{BuscarPíxelesConUnSoloVecino}(I_s)$
2 -	<b>for each</b> (p in P)
3 -	segmento = []
4 -	$I_{aux} = I_s$
5 -	salir = false
6 -	<b>do</b>
7 -	Eliminar(p, $I_{aux}$ )
8 -	<b>if</b> (Vecinos(p, $I_{aux}$ ) == 1)
9 -	p = Vecino(p, $I_{aux}$ )
10 -	segmento.insert(p)
11 -	<b>else</b>
12 -	salir = true
13 -	<b>while</b> (!salir)
14 -	<b>if</b> (segmento.size() < G)
15 -	Eliminar(s, $I_s$ )
16 -	$I_L = I_s$

Una vez esqueletizada, se procede a la detección de los extremos de los trazos de la firma. Para encontrarlos se buscan sobre el esqueleto aquellos puntos que sólo tienen un vecino y que forman

parte de un segmento cuya longitud sea superior a un mínimo. Tras diversas pruebas este mínimo se ha fijado en medio centímetro. Esto hace que el algoritmo de detección de extremos quede como una variación del Algoritmo 4.1 en la que se usa para  $G$  un valor igual a medio centímetro en lugar de usar la estimación del grosor del trazo.

Una vez encontrados los extremos de los trazos ya se puede abordar el problema de su acortamiento o alargamiento. El proceso que acorta los trazos es sencillo, consiste en eliminar los píxeles correspondientes al final de los trazos detectados. Otra vez, este algoritmo es una variación del 4.1. La única diferencia consiste en que, en vez de eliminar los píxeles sobre el esqueleto, se realiza una erosión de tales píxeles con un elemento estructurante de tamaño igual al doble del grosor del trazo estimado según la fórmula 4.1.

Por otro lado, el alargamiento de los extremos es un proceso más complejo. Para empezar se necesita conocer la dirección en la que el extremo debe crecer. Para ello, se utiliza la recta de regresión de los puntos que forman el extremo. Si el extremo en cuestión está en curva, la prolongación en la dirección marcada por la recta de regresión no sería adecuada. Para detectar este caso, se calcula el error cuadrático resultante de comparar los puntos finales del extremo y la recta de regresión. Si su valor no es próximo a cero se supone que el extremo está en curva y se descarta su prolongación. En otro caso, se alarga en la dirección indicada utilizando un elemento estructurante de dilatación cuadrado de tamaño  $G$ . La Figura 4.5 presenta un ejemplo de la aplicación de este proceso sobre la imagen de una firma de la muestra A.



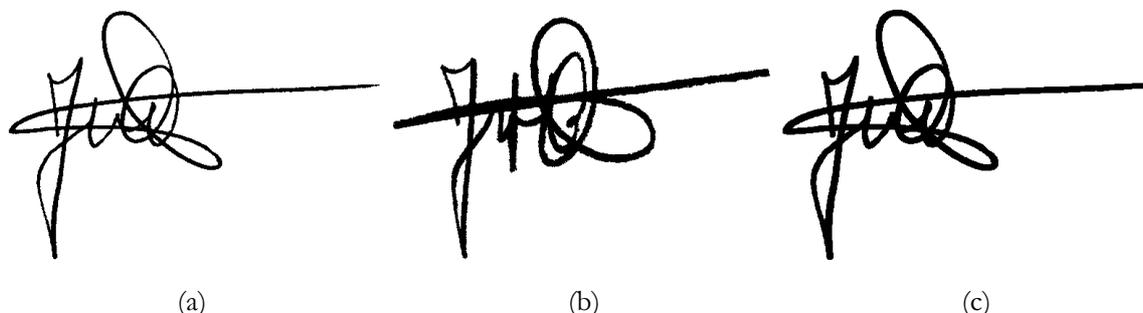
**Figura 4.5.-** Ejemplo de modificación de los extremos de los trazos. En (a) se alargan los extremos mediante el trazo marcado en amarillo. En (b) se acortan los extremos eliminando los tramos coloreados de rojo.

### Modificación del grosor

Otra variación que se puede apreciar al comparar las firmas consiste en los diferentes grosores de trazo (ver Figura 4.6). Estas diferencias tienen su origen en el útil de escritura usado para firmar. Así, los trazos más finos corresponden a bolígrafos de punta fina, mientras que los más gruesos

corresponden a *rollers* o rotuladores. Así, sobre las imágenes digitalizadas, el grosor de los trazos suele oscilar entre los 5 y los 10 píxeles a 300 DPIs.

Para introducir esta variabilidad en la muestra sintética se realiza un proceso morfológico, utilizando un elemento estructurante fijo, que modifique el grosor estimado según la ecuación 4.1. Si, a 300 DPIs, el grosor estimado es inferior a 7 píxeles se realiza una dilatación del trazo y si es superior, una erosión.



**Figura 4.6.-** Ejemplares que muestran diferencias de grosor y su simulación mediante dilatación. Las imágenes (a) y (b) corresponden a dos firmas de un mismo individuo usando diferentes útiles de escritura. La imagen (c) es a una dilatación de la imagen (a).

### Elongaciones iniciales o finales

Otra diferencia que puede apreciarse en las firmas de un individuo son las elongaciones que suele realizar al inicio o al final de su firma. Por ejemplo, en la Figura 4.7 se aprecia este efecto sobre dos ejemplares, representados en diferentes colores, proporcionados por un individuo de la muestra A. En este caso se aprecia que la figura que inicia la firma por la izquierda, una especie de alfa invertida, es más alargada en el ejemplar superior, mientras que la longitud total de la firma aparece aproximadamente igual.

Para reproducir este efecto se ha realizado una transformación que altera el inicio o el final de la firma. Si el ancho de la imagen se normaliza entre 0 y 1, estas alteraciones, inicial y final, se realizan respectivamente mediante las siguientes funciones definidas por trozos.

$$I_{inicio}(x, y) = \begin{cases} I\left(\frac{x}{2}, y\right) & \text{si } x < 0.2 \\ I\left(\frac{9x-1}{8}, y\right) & \text{si } x \geq 0.2 \end{cases} \quad I_{fin}(x, y) = \begin{cases} I\left(\frac{9x}{8}, y\right) & \text{si } x < 0.8 \\ I\left(\frac{x+1}{2}, y\right) & \text{si } x \geq 0.8 \end{cases} \quad (4.2)$$

donde  $I$  es una matriz correspondiente a la imagen original.

La Figura 4.8 presenta el resultado de la aplicación de las ecuaciones 4.2 sobre una firma de ejemplo. La curva (a) de la Figura 4.8 es la función identidad, por lo que la imagen corresponde a la firma original sin alteración alguna. La curva (b) tiene una pendiente menor en el 20% inicial de su recorrido, esto se traduce en una elongación de la parte izquierda del trazo. Finalmente, la curva (c) disminuye su pendiente en el 20% final de su recorrido, lo que elonga la parte derecha del trazo.



Figura 4.7.- Ejemplo que manifiesta elongación inicial y final de la firma, extraído de la muestra A

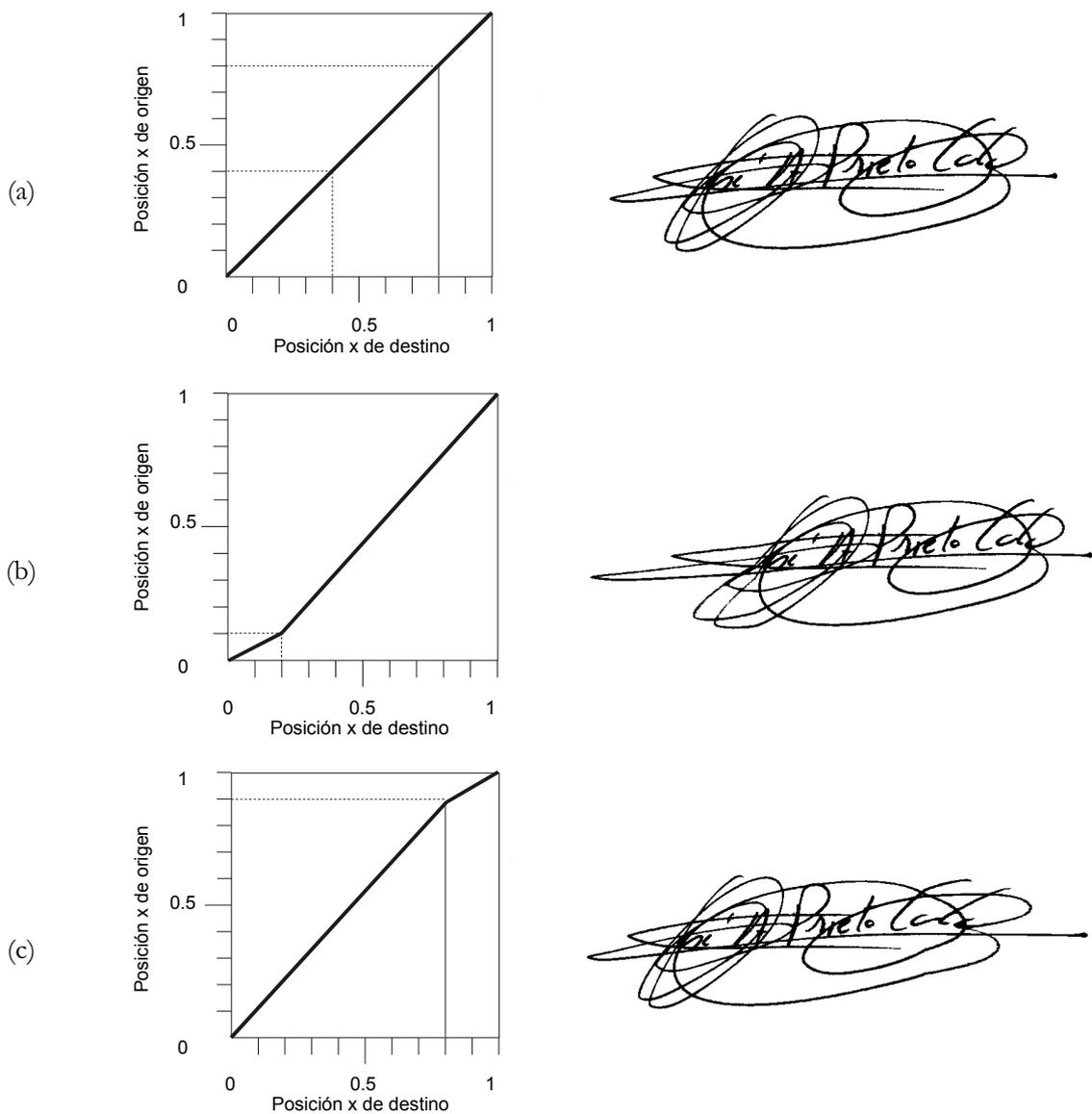


Figura 4.8.- Elongación de los extremos de una firma y función de transformación utilizada: (a) firma original, (b) elongación del inicio y (c) elongación del final.

### 4.1.2 Resultado de la generación sintética de firmas

La Figura 4.9 recoge un resumen del proceso de generación de muestra sintética. Se debe señalar que estos procesos se han ordenado con objeto de reducir el tiempo de ejecución. Así, los procesos que requieren el esqueletizado (como la modificación de extremos o la dilatación) se realizan en primer lugar. A continuación, se realizan los procesos de elongación y finalmente los de giro.

Este conjunto de transformaciones da lugar a  $3 \times 2 \times 3 \times 5 = 90$  firmas sintéticas por cada firma original. Los resultados de este proceso se evaluarán mediante el éxito o fracaso de los métodos de verificación que los utilizan y que se exponen a continuación.

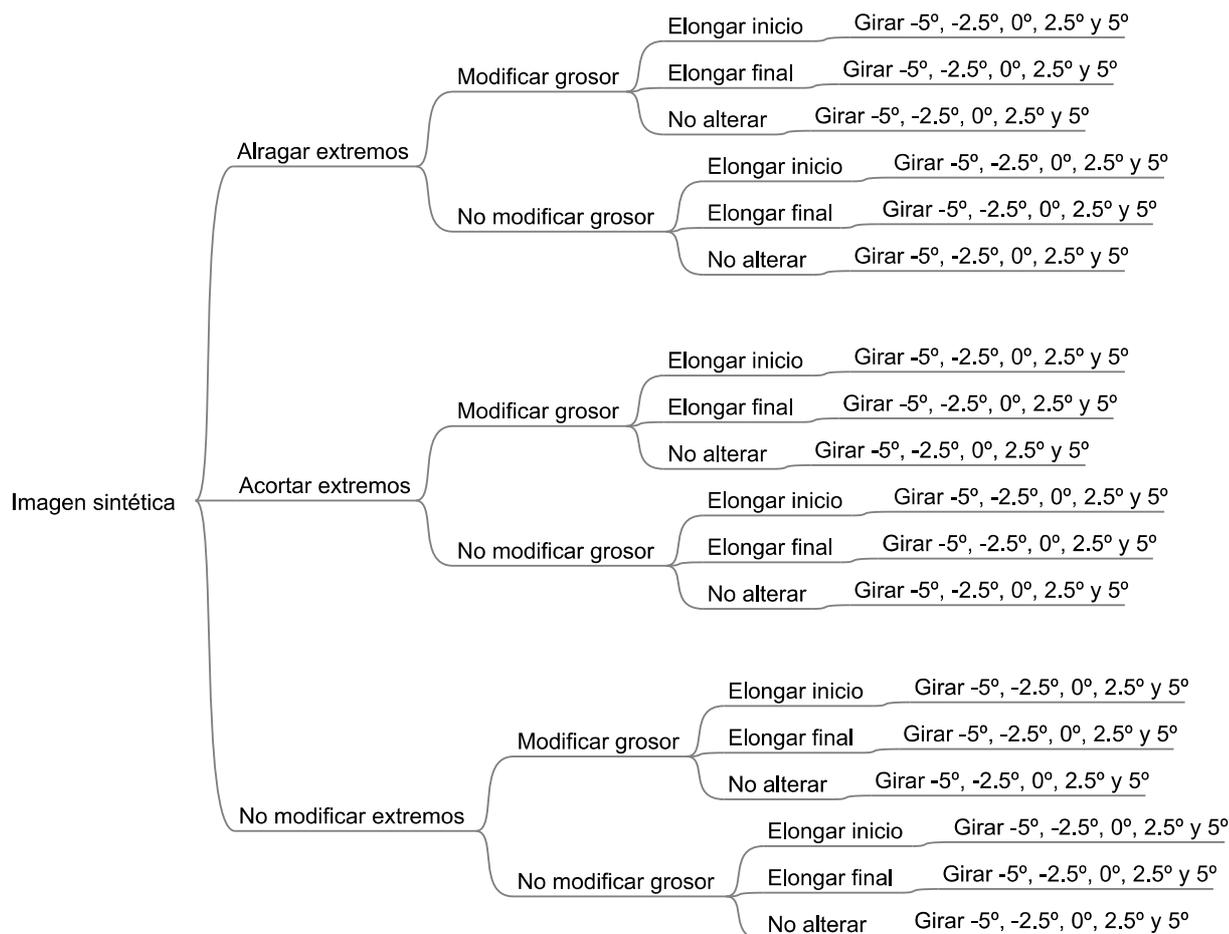
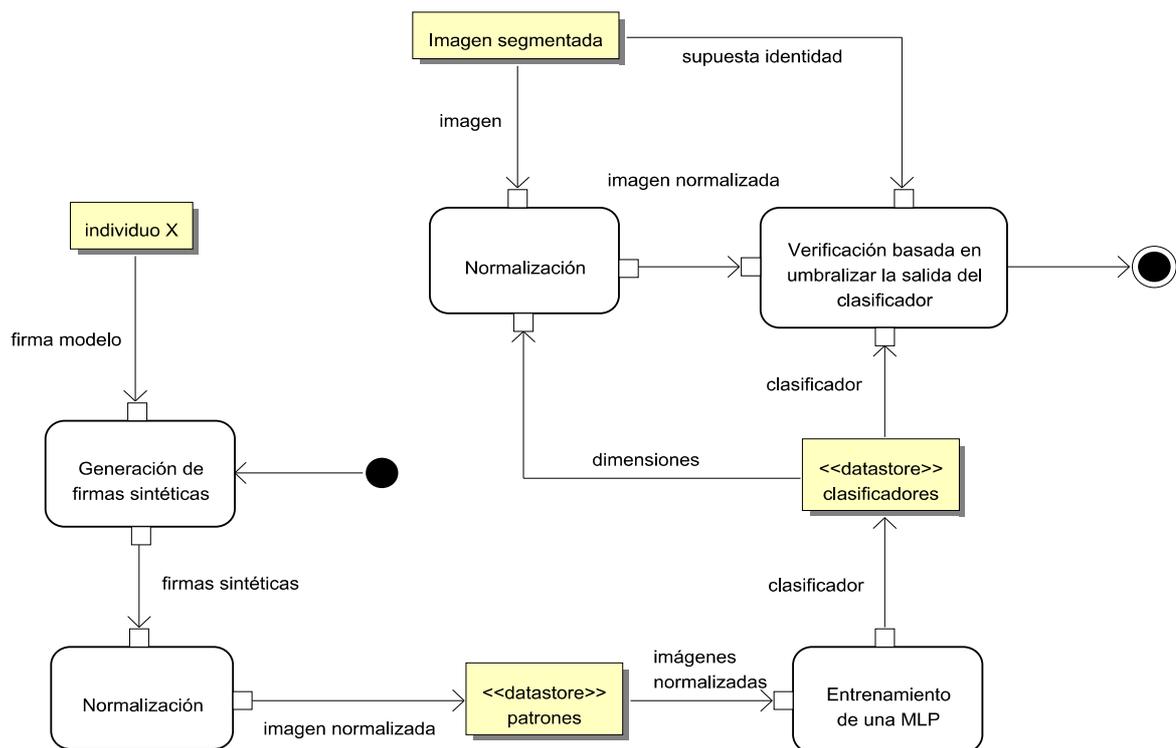


Figura 4.9.- Árbol de transformaciones que da origen a la muestra sintética.

## 4.2 Verificación de firmas utilizando una Red de Neuronas

En esta sección se expone un método de verificación de firmas que se basa en dos elementos: la generación automática de firmas sintéticas y la clasificación entre firmas genuinas y falsificaciones mediante un *MLP* (Perceptrón Multicapa) a cuya entrada se presenta directamente una imagen de la firma normalizada. Una descripción de alto nivel del método puede verse en el Diagrama de Actividad de la Figura 4.10.



**Figura 4.10.-** Diagrama de Actividad para la verificación utilizando una Red de Neuronas *MLP*.

Este método se engloba dentro de la categoría de enfoques basados en clasificadores novedosos comentado en el apartado 2.3.1 y encuentra referencias en los trabajos [57], [128] y [173]. La principal diferencia con estos trabajos, que también utilizan un *MLP* como único mecanismo de verificación, estriba en que la muestra utilizada para el entrenamiento procede de un proceso de generación sintética a partir de una única firma de referencia, mientras que en aquellos trabajos se utilizaba abundante muestra real obtenida de cada individuo.

El método que se propone utiliza el enfoque *OCON* (*one-class-one-network*) propuesto por Baltzaquis y Papamarkos en 2001 [106]. De esta manera se entrena una Red de Neuronas por cada una de las posibles firmas a verificar. La ventaja de este enfoque está en la escalabilidad, pues para añadir un nuevo individuo solo se debe entrenar un nuevo clasificador manteniendo intacto el resto. Su inconveniente, respecto a un sistema que solo utilizase un clasificador, lo encontramos en el aumento de recursos de almacenamiento.

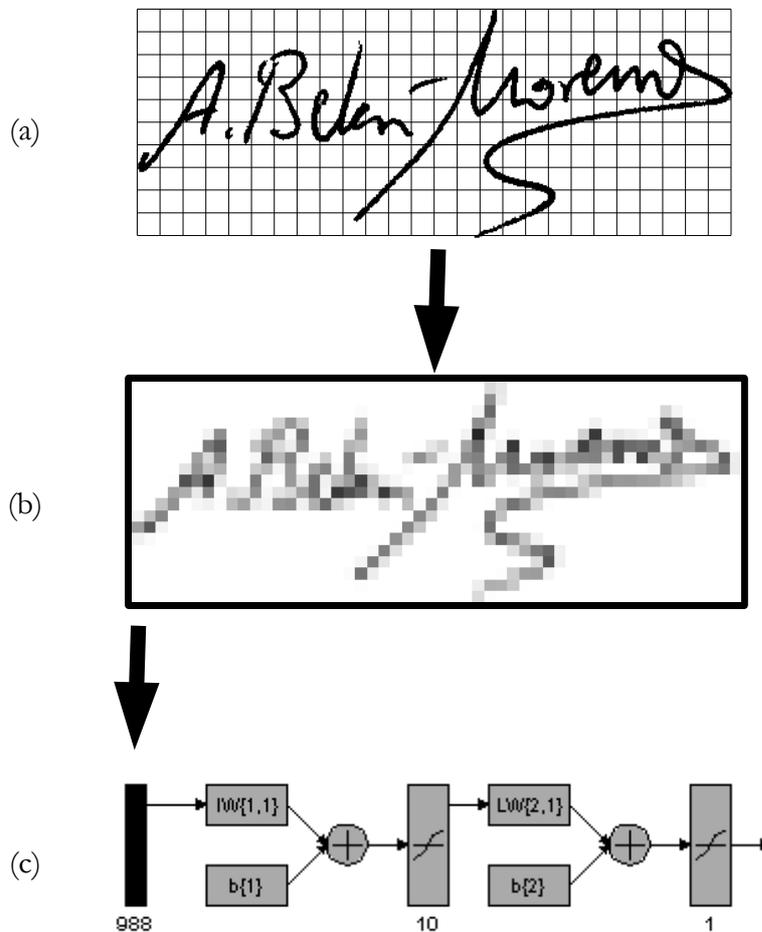
#### 4.2.1 Detalle del proceso de verificación basado en la imagen

Como se ha señalado este proceso de verificación sólo utiliza la propia imagen de la firma como vector de características. Así, como el clasificador utilizado es una Red de Neuronas *MLP*, los píxeles de la imagen corresponden a la entrada de la red. Con objeto de no ralentizar innecesariamente el entrenamiento y el funcionamiento de la red, las imágenes se han normalizado a una resolución menor utilizando interpolación bilineal.

Como se utiliza el enfoque *OCON* cada Red de Neuronas, una por individuo, puede tener a la entrada un número de neuronas diferente. Esto permite ajustar la entrada de cada red a las proporciones de la firma a verificar, manteniendo las proporciones originales de la firma modelo en la normalización de las firmas a verificar. De esta manera se pretende minimizar la pérdida de

información que este proceso de normalización pueda causar. Así, para crear la Red de Neuronas de un individuo, las neuronas de entrada se disponen en un rectángulo que tiene las mismas proporciones que la firma modelo y con un número de neuronas aproximadamente igual a 1000. Por ejemplo, la imagen (a) de la Figura 4.11 que tiene unas dimensiones de 489X185 se reescala a la imagen (b) de tamaño 52X19 utilizando interpolación bilineal. Los valores de los puntos de esta imagen reescalada serán la entrada a la Red de Neuronas (c) de la Figura 4.11.

La configuración de la salida de la red es muy sencilla pues sólo necesita una neurona de salida, ya que el sistema se puede entrenar para que devuelva 1 si la firma debe aceptarse y -1 si debe rechazarse. Por otro lado, la configuración de los patrones de entrenamiento es más complicada. El conjunto de entrenamiento para el caso en el que la firma debe aceptarse es más sencillo de determinar que el conjunto para el caso en el que la firma debe rechazarse. Esto se debe a que este segundo caso es mucho más abierto. Más en detalle, para entrenar la clase de las firmas a aceptar se utiliza la muestra sintética, que trata de reproducir la variabilidad intrapersonal. Sin embargo, el conjunto para entrenar la clase de las firmas a rechazar debería contemplar todas las falsificaciones posibles. En este caso se ha optado, con resultados positivos, por utilizar todos los ejemplares de la muestra A a modo de ejemplos a rechazar.



**Figura 4.11.-** Representación del proceso de verificación usando la imagen y un Perceptrón.

Utilizando la muestra A se han experimentado diferentes posibilidades para las capas ocultas. En particular, se ha probado a no poner capas ocultas, a poner una y a poner dos. También se ha probado a poner 5, 10 y 20 neuronas en las capas ocultas. Se ha observado que el uso de más de una capa oculta, o que ésta disponga de más de 10 neuronas no repercute en el resultado obtenido, por ello se ha elegido esta configuración para todos los casos. En la Figura 4.11 (c), que presenta dicha configuración, se puede apreciar la entrada (compuesta de 988 neuronas) la capa oculta (de 10 neuronas) y la capa de salida (con 1 neurona). Obsérvese que en la figura también se representan las matrices correspondientes a los pesos de las conexiones entre neuronas ( $W_{1,1}$ ) y los términos constantes o *bias* ( $b_{1,1}$ ).

También ha sido necesario automatizar el proceso de aprendizaje de las diferentes Redes de Neuronas. Idealmente se podría utilizar una muestra de validación para detener el proceso de aprendizaje en el punto óptimo de generalización. Sin embargo, la escasez de la muestra y el deseo de utilizar toda la muestra sintética en el aprendizaje nos ha conducido a una simplificación de este proceso. Así, tras varios ensayos sobre la muestra A se ha determinado que el proceso de entrenamiento se detiene si el error cuadrático medio durante el entrenamiento alcanza el valor 0.01 o si se alcanzan los 3000 ciclos de entrenamiento.

#### 4.2.2 Experimentos sobre la muestra de test

La primera prueba ha consistido en entrenar 25 Redes de Neuronas correspondientes a cada uno de los 25 individuos de la muestra B. Para aprender la clase de las firmas genuinas de cada red se utiliza la muestra sintética generada a partir del primer ejemplar del individuo cuya identidad se desea verificar. Para entrenar la clase de los rechazos se utiliza siempre la muestra A.

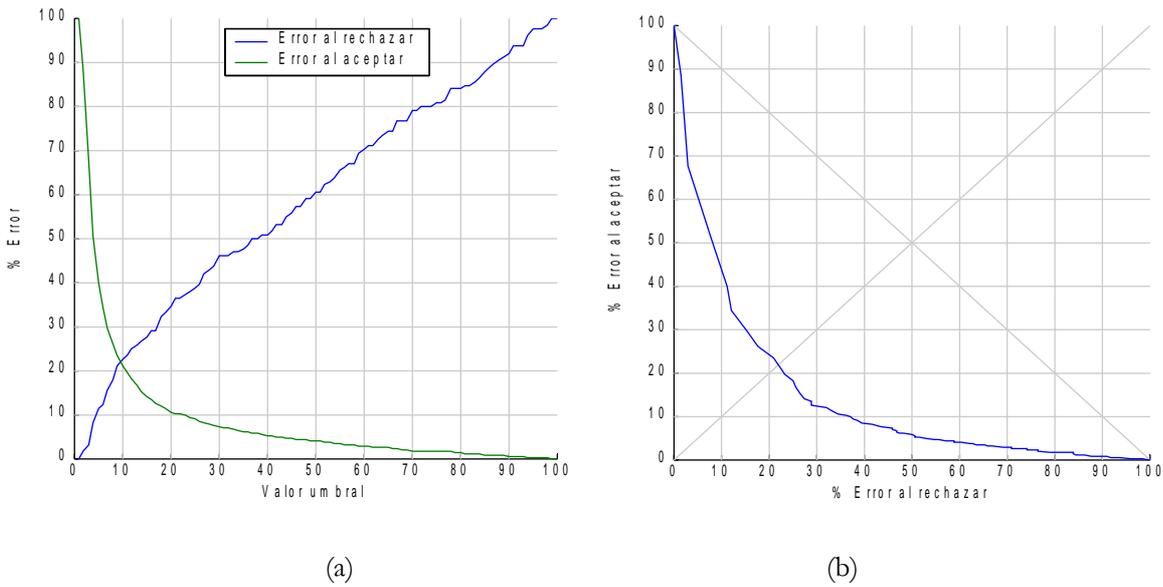
Tras el entrenamiento se confeccionan las curvas *FRR* y *FAR* de la Figura 4.12. Para obtener la curva *FRR* se utilizan los 5 ejemplares restantes de cada uno de los individuos de la muestra B los cuales no han sido utilizados hasta ahora. Para obtener la curva *FAR* se utilizan todos los ejemplares de la muestra B excepto los del individuo que se debe aceptar (esto es  $24 \times 6 = 144$  ejemplares). Obsérvese que la clase de los rechazos se ha entrenado con las firmas de la muestra A y el test se realiza con las firmas de la muestra B, la cual corresponde a individuos diferentes.

En la curvas de la Figura 4.12 se aprecia un *EER* de 22.4%. Estos resultados son similares a los presentados por Rash [173] para las falsificaciones de tipo *casual* que fueron del 20%. Por lo que se puede deducir que en este caso el sistema de generación de muestra sintética ha cumplido su objetivo y ha capturado la variabilidad intrapersonal que influye en este tipo de sistemas de verificación.

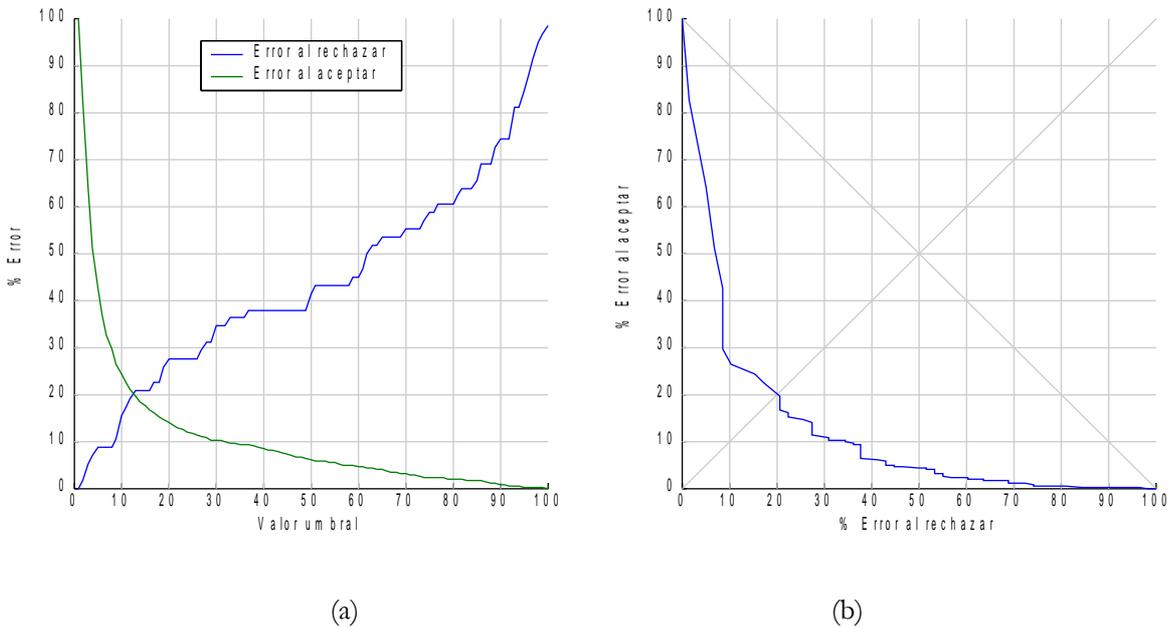
La segunda prueba ha consistido en entrenar 16 Redes de Neuronas, una para cada uno de los 16 individuos de la muestra E. De nuevo, para aprender la clase correspondiente a las firmas genuinas, se utiliza la muestra sintética generada a partir del primer ejemplar del individuo cuya identidad se desea verificar. Para entrenar la clase de los rechazos se utiliza la muestra A.

Tras el entrenamiento se confeccionan las curvas *FRR* y *FAR* de la Figura 4.13. Para obtener la curva *FRR* se utiliza el resto de ejemplares del individuo a verificar una vez excluido aquél que se utilizó para generar la muestra sintética y por tanto para entrenar. Para obtener la curva *FAR* se utilizan todos los ejemplares de la muestra E excepto los del individuo que se debe aceptar.

El *ERR* del 20% obtenido significa una mejora resultados obtenidos para la muestra B, aunque esta diferencia no puede considerarse significativa. Esto parece indicar que el sistema se comporta igual sobre firmas limpias (muestra B) que sobre firmas ruidosas obtenidas en condiciones reales (muestra E).



**Figura 4.12.-** Curvas de error obtenidas al usar una Red de Neuronas sobre la muestra B: (a) presenta el error al aceptar ( $FAR$ ) y al rechazar ( $FRR$ ) y (b) es una curva ROC.



**Figura 4.13.-** Curvas de error obtenidas al usar una Red de Neuronas sobre la muestra E: (a) presenta el error al aceptar ( $FAR$ ) y al rechazar ( $FRR$ ) y (b) es una curva ROC.

### 4.3 Verificación basada en características discriminantes

En esta sección se expone un método de verificación de firmas que se basa en tres elementos: la generación automática de firmas sintéticas (ver subsección 4.1.2), una recopilación de las principales

características discriminantes geométricas propuestas en trabajos clásicos y una etapa de clasificación basada en un Perceptrón Multicapa siguiendo el enfoque *OCON*. Una descripción de alto nivel del método puede verse en el Diagrama de Actividad de la Figura 4.14.

El enfoque que aquí se plantea está basado en un conjunto de características ya documentadas como significativas y discriminantes por otros autores. Por ello, lógicamente, el uso de estas características discriminantes no es algo novedoso. Sin embargo, su uso sobre una muestra generada sintéticamente no aparece documentada.

Ya se enumeraron en el apartado 2.3.1 las principales características que habitualmente se utilizan en relación con la verificación de firmas. Evidentemente, dado la gran cantidad de trabajos existentes, ha sido necesario realizar una selección. Así, nos hemos centrado en las características globales más sencillas. Se trata de características que individualmente no aportan demasiada información discriminante, pero que en conjunto veremos que permiten abordar el problema de verificación satisfactoriamente.

### 4.3.1 Descripción de las características discriminantes

Se ha construido un vector de 37 características discriminantes (ver Tabla 4.1). La mayoría de las características se basan en el estudio geométrico de la imagen de la firma o de los histogramas de las proyecciones sobre los ejes de dicha imagen. En los siguientes apartados se describe con detalle en cada una de ellas y cómo se ha implementado su obtención automática.

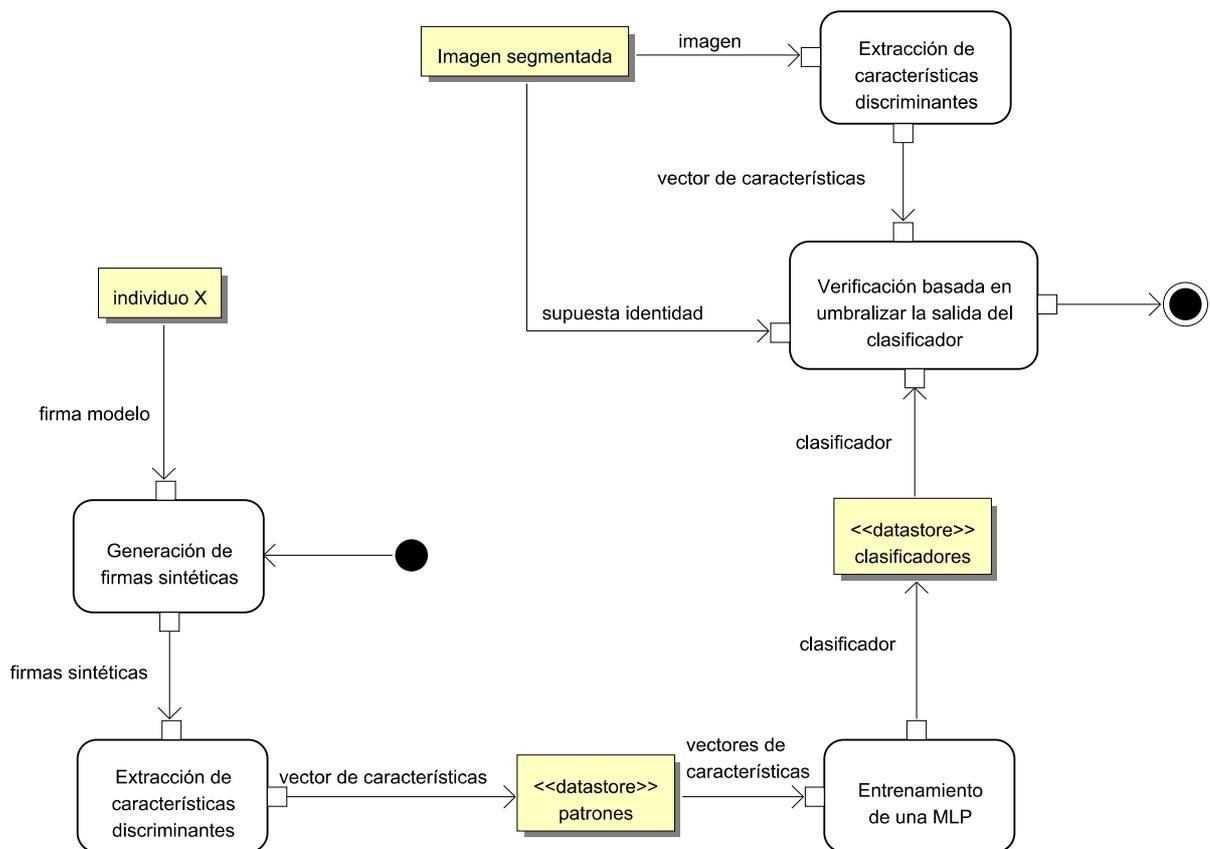


Figura 4.14.- Diagrama de Actividad UML para el método basado en características globales.

Título del conjunto de características	Número de características
Proporciones del rectángulo que circunscribe a la firma	1
Pendiente y posición de la recta de mínima inercia y del eje mayor del cierre convexo	$2 \times 2 = 4$
Número de componentes 4 y 8 conexas	2
Número de regiones cerradas (bucles)	1
Número de intersecciones entre los trazos de la firma	1
Área del cierre convexo, del relleno respecto al eje de mínima inercia y de la firma esqueletizada respecto al rectángulo que la contiene	3
Índice de redondez o compacidad	1
Centro de masa global y centros de masa parciales	$2 + 2 \times 4 = 10$
Línea base global y medidas superiores e inferiores respecto al eje X e Y	$3 \times 2 = 6$
Diferentes relaciones entre la medida de la curtosis y la de asimetría	8

Tabla 4.1.- Características discriminantes implementadas.

### Proporciones del rectángulo que circunscribe a la firma

Una característica sencilla de obtener consiste en el cociente entre el alto y el ancho del rectángulo mínimo que contiene completamente la firma y que tiene lados paralelos a los bordes de la imagen según se ha capturado (*bounding box*) [23][2][106].

La Figura 4.15 presenta la frecuencia de las desviaciones respecto a la media de las proporciones observada en la muestra A. Dicho histograma indica que las proporciones de estos rectángulos para un individuo tienen una variabilidad media de aproximadamente un 30%. Hay que señalar que las variaciones de tamaño observadas no se debían a un cambio en la escala de la firma, pues las zonas de firmado tenían siempre el mismo tamaño, sino que se debían a que se alargaban o se acortaban los trazos, haciendo que variase el tamaño del rectángulo que circunscribe a cada firma.

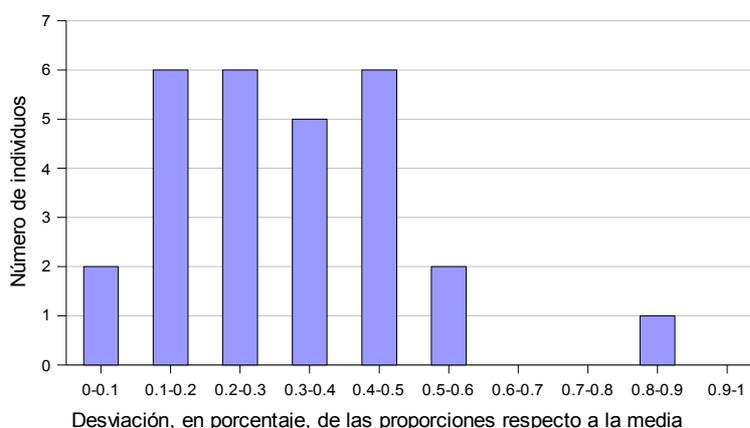


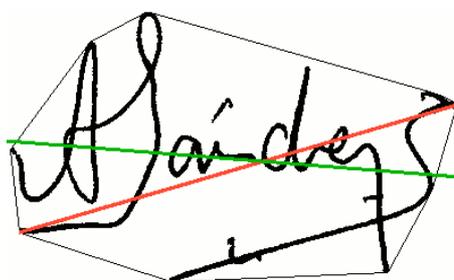
Figura 4.15.- Histograma sobre la variación de las proporciones de los rectángulos mínimos que contienen a las firmas.

### Pendientes de la recta de mínima inercia y del eje mayor del cierre convexo

En esta sección se presentan dos medidas la inclinación y orientación habitual de la firma [52]. Así, en una búsqueda de un vector estable dentro del grafismo se opta por dos alternativas: el eje de mínima inercia [87] y el eje mayor del polígono resultante del cierre convexo del grafismo (ver Figura 4.16).

De cada una de estas medidas se obtienen 2 características discriminantes: el ángulo y el corte con el eje Y. Esta última se normaliza respecto al alto del mínimo rectángulo horizontal que contiene la firma con objeto de hacerla invariante a escala.

Además de usar estos ángulos como características discriminantes, algunos autores las utilizan en una etapa de preproceso mediante rotación [87] previo al cálculo del resto de características. Sin embargo, otros autores prefieren no realizar dicha normalización debido a la alta influencia que podría tener sobre el cálculo del resto de características en caso de error y a que la orientación de las firmas de un mismo individuo suele ser bastante estable y puede usarse como característica discriminante [83].



**Figura 4.16.-** Ejemplo de envoltura convexa y de posición de los ejes. En rojo el mayor eje del polígono y en verde el eje de mínima inercia.

### Número de componentes conexas

El uso del número de componentes conexas como aproximación al número de elementos constituyentes de la firma es una característica discriminante utilizada en varios trabajos [55][94]. Como se puede hablar de agrupaciones 4 u 8 conexas, dependiendo de si se sigue el criterio de 4 conexión o de 8 conexión, se ha optado por crear dos características una por cada criterio.

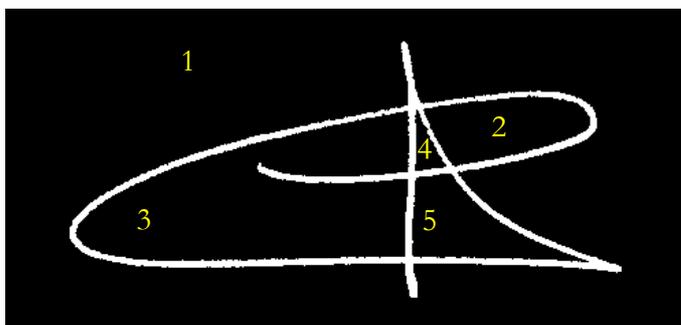
Ya se ha justificado en la sección 3.4.2 la estabilidad de esta característica. Se puede añadir que el análisis de la muestra A revela que su variabilidad intrapersonal radica en dos factores: falta de resolución en la digitalización y pequeñas alteraciones en la posición de los elementos constituyentes de la firma. La falta de resolución motiva que en la imagen aparezcan unidos trazos que en el documento original están separados. Por otro lado, cuando se alarga un trazo, como el de la 'S' final de la Figura 4.3, se pueden unir objetos que en otros ejemplares están separados.

### Número de regiones cerradas (bucles) encontradas en la imagen de la firma

El número de regiones encerradas por el trazo de la firma es una característica que aparece en los trabajos de Baltzakis y Papamarkos [106][71]. Este número constituye una cota superior del número de bucles realizados al trazar la firma, pues además incluye los bucles resultantes de las intersecciones con otros trazos anteriores.

El algoritmo utilizado para detectar estos bucles es sencillo. Consiste en añadir un margen a la imagen y, tras esto, invertirla. El número de componentes conexas que se encuentre corresponderá con el número de regiones cerradas más una, correspondiente al fondo de la imagen (ver Figura 4.17).

La estabilidad de esta característica es alta, pues su variabilidad suele deberse al cambio de posición o longitud de algún trazo largo que motiva nuevas intersecciones y por tanto nuevos bucles.



**Figura 4.17.-** Ejemplo de detección de bucles formados por los trazos usando el inverso de la imagen. El número de componentes conexas se corresponde al número de bucles más uno.

### Número de intersecciones entre los trazos de la firma

Esta característica trata de medir el número de intersecciones producidas por los trazos que componen la firma [71][106]. La localización de tales intersecciones se realiza sobre la imagen esqueletizada de la firma mediante un algoritmo de dos fases: una primera de etiquetado y una segunda de eliminación de redundancia.

En la primera fase se etiquetan como candidatos a *punto de cruce* aquellos puntos de la imagen cuyo entorno coincide con alguno de los 16 patrones de la Figura 4.18. En la segunda fase se agrupan aquéllos que se encuentren a una distancia inferior a un umbral prefijado  $U$ . Para el caso de las firmas se ha tomado el umbral  $U$  igual al doble del grosor medio estimado para las firmas (ver punto de modificación de los extremos en 4.1.1). La Figura 4.19 presenta el resultado de esta operación sobre una firma de la muestra A. Obsérvese que es posible la aparición de falsos positivos, como por ejemplo, el que aparece en el extremos derecho de la curva que cierra la firma por abajo.

### Áreas del esqueleto, del cierre convexo, del relleno respecto al eje de mínima inercia

Las medidas de área, entendida como el recuento del número de píxeles sobre la imagen, es una característica común en muchos trabajos [58][23][2][106][18]. Respecto a qué área se debe medir, la literatura propone diferentes opciones, de las cuales en este trabajo recogemos tres:

- El área del cierre convexo [171] de la figura de la firma (ver Figura 4.20 (a)).
- El área de la proyección vertical sobre el eje de mínima inercia de los píxeles de la firma (ver Figura 4.20 (b)).
- El área del trazo de la firma. En este caso, para evitar dependencias con el grosor del útil de escritura, se cuenta el número de píxeles sobre la imagen esqueletizada.

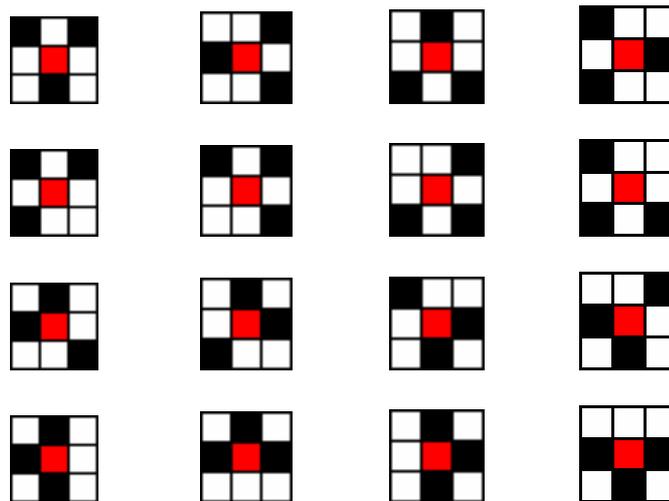


Figura 4.18.- Patrones utilizados para detectar los puntos de cruce entre los trazos de una firma.

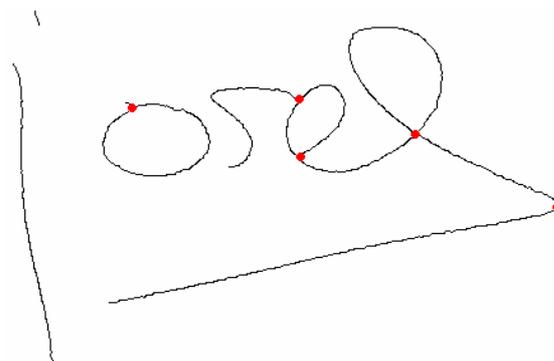


Figura 4.19.- Puntos de cruce entre trazos sobre la imagen esqueletizada de una firma de la muestra A.

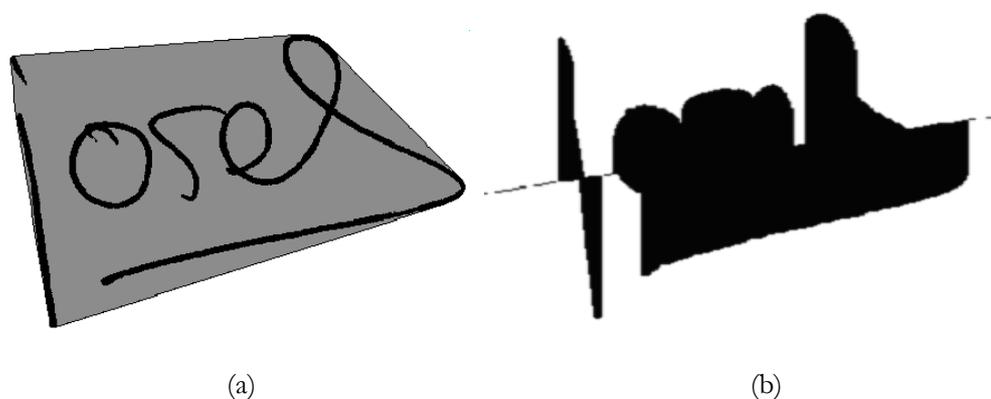
En cualquiera de los tres casos, como el área se calcula por recuento de píxeles, no es una medida independiente a cambios de escala. Es por ello que se opta por su normalización antes de utilizarla como característica. Una posible normalización consiste en dividir por el área de la caja, rectangular y paralela a la horizontal, que circunscribe la firma. Al ser siempre este valor una cota superior, y al estar en las mismas unidades, el resultado será un valor adimensional entre 0 y 1 (normalmente muy por debajo de 1).

### Índice de redondez o compacidad

Este índice mide la similitud entre la firma y un círculo [2]. Este índice se confecciona comparando el área del polígono que encierra la firma (el cierre convexo) con el área del menor círculo que la circunscribe. Así,  $I_{redondez}$  se define según:

$$I_{redondez} = \frac{A_{cv}}{D_{cv}^2} \quad (4.3)$$

donde  $A_{cv}$  es el área del polígono correspondiente al cierre convexo y  $D_{cv}$  es la longitud del mayor eje de dicho polígono.



**Figura 4.20.-** A la izquierda (a) se colorea en gris el área del cierre convexo. A la derecha (b) se marca en negro el área de la proyección vertical sobre el eje de mínima inercia.

### Centros de masas globales y parciales

El centro de masas de una imagen es una medida habitual en el reconocimiento de patrones [106][2] que da lugar a dos características discriminantes  $(C_x, C_y)$ . Este par es el resultado de la suma vectorial de la posición de todos los puntos activos de la imagen dividida por el número total de puntos. Debido a la variabilidad en tamaño de las imágenes que contienen las firmas, se normaliza el par resultante, dividiendo sus coordenadas por el alto o el ancho de la firma según corresponda. Más formalmente:

$$C_x = \frac{1}{N \cdot \text{Ancho}} \sum_{n=1}^N I(n) \cdot x \quad C_y = \frac{1}{N \cdot \text{Alto}} \sum_{n=1}^N I(n) \cdot y \quad (4.4)$$

donde  $I(n)$  son los  $N$  puntos activos de la imagen.

Los centros de masas parciales son características obtenidas al dividir la imagen de la firma en varios sectores y calcular el centro de masas global de cada sector. En este trabajo se ha optado por la misma formulación que siguen en su trabajo Ismail y Gad [2]. Se divide la imagen en 4 sectores mediante dos líneas paralelas a los bordes que pasan por el centro geométrico de la misma, y luego se calcula el centro de masas de cada sector. Este proceso da lugar a 8 nuevas características discriminantes.

### Línea base global y límites superiores e inferiores de las proyecciones

Las proyecciones sobre los ejes de abscisas y ordenadas de los píxeles de la firma tienen una forma característica que varía de un individuo a otro (ver Figura 4.21). Varios trabajos [2][106] utilizan ciertas medidas sobre estas proyecciones como características discriminantes para la verificación de firmas. De estos trabajos aquí se utilizan tres características: la posición de su línea base y las posiciones de sus límites superior e inferior.

La *línea base* (*base-line* o *BSL*) se corresponde al punto de la proyección de la imagen sobre la vertical donde el número de elementos es máximo.

$$P_m = \max\{P(y)\} \text{ donde } y=1\dots M \text{ siendo } M \text{ el alto de la imagen} \quad (4.5)$$

Tras hallar la *BSL* se realiza el cálculo del *límite superior* (*upper-limit* o *UL*) y del *límite inferior* (*lower-limit* o *LL*) de la firma. Para realizar este cálculo en primer lugar es preciso obtener dos rectas, que llamaremos  $r_1$  y  $r_2$ , que unen el punto *BSL* de la proyección con los extremos del mismo. Los límites superior e inferior de la escritura se corresponden a aquellos puntos donde la distancia de la proyección a ambas rectas es máxima. Es decir:

$$UL = \max\{|P(y_1) - r_1|\} \quad LL = \max\{|P(y_2) - r_2|\} \quad (4.6)$$

donde  $y_1$  varía entre el punto superior del histograma y el punto *BSL* e  $y_2$  lo hace entre el punto *BSL* y el punto inferior.

En la imagen 4.21 se puede observar un ejemplo del cálculo de estos límites sobre un ejemplar de la muestra A. En verde se presentan las rectas  $r_1$  y  $r_2$ . La recta roja central representa la *BSL* y las rojas adyacentes representan a *UL* y *LL*, respectivamente. Para el eje vertical se obtienen 3 características equivalentes proyectando sobre el eje de abscisas. Estas características se normalizan dividiéndolas por el ancho o por el alto del mínimo rectángulo que contiene a la firma, según corresponda.



**Figura 4.21.-** Ejemplo de cálculo de la *BSL*, del *UL* y del *LL* de una firma, utilizando la proyección sobre el eje vertical de los píxeles activos de la imagen.

#### Curtosis, asimetría y relaciones entre ambas

Por último, se incluyen una serie de características propuestas inicialmente por Bajaj y Chaudhury [56], y luego usadas repetidamente en otros trabajos [23][2], relativas a la curtosis y a la asimetría de las proyecciones sobre los ejes de la firma.

$$K_v = \frac{\mu_{v4}}{\mu_{v2}^2} \quad S_v = \frac{\mu_{v3}}{\mu_{v2}^{1.5}} \quad R_b = \frac{\mu_{v3}}{\mu_{v4}^{0.75}} \quad K_b = \frac{\mu_{b4}}{\mu_{b2}^2} \quad S_b = \frac{\mu_{b3}}{\mu_{b2}^{1.5}} \quad R_b = \frac{\mu_{b3}}{\mu_{b4}^{0.75}} \\ VH_1 = \frac{\mu_{v2}}{\mu_{b2}} \quad VH_2 = \frac{\mu_{v4}}{\mu_{b4}} \quad (4.7)$$

donde  $\mu_{bn}$  representa el momento  $n$ -ésimo sobre la horizontal y  $\mu_{vn}$  el correspondiente momento  $n$ -ésimo sobre la vertical.

#### 4.3.2 Análisis de las características

En este apartado se estudia la importancia relativa de las características utilizadas mediante un análisis del *Coficiente de Fisher* (*CF*) asociado a cada una de ellas.

El problema de la verificación de firmas es un problema biclase, es decir se trata de clasificar cada firma como genuina (G) o como falsa (F). Es por ello, que se debe utilizar la formulación biclase (4.8) del ratio de Fisher. Se puede apreciar que este coeficiente, calculado para cada característica, es mayor cuando los centroides de las dos clases se encuentran separados que cuando se encuentran próximos. También aumenta cuando la desviación estándar disminuye, es decir cuando los patrones están concentrados. Por ello, una característica discriminante con un CF mayor indica una mayor capacidad discriminante.

$$CF = \frac{(\mu_G - \mu_F)^2}{\sigma_G^2 + \sigma_F^2} \quad (4.8)$$

El CF debe calcularse para cada característica de un clasificador. Como se sigue el enfoque *OCON* (*one-class one-network*) el número de clasificadores es elevado. Así, si calculamos el CF para cada uno de las características de cada individuo de la muestra A (140 ejemplares) obtendríamos una matriz de 140 filas y 37 columnas. Como esta matriz no ofrecería una información fácilmente analizable, se opta por un análisis más estadístico. Por ello, se ha calculado la media y la variación del CF para cada característica a través de los 37 clasificadores. Los resultados se recogen en la Figura 4.22 (a).

En la Figura 4.22 (a) se puede apreciar que todas las características tienen un CF similar exceptuando el área del cierre convexo. Este valor atípico se debe a que existe una clase para la que el valor del ratio de Fisher es muy alto, lo cual explica también su alta variación. Excluyendo éste y otros pocos casos extremos se obtiene la Figura 4.22 (b) en la que se aprecia unos valores CF más equilibrados entre las diferentes características. Este hecho aporta confianza en el uso de estas características.

Del estudio realizado se puede observar que entre las características con mayor poder discriminante hay algunas habitualmente olvidadas por la literatura como: el ángulo del mayor eje del cierre convexo o el área del relleno respecto al eje de mínima inercia.

### 4.3.3 Clasificador utilizado

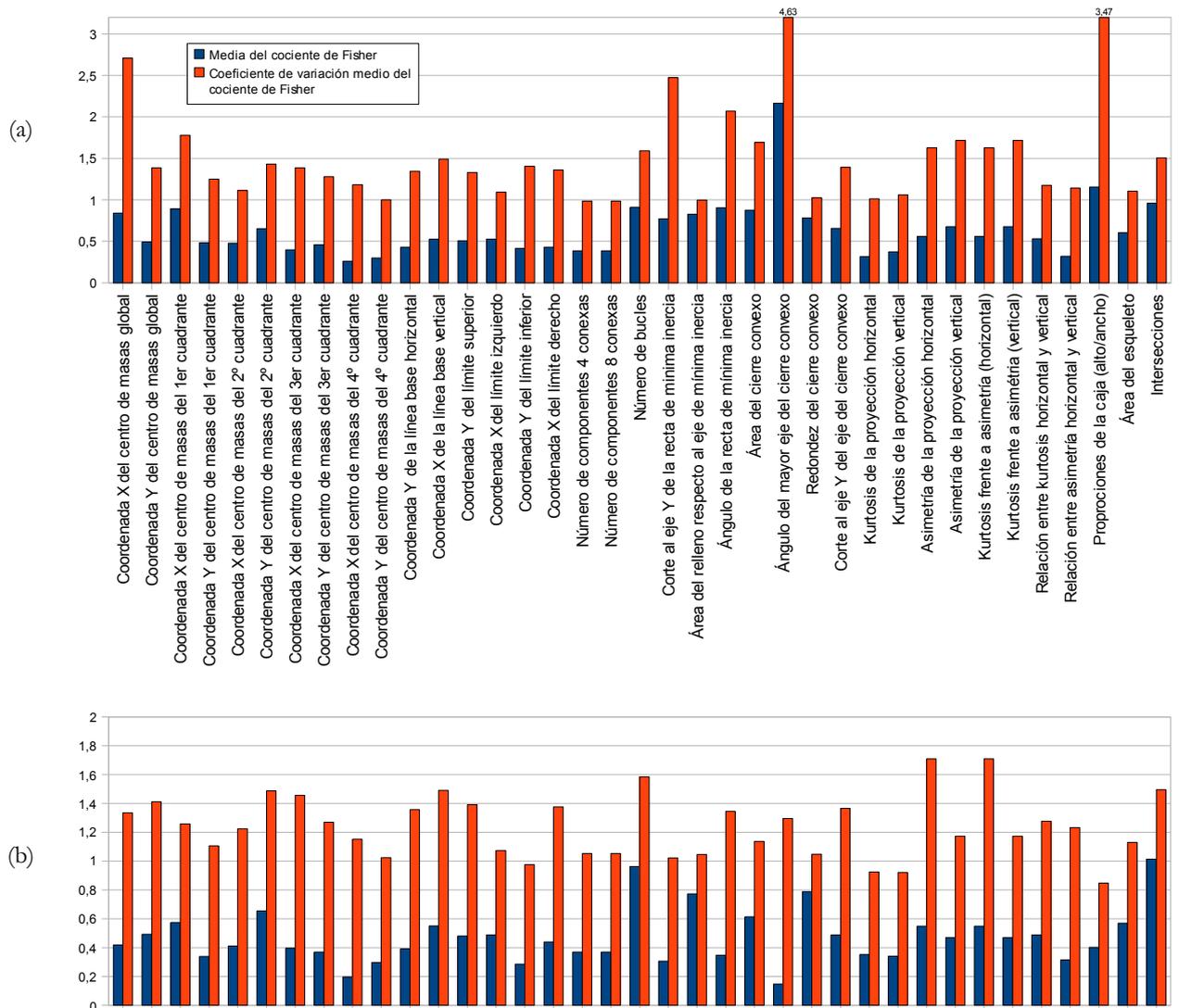
Las características discriminantes descritas deberían permitir ajustar un clasificador para utilizarlo en el problema de la verificación de firmas. De nuevo se ha optado por el uso de una Red de Neuronas *MLP* como clasificador para cada uno de los individuos a verificar siguiendo el enfoque *OCON*.

Dada la naturaleza de los datos, la estructura de la Red de Neuronas debe disponer de una capa de entrada con 37 neuronas (una por cada característica discriminante) y una capa de salida con una única neurona que determinará si aceptar o rechazar la firma. Además, tras diferentes pruebas sobre la muestra A en las que se ensayan diferentes configuraciones para las capas ocultas, se disponen 10 neuronas en una única capa intermedia.

### 4.3.4 Experimentos sobre la muestra de test

De nuevo, mientras que para el diseño de los algoritmos se ha utilizado la muestra A, para la realización de experimentos se utilizará el resto de muestras. Así, en esta sección se analizarán los resultados sobre las muestras B y E (ver sección 3.2.3).

Como la muestra B consiste en las firmas de 25 individuos diferentes, se precisa entrenar 25 Redes de Neuronas. En el entrenamiento de la red de cada individuo se utiliza sólo una de las firmas, reservando el resto para la fase de test. Gracias al algoritmo de generación de muestra sintética ya comentado, cada una de estas firmas modelo da lugar a un conjunto de entrenamiento compuesto de 90 imágenes.



**Figura 4.22.-** Media y desviación de los Coeficientes de Fisher para diferentes características discriminantes medidas sobre diferentes clasificadores de la muestra A. El histograma superior (a) se realiza antes de eliminar casos extremos y el inferior (b) después de eliminarlos.

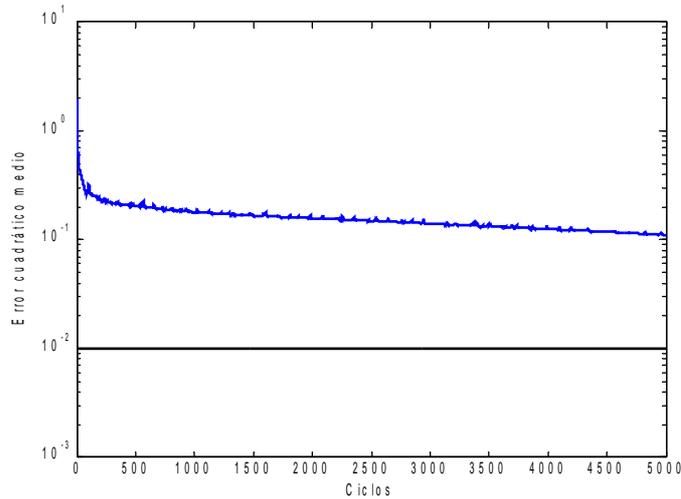
El experimento consiste en presentar a cada clasificador correspondiente a cada uno de los diferentes individuos, las 5 firmas propias, excluyendo la que se usa para entrenar el sistema. En estos 5 ensayos se esperará un resultado positivo de aceptación de la firma. Luego, se presentan las 6 firmas de los 24 individuos restantes, esperando obtener, en estos casos, rechazos.

Dado el alto número de redes a entrenar, el proceso de entrenamiento debe automatizarse. Esta automatización se concreta en detener el entrenamiento al alcanzar los 5000 ciclos o si el error cuadrático medio es inferior 0.001. La Figura 4.23 presenta como ejemplo el error cuadrático medio producido durante el entrenamiento de una red. Este ejemplo es representativo de los entrenamientos observados al entrenar las diferentes redes.

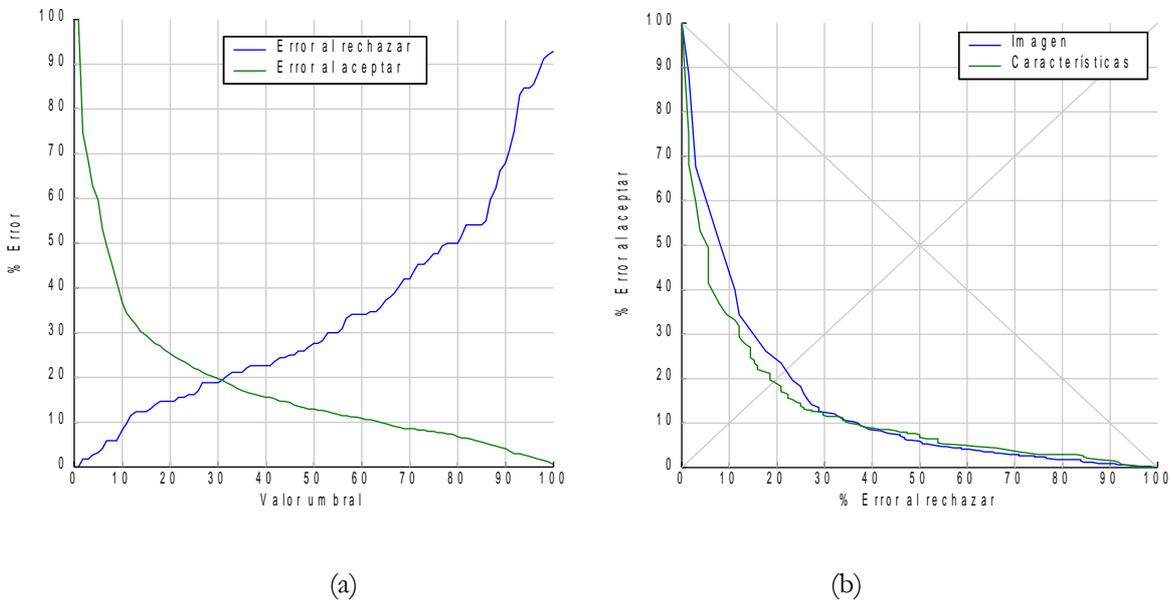
En la Figura 4.24 (a) se puede observar que el *EER* obtenido es aproximadamente del 20%. Comparativamente, la Figura 4.24 (b) permite apreciar que los resultados son ligeramente mejores

que los obtenidos cuando se utiliza directamente la imagen normalizada de la firma como único elemento discriminante.

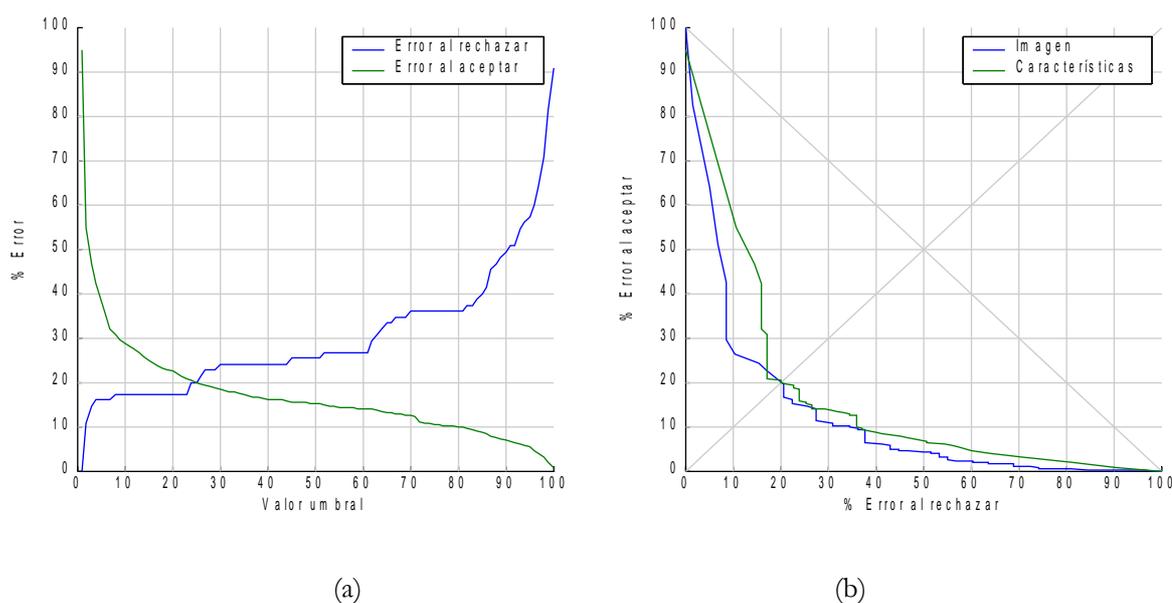
La Figura 4.25 muestra que los resultados sobre la muestra E son similares, si bien en este caso los resultados obtenidos por la imagen como única característica son ligeramente mejores. Esto implica una mayor robustez de ese enfoque frente al ruido de diferente tipo que introduce la muestra E.



**Figura 4.23.-** Curva de error obtenido durante el entrenamiento de la Red de Neuronas del primer individuo de la muestra B.



**Figura 4.24.-** Curvas de error obtenidas al usar el método basado en características discriminantes sobre la muestra B: (a) presenta el error al aceptar ( $FAR$ ) y al rechazar ( $FRR$ ) y (b) es una curva ROC que superpone el resultado obtenido con el del método basado en la presentación directa de la imagen a un  $MLP$ .



**Figura 4.25.-** Curvas de error obtenidas al usar el método basado en características discriminantes sobre la muestra E: (a) presenta el error al aceptar ( $FAR$ ) y al rechazar ( $FRR$ ) y (b) es una curva ROC que superpone este resultado con el del método basado en la presentación directa de la imagen a un  $MLP$ .

## 4.4 Verificación de firmas mediante Redes de Compresión

En esta sección se expone en detalle una idea, original de esta Tesis: la utilización de las Redes de Compresión para el problema de verificación *off-line* de firmas. En esta sección se presentan intuitivamente las Redes de Compresión y a continuación se describe el uso que de ellas se hace para la verificación de firmas. Para una introducción más formal a las Redes de Compresión remitimos al lector al anexo B de este documento.

### 4.4.1 Uso de las Redes de Compresión para la verificación de firmas

Las Redes de Neuronas de Compresión son unas redes que se entrenan para que presenten en sus neuronas de salida el mismo patrón que se le muestra en las de entrada, utilizando una función de activación lineal en sus neuronas. La particularidad de las Redes de Compresión estriba en su estructura. Ésta es tal que, en la única capa oculta que tiene una red de este tipo se dispone un número mucho menor de neuronas que las que hay en la entrada o la salida. Esto hace que para que la red pueda ofrecer a la salida el mismo patrón que recibe a la entrada deba comprimir, en esas pocas neuronas de la capa oculta, toda la información de la entrada.

Así, si el número de neuronas ocultas es menor que el número de neuronas de entrada y de salida, se está obligando a la red a codificar la entrada en un subespacio de dimensión menor, que conserva la mayor parte de la información. En definitiva, se fuerza a la red a capturar las regularidades y redundancias presentes en los patrones, extrayendo un conjunto de características menos correlacionadas que las originales (que serían todos los valores de la entrada). Por esto, las Redes de Compresión se conocen también como Redes Codificadoras o Redes Autocodificadoras [174].

Kohonen [175], utilizando imágenes del rostro humano como entrada y salida de la red, comprobó que cuando un patrón de entrada a la red estaba incompleto, a la salida la memoria autoasociativa

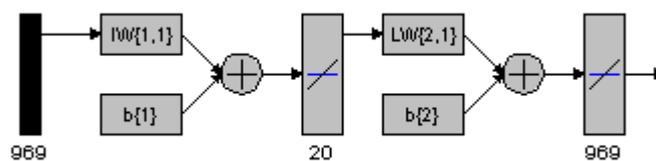
aproximaba las partes que faltaban del patrón. Según algunos autores, como D. Valentin y otros [174], las reconstrucciones predecían con éxito las zonas faltantes. Una idea de esta Tesis es que esta característica puede utilizarse para construir un sistema fiable de verificación. Al medir la diferencia entre la entrada y la salida de una Red de Compresión se está obteniendo un valor que permite medir el parecido entre el patrón, que se le presente a la red, y aquellos patrones con los que se entrenó. Es por ello que se pensó en utilizar este error como característica discriminante para la verificación de firmas. En los siguientes apartados se describen las etapas de entrenamiento de las Redes de Compresión y los primeros ensayos de verificación utilizando directamente estas redes.

### Estructura de la Red de Compresión para la verificación de firmas

La estructura de las Redes de Compresión viene determinada por las imágenes de las firmas. En particular, la Red de Compresión entrenada para cada individuo tiene a la entrada (y por tanto a la salida) un número de neuronas aproximadamente igual a 1000. Esta inexactitud se debe a que la entrada de cada red sigue las proporciones de la firma modelo que se utiliza para generar la muestra sintética. Así, si la firma de un individuo tiene unas proporciones de 1 a 3, tanto la entrada como la salida de la red, que se construya para ella, tendrán estas proporciones.

Otra variable consiste en el número de unidades ocultas. Este valor debería coincidir con el mínimo número de autovectores que describan la mayor parte de la varianza de las muestras [174]. Este número es difícil de predecir a priori, por lo que se optó por determinarlo mediante experimentación. Para ello, se han realizado diferentes pruebas sobre la muestra A con diferentes configuraciones. En estas pruebas se ha observado que un número de neuronas ocultas mayor a 15 no mejora los resultados. Es por ello, que el número de neuronas ocultas se ha fijado en 20, estableciendo un margen de seguridad de 5 neuronas.

Las estructuras de las redes descritas hasta el momento serán similares a la de la Figura 4.26. Sin embargo, aún quedan detalles por especificar. Los principales son: el formato al que se normalizan las firmas para su presentación a la red, los parámetros a usar para el algoritmo de ajuste de pesos, los valores de inicio de los pesos y el número de unidades ocultas. A priori, no existe una regla teórica fija para elegir estas opciones sino que suelen usarse criterios experimentales para la medida del error y algún método sistemático de prueba.



**Figura 4.26.-** Ejemplo de la Red de Compresión utilizada para el individuo 1 de la muestra A.

Tras diversos experimentos, se ha comprobado que los pesos de las conexiones de la red deben inicializarse con valores pequeños (entre -0.001 y 0.001). En caso contrario, tras las primeras iteraciones, el uso de una función lineal como función de activación hace que, los valores de salida de algunas neuronas alcancen valores muy altos, produciendo errores de desbordamiento en los programas. Por otro lado, el algoritmo *Backprop*, que se utiliza para el entrenamiento de la red, necesita que se configure adecuadamente el parámetro que define la velocidad de convergencia  $\mu$ . En general, el valor que se utilice debería ser suficientemente grande para evitar que el proceso caiga

en mínimos locales, y lo bastante pequeño para evitar que oscile sin converger sobre el espacio de soluciones  $n$ -dimensional. En nuestro caso, la elección de un valor alto provocaba fuertes oscilaciones que motivaban errores de desbordamiento a las pocas iteraciones. Por suerte, en la superficie del error generada por las Redes de Compresión no existen mínimos locales [176], por lo que se puede tomar un valor del parámetro  $\mu$  todo lo pequeño que se desee con la seguridad de que siempre convergerá al mínimo absoluto. El único problema en este caso puede ser que quizás converja demasiado lentamente. Al principio se probaron varios valores, hasta comprobar que un parámetro  $\mu$  igual a 0.0001 dotaba al sistema de suficiente estabilidad y no hacía excesivamente lento el aprendizaje. Con la red de 5 unidades ocultas incluso se utilizó 0.01 sin que se produjesen errores de desbordamiento.

Por último, las imágenes de las firmas deben normalizarse antes de presentarlas a la red. Esta normalización se realiza en tamaño, para que coincida con la entrada que exija cada red, y también en valores de intensidad. La normalización en tamaño utiliza una interpolación bilineal para minimizar la pérdida de resolución. Además, las intensidades de los puntos se normalizan a valores en el intervalo  $[-1,1]$ .

### Entrenamiento de las Redes de Compresión

El entrenamiento se realiza con el método *Backprop* sobre el conjunto de patrones de entrenamiento. El orden en que se evalúan los pesos de la red es el mismo en que se crearon (orden topológico), ya que el algoritmo *BackProp* no impone ningún criterio, y según C. G. Looney [177], a priori, otro criterio de ordenación no aporta ventajas.

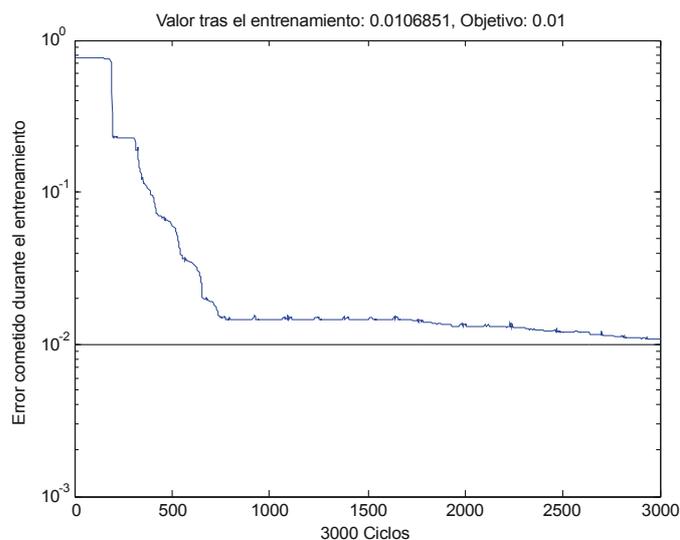
Como se ha indicado, una ventaja de las Redes de Compresión estriba en que su entrenamiento no puede caer en un mínimo local pues, por su arquitectura, esta Red de Neuronas sólo tiene un mínimo. Por ello, con objeto de reducir el error cometido por la red, el número de ciclos de entrenamiento puede ser tan elevado como se desee. En nuestro caso cada red se entrena hasta alcanzar un error de 0.01 o hasta superar 3000 ciclos. La curva descrita por el aprendizaje, para el caso de una de las firmas, se presenta en la Figura 4.27.

#### 4.4.2 Resultados del uso de las Redes de Compresión

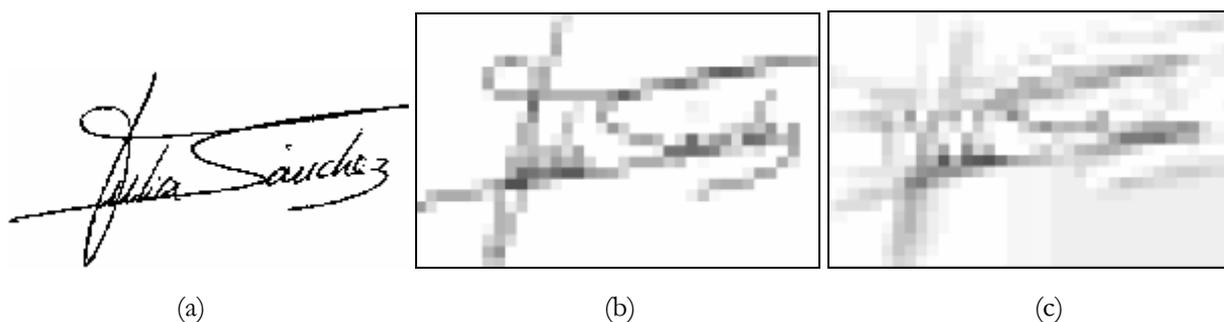
La primera idea que se planteó para la verificación de firmas utilizando las Redes de Compresión se basaba en utilizar directamente la diferencia entre la entrada y la salida de la red como característica discriminante. La idea inicial consistía en que, si a una red, entrenada para las firmas de un individuo, se le presenta una firma de tal individuo el error obtenido al comparar la salida de la red con su entrada debería ser pequeño. Por otro lado, si a la misma red se le presentase la firma de otro individuo, distinto al usado para entrenar la red, el error debería ser grande.

La Figura 4.28 ilustra la salida obtenida por una Red de Compresión al presentarle una firma que debería aceptar. En este ejemplo la red está entrenada con los patrones sintéticos del individuo 4 y la firma presentada también es del individuo 4. Se puede observar un alto parecido entre la entrada a la red y la salida obtenida.

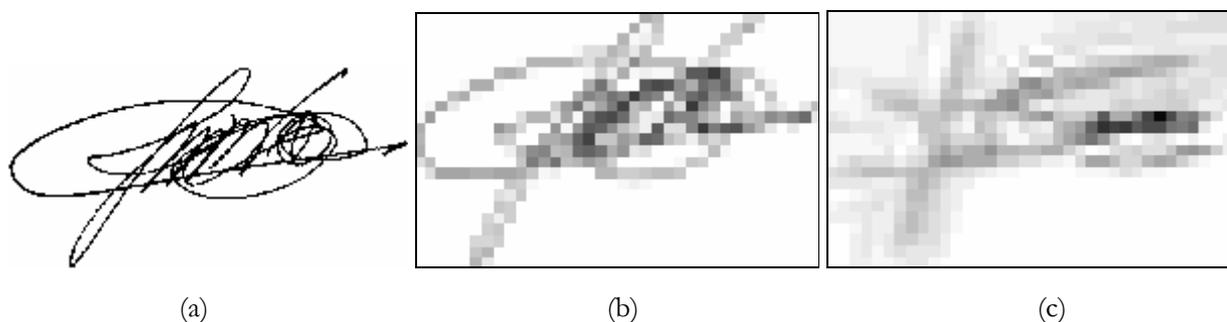
La Figura 4.29 ilustra el caso contrario, es decir la salida que ofrece una red cuando se le presenta un individuo que debe rechazar. En este ejemplo particular se vuelve a utilizar la red correspondiente al individuo 4, pero esta vez se le presenta una firma del individuo 2. Se puede apreciar una mayor diferencia entre la entrada y la salida que en el caso de la Figura 4.28.



**Figura 4.27.-** Entrenamiento de la Red de Compresión de uno de los individuos. En particular, corresponde al entrenamiento de una red para el individuo 9 de la muestra A.



**Figura 4.28.-** Resultado de presentar a la red del individuo 4 una firma del individuo 4: (a) firma a verificar, (b) entrada presentada a la Red de Compresión y (c) salida obtenida por la misma.

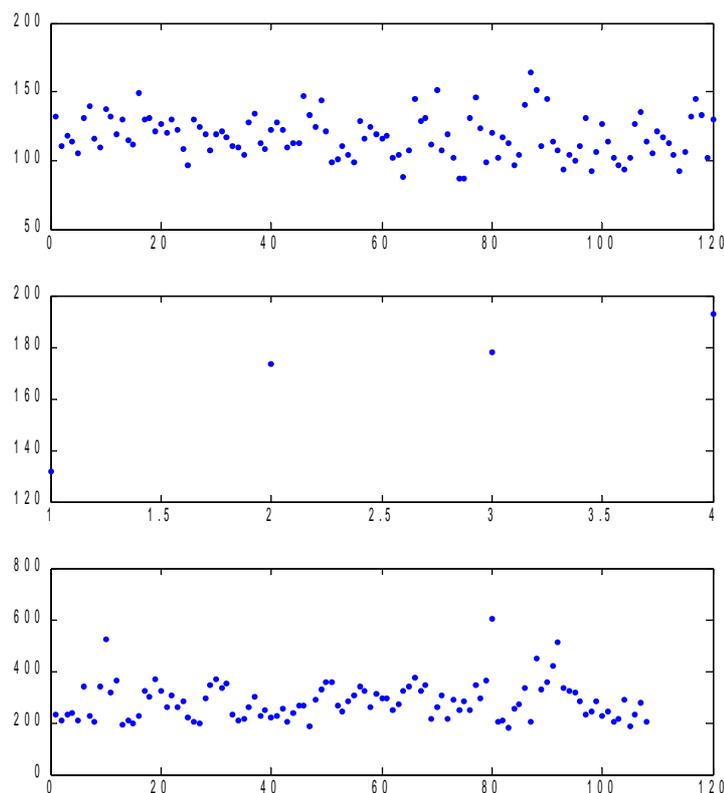


**Figura 4.29.-** Resultado de presentar a la red del individuo 4 una firma del individuo 2: (a) firma a verificar, (b) entrada presentada a la Red de Compresión y (c) salida obtenida por la misma.

La diferencia entre la entrada y la salida se puede medir mediante el error cuadrático medio, es decir, mediante la suma de las diferencias al cuadrado entre las entradas y las salidas a la red. La Figura 4.30 muestra las diferencias entre la entrada y la salida que se obtienen para una firma de la muestra A al presentar a la red (una vez entrenada) los mismos patrones sintéticos usados en el entrenamiento, otros patrones del mismo individuo y patrones de otros individuos. En el caso de la Figura 4.31 se puede comprobar que la diferencia, que en adelante llamaremos error de la red, cumple que:

- está entre 100 y 150 para los patrones sintéticos,
- está entre 170 y 195 para la muestra genuina del individuo (descartando la primera firma que se usó para generar la muestra sintética),
- finalmente, está por encima de 200 para el resto de individuos de la muestra A.

Del experimento anterior se deduce que parece existir un umbral, en este caso cercano a 200, por encima del cual se deberían rechazar los patrones y por debajo del cual se deberían aceptar. Sin embargo, los márgenes observados para este ejemplo parecen muy pequeños. Debe observarse que, en este caso, la muestra sintética es de poca ayuda para encontrar ese umbral, pues nunca supera el valor 150. Además, se ha comprobado que el valor umbral es cambiante y varía cuando se analiza la red de otro individuo. En un intento de determinar dicho umbral de una forma paramétrica se han realizado varios experimentos que se detallan en los siguientes párrafos.



**Figura 4.30.-** Error cometido por la Red de Compresión del individuo 1 de la muestra A: (a) error de las firmas sintéticas de entrenamiento, (b) error para diferentes firmas del individuo 1 y (c) error ante firmas de otros individuos.

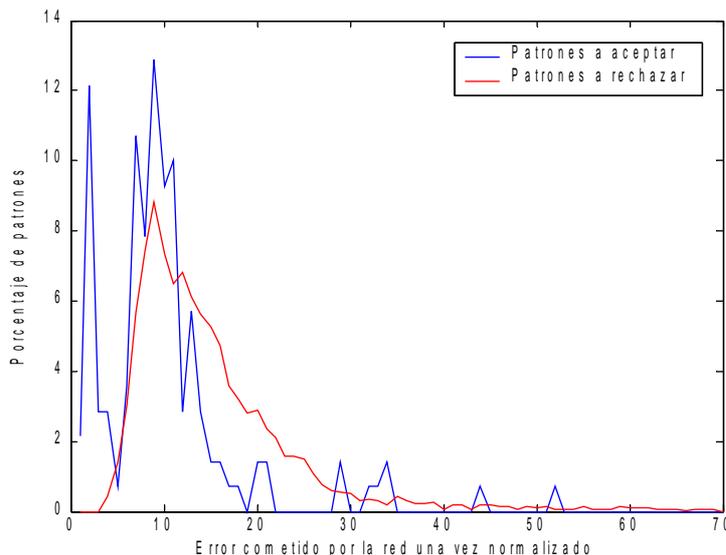
Primeramente, tras realizar algunas pruebas con la muestra A, se opta por fijar el umbral de rechazo en el doble de la media del error observado para la muestra sintética. Esto es, se utiliza un umbral diferente para cada individuo que depende del error observado por su red frente a la muestra sintética que se usa para entrenarla. Los resultados de esta idea fueron poco alentadores, obteniendo un 40% de *EER* como resultado final.

También se ha creado para cada red un Clasificador Euclídeo que separa la clase de los patrones sintéticos de una clase formada por los patrones de la muestra A que deberían ser rechazados. Este clasificador ha obtenido resultados de un 38% de *ERR*.

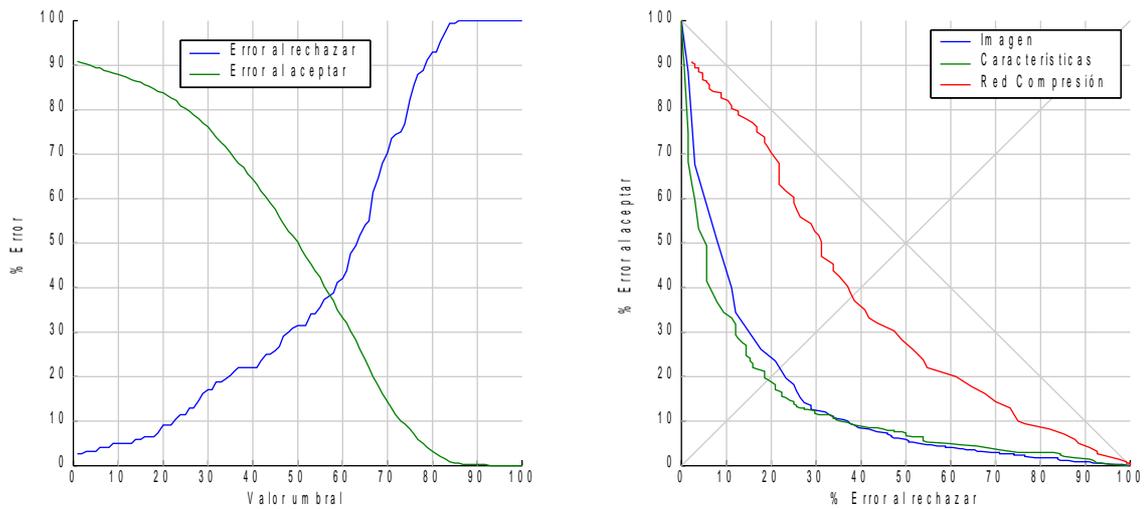
También se ha probado un clasificador estadístico basado en la distancia de Mahalanobis, pero los resultados fueron peores debido a la baja desviación típica de los patrones sintéticos en comparación con la desviación obtenida por los patrones de la muestra A (ver Figura 4.31).

Finalmente, se ha comprobado mediante el test de Kolmogorov-Smirnoff, que mas del 85% de las redes siguen una distribución Normal para el error cuando se le presentan los patrones de la muestra A. Basándonos en este hecho, y utilizando de nuevo la muestra A como modelo de la clase a rechazar, se ha probado a usar el valor de probabilidad de pertenencia a cada una de estas clases como característica discriminante. Para ello se calcula la media y la desviación típica del error arrojado por cada red cuando se le presentan patrones de la muestra A, y se utiliza la función de distribución acumulativa como característica discriminante. Los resultados que se obtuvieron con este enfoque fueron muy similares a los logrados con el Clasificador Euclídeo.

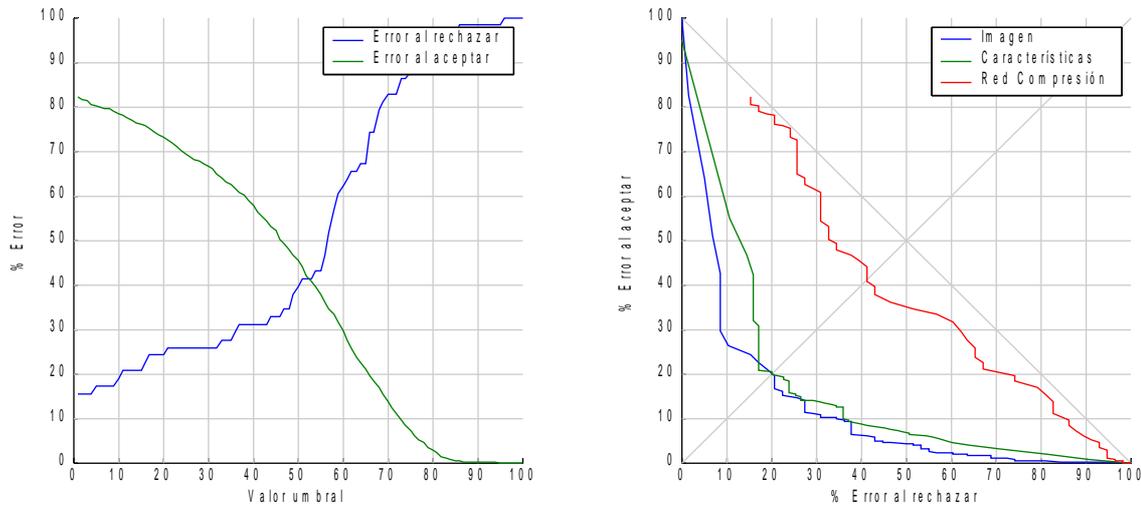
Los resultados de las figuras 4.32 y 4.33 muestran los mejores resultados obtenidos, los cuales fueron producidos por el Clasificador Euclídeo. Se puede apreciar que, sobre la muestra B, el *EER* obtenido está próximo al 38%, y en el caso de la muestra E, por encima del 41%. También puede apreciarse que son unos resultados comparativamente mucho peores que los obtenidos para el enfoque basado en características discriminantes o para el basado en el uso directo del Perceptrón Multicapa sobre la imagen.



**Figura 4.31.-** Diferencias entre la entrada y la salida de las Redes de Compresión para la muestra A.



**Figura 4.32.-** Curvas de error obtenidas al usar el método basado en Redes de Compresión sobre la muestra B: (a) presenta el error al aceptar ( $FAR$ ) y al rechazar ( $FRR$ ) y (b) es una curva ROC que superpone el resultado obtenido con: el del método basado en la presentación directa de la imagen a un  $MLP$  y el del método basado en características discriminantes globales.



**Figura 4.33.-** Curvas de error obtenidas al usar el método basado en Redes de Compresión sobre la muestra E: (a) presenta el error al aceptar ( $FAR$ ) y al rechazar ( $FRR$ ) y (b) es una curva ROC que superpone el resultado obtenido con: el del método basado en la presentación directa de la imagen a un  $MLP$  y el del método basado en características discriminantes globales.

## 4.5 Método de Cortes Posicionales para la verificación de firmas

En la Figura 4.30 se observa que existe una clara diferencia entre el error cometido por la red para los patrones de una firma genuina y de una falsificación casual. Los pobres resultados obtenidos pueden achacarse a la dificultad para encontrar de manera automática un umbral óptimo para cada

red. Esta dificultad se acrecienta debido a que sólo se dispone como modelo de un ejemplar de la firma a verificar (la cual además ya se usó en el entrenamiento de la red). El método de *Cortes Posicionales*, que aquí se propone, salva esta dificultad y consigue utilizar las Redes de Compresión para verificar las firmas.

La idea fundamental del método de los Cortes Posicionales consiste en eliminar la necesidad de utilizar directamente un umbral sobre el error que comete la red. En vez de ello, se utiliza un umbral sobre un ranking resultante de ordenar los errores cometidos al presentar a la Red de Compresión diferentes permutaciones de la firma a verificar. Así, para verificar una firma, se presenta a la Red de Compresión la imagen de la firma en cuestión y otras imágenes obtenidas mediante permutaciones de una división en columnas de la misma. En el caso de que la firma a verificar sea correcta es de esperar que la firma sin alterar, y aquéllas que tengan menores distorsiones respecto a ésta, obtengan un error menor al comprar la salida y la entrada de la Red de Compresión. Por otro lado, en el caso de que la firma a verificar sea falsa el azar determinará qué permutación se asemejará más a la falsificación.

En la Figura 4.34 se presenta un Diagrama UML de Actividad que describe el método de verificación propuesto. Se puede apreciar que primero, partiendo de la muestra sintética, se entrena una Red de Compresión por cada uno de los individuos cuya firma se desee poder verificar. Por tanto, el método sólo requiere una firma de cada individuo para la etapa de aprendizaje. Más tarde, cuando se desea verificar la validez de una firma, ésta se divide en  $n$  bandas verticales. Luego, se toman estas bandas y con ellas se generan  $n!$  imágenes mediante las permutaciones de sus posiciones. A continuación, se presenta a la Red de Compresión la firma sin modificar y las  $n!$  permutaciones. Si la firma sin modificar obtiene uno de los menores errores la firma debería aceptarse, si por el contrario, no está entre las de menor error debería rechazarse. Parece posible por tanto definir una medida de la confianza basada en la posición de la firma a verificar dentro de ese ranking.

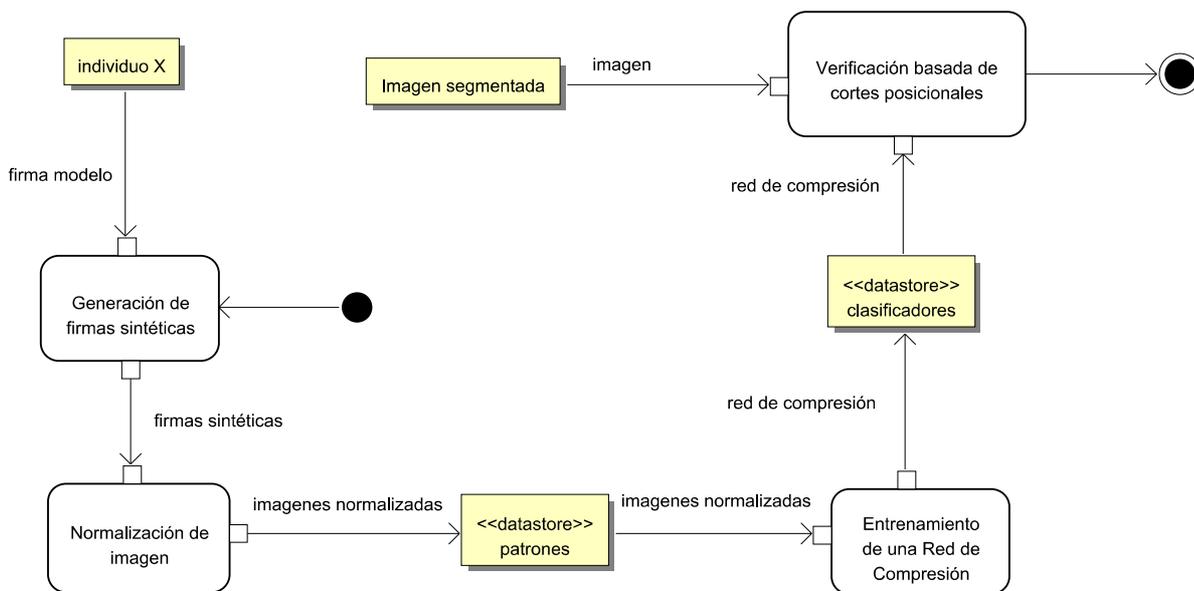


Figura 4.34.- Diagrama de Actividad UML que describe el método de Cortes Posicionales.

En un primer intento se pensó que cualquier tipo de permutación de la firma serviría para construir este ranking. Así, se probaron variaciones basadas en teselaciones, en bandas horizontales y en bandas verticales. Sin embargo, las pruebas realizadas sobre la muestra A revelaron que los mejores resultados se obtenían utilizando exclusivamente los cortes verticales. Este hecho quizás se deba a la habitual horizontalidad de las firmas, lo cual motiva que las permutaciones generen ejemplares más parecidos a los reales. El Algoritmo 4.2 ofrece una descripción formal del método propuesto.

#### Algoritmo 4.2 de verificación basado en Cortes Posicionales

**Nombre:** VerificarConCortesPosicionales

**Entradas:** F : La imagen de la firma que presuntamente pertenece a I  
n : El número de cortes posicionales  
R : La Red de Compresión del individuo I ya entrenada

**Salida:** c : La confianza en la firma

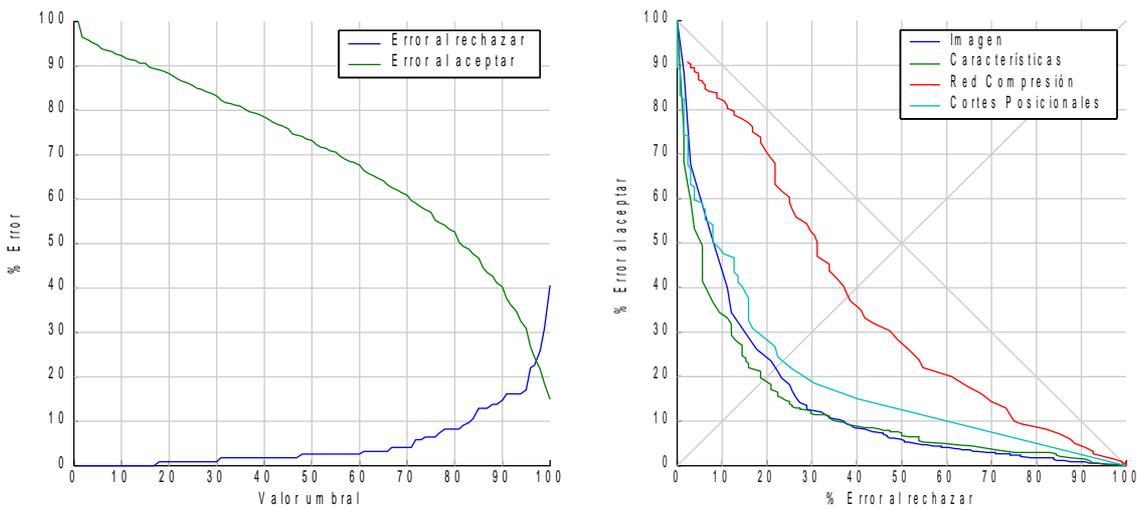
```

1 -     c[1..n] = ObtenerCortesPosicionales(F,n)
2 -     p[1..n!-1] = ObtenerPermutaciones(F,c)
3 -     error = (R(F) - F) ^ 2
4 -     mejores = 0
5 -     for (i=1..n!)
6 -         error_aux = (R(p[i]) - F) ^ 2
7 -         if (error_aux < error)
8 -             mejores = mejores + 1
9 -     c = 1 - mejores / n!
```

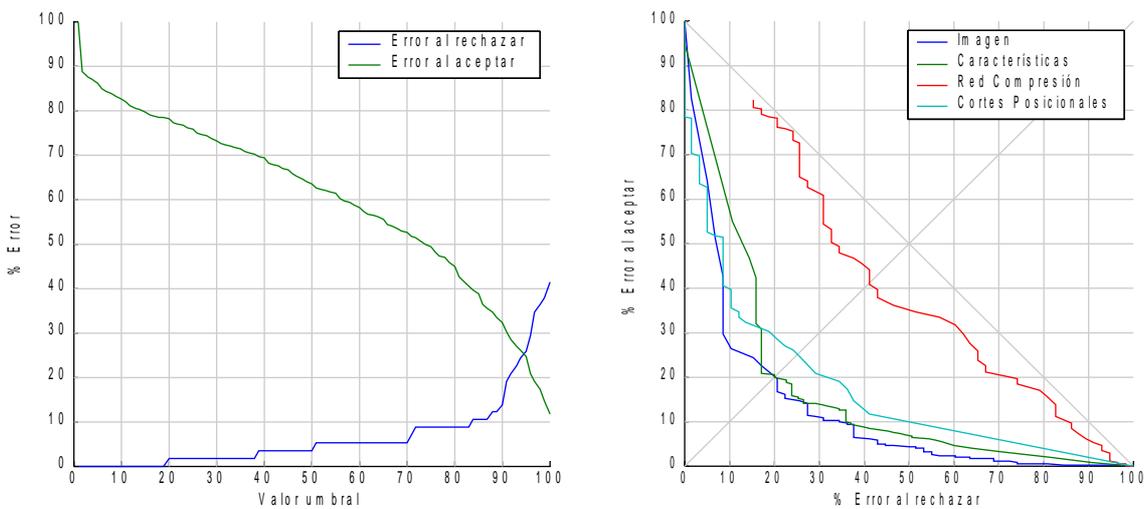
#### 4.5.1 Resultados del método de Cortes Posicionales

La Figura 4.35 presenta los resultados obtenidos por el clasificador de Cortes Posicionales sobre la muestra B. En particular, puede apreciarse un *EER* de aproximadamente un 24%. Se puede comprobar que existe una mejora substancial respecto a los resultados ofrecidos por el uso directo de la Red de Compresión, evitándose el problema de variabilidad en el umbral que aquel enfoque precisaba. Además, se puede observar que los resultados son similares, quizás ligeramente peores, a los que ofrecen los enfoques presentados al principio de este capítulo.

La Figura 4.36 presenta los resultados sobre la muestra E. En este caso, se aprecia que la pauta es similar a la comentada para la muestra B: mejora significativa respecto al uso directo de la Red de Compresión y resultados similares a los enfoques precedentes. Ahora, que la muestra utilizada es más realista, la pequeña ventaja que se apreciaba en el experimento anterior no resulta tan clara, existiendo una zona de bajo error al rechazar en la que los mejores resultados los ofrece este método. Sin embargo, dicha diferencia tampoco es significativa.



**Figura 4.35.-** Curvas de error obtenidas al usar el método de Cortes Posicionales sobre la muestra B: (a) presenta el error al aceptar (*FAR*) y al rechazar (*FRR*) y (b) es una curva *ROC* que superpone el resultado obtenido con: el del método basado en la presentación directa de la imagen a un *MLP*, el basado en características discriminantes globales y el basado en Redes de Compresión.



**Figura 4.36.-** Curvas de error obtenidas al usar el método de Cortes Posicionales sobre la muestra E: (a) presenta el error al aceptar (*FAR*) y al rechazar (*FRR*) y (b) es una curva *ROC* que superpone el resultado obtenido con: el método basado en la presentación directa de la imagen a un *MLP*, el basado en características discriminantes globales y el basado en Redes de Compresión.

## 4.6 Resumen del capítulo

En este capítulo se ha presentado un método de generación de muestra sintética que simula la variación intrapersonal inherente a las diferentes firmas de un individuo.

Se han estudiado los resultados de verificación ofrecidos por dos métodos clásicos de verificación: uno basado en el uso de una Red de Neuronas sobre la propia imagen de la firma y otro basado en características discriminantes. Ambos métodos se han modificado mediante el uso de muestra sintética para que sólo sea preciso utilizar una firma durante sus fases de aprendizaje.

Además, se han presentado dos enfoques novedosos basados en Redes de Compresión. El primero, utilizando directamente el error de la Red de Compresión al reconstruir la imagen que se presenta a su entrada como característica discriminante. El segundo método, denominado de Cortes Posicionales, está basado en la comparación de diferentes imágenes que se obtienen al permutar diferentes cortes verticales practicados sobre la firma a verificar.



## Capítulo 5 Verificación de firmas mediante *snakes*

En este capítulo se expone en detalle una idea, original de esta Tesis, de utilizar *snakes* para el problema de verificación *off-line* de firmas.

### 5.1 Formulación original de los *snakes*

Un *snake*, en su formulación original [26], consiste en una curva o contorno, abierto o cerrado, que se desplaza sobre una imagen a la que trata de adaptarse para lograr una segmentación de una parte de la misma. La curva que define el *snake* corresponde a una curva paramétrica (una línea poligonal, un spline, una curva de Bezier...) definida por una secuencia de *puntos de control*. El caso más simple consiste en una línea poligonal con  $N$  vértices, donde los puntos de control son los vértices  $v_i$  de dicha línea poligonal, tomando  $i$  valores entre 1 y  $N$ . En adelante, se utilizará esta formulación y se usará indistintamente el término vértice o punto de control. Para una introducción más formal a los *snakes* se recomienda la lectura del anexo A.

#### Funciones de energía

Al *snake* en cada instante  $t$  se le asigna una energía resultado de la suma de una *energía interna* al *snake* y otra *energía externa* que actúa sobre él (véase ecuación 5.1). La energía interna se deriva de la posición relativa en el instante  $t$  de los puntos de control que definen la curva  $v_i(t)$ . Por otro lado, la energía externa se deriva de la posición de esos puntos del *snake* sobre la imagen en ese mismo instante. En este contexto el *snake* se mueve sobre la imagen buscando una posición de mínima energía.

$$E_{snake}(t) = E_{interna}(t) + E_{externa}(t) \quad (5.1)$$

En su formulación original la fuerza interna del *snake* consta de dos términos: el primero, ponderado por un factor  $\alpha$ , controla la elasticidad de los segmentos del *snake* y el segundo, ponderado por un factor  $\beta$ , controla su rigidez [178]. Así, un valor alto de  $\alpha$  permite que los segmentos se estiren mucho, mientras que un  $\alpha$  bajo hace que se contraigan. A su vez, en el caso de un *snake* abierto, un valor alto del factor  $\beta$  permite que el *snake* se curve mucho, mientras que un  $\beta$  bajo fuerza al *snake* a que sus segmentos se configuren en línea recta. Para la energía externa, Kass y otros, proponen un parámetro  $w$  que pondera la atracción del *snake* hacia los valores de la imagen.

Por otro lado, la energía externa se deriva de la posición que tenga el *snake* sobre la imagen. En general, esta energía se suele definir de manera que disminuya cuando se aproxime a ciertos elementos objetivo dentro de la imagen, como bordes o esquinas. De esta forma, determinados objetos de la imagen “atraen” al *snake*. Para que la energía externa exhiba este comportamiento se suele utilizar una formulación que contiene la propia imagen y sus derivadas, de manera que la energía sea mayor ante ciertos niveles de intensidad o en los bordes de los objetos.

### Proceso habitual de inicialización y convergencia del *snake*

El primer paso para utilizar un *snake* es definirlo. Como su uso habitual es la segmentación de objetos sobre imágenes, se suele crear una línea poligonal cerrada con una forma similar al objeto que se desea segmentar. A continuación, el *snake* se posiciona en un lugar próximo a la zona donde se sospecha que se encuentra el objeto a segmentar. Es importante la proximidad a la zona para minimizar la posibilidad de caída en mínimos locales. Para terminar esta primera etapa de inicialización se deben asignar valores convenientes a los parámetros que controlan el comportamiento del *snake*.

Tras la etapa de inicialización se aplica un proceso iterativo de descenso del gradiente, que minimiza localmente la energía del *snake* sobre la imagen variando la disposición de sus puntos de control. En este proceso, el *snake* se desplaza sobre la imagen hasta alcanzar una posición estable, momento en el cual el proceso finaliza. La convergencia de este proceso está asegurada, ya que el número de estados es finito (pues se trabaja sobre una imagen digital) y en cada paso la energía sólo puede descender.

### Uso habitual de los *snakes*

El método de ajuste del *snake* descrito funciona bien en presencia de ruido. Por ello, el uso habitual de los *snakes* ha sido la segmentación de objetos en imágenes digitales ruidosas, encontrando numerosos trabajos en el área de las imágenes médicas y meteorológicas. Por ejemplo: segmentación de huesos en radiografías [179], segmentación de secciones del cerebro en imágenes de resonancia magnética, segmentación de fetos en ecografías [178] o segmentación de nubes en imágenes de satélite [180].

#### 5.1.1 Aplicación de los *snakes* para la verificación *off-line* de firmas

Una idea original de esta Tesis<sup>11</sup> consiste en utilizar los *snakes* para un proceso de verificación en vez de para su habitual uso en problemas de segmentación de objetos.

La clave para su aplicación al problema considerado consiste en crear un *snake* con la forma que se desea verificar utilizando una firma modelo (o firma de aprendizaje). Luego se debe posicionar el *snake* sobre la imagen que contiene la firma a verificar con una orientación y una posición cercana a dicha firma. Tras esto se debe iterar el *snake* hasta que su energía converja a un mínimo. Finalmente, se comprueba si la adaptación conseguida es aceptable. Este proceso se encuentra descrito por el Diagrama de Actividad de la Figura 5.1.

Así, puede deducirse que, con objeto de disponer de un sistema completamente automático de verificación, se debe contar con los siguientes elementos:

- Un algoritmo que permita crear un *snake* a partir de la imagen de una firma modelo.
- Un algoritmo que permita el ajuste del *snake* y que se adapte a las singularidades del problema.
- Un conjunto de características discriminantes, basadas en el resultado del proceso de ajuste, que permitan determinar la validez o falsedad de una firma.

---

<sup>11</sup> Ésta no es la primera vez que se utiliza la técnica de *snakes* en relación con la problemática de las firmas. En 1994, Pavlidis y otros [16][15] utilizaron *snakes* para el problema de la identificación de firmas. Estos autores propusieron que la deformación de un polígono que envuelve la firma unido a un proceso de convergencia basado en *snakes* se puede utilizar para identificar al firmante de entre un conjunto predeterminando de firmantes. El enfoque planteado en esta Tesis es muy diferente al de ese trabajo. La primera diferencia está en que aquí se aborda el problema de la verificación y no el de la identificación. Pero la diferencia más importante consiste en que en el trabajo de Pavlidis el *snake* no se configura con la forma de los trazos sino con la envolvente de la firma.

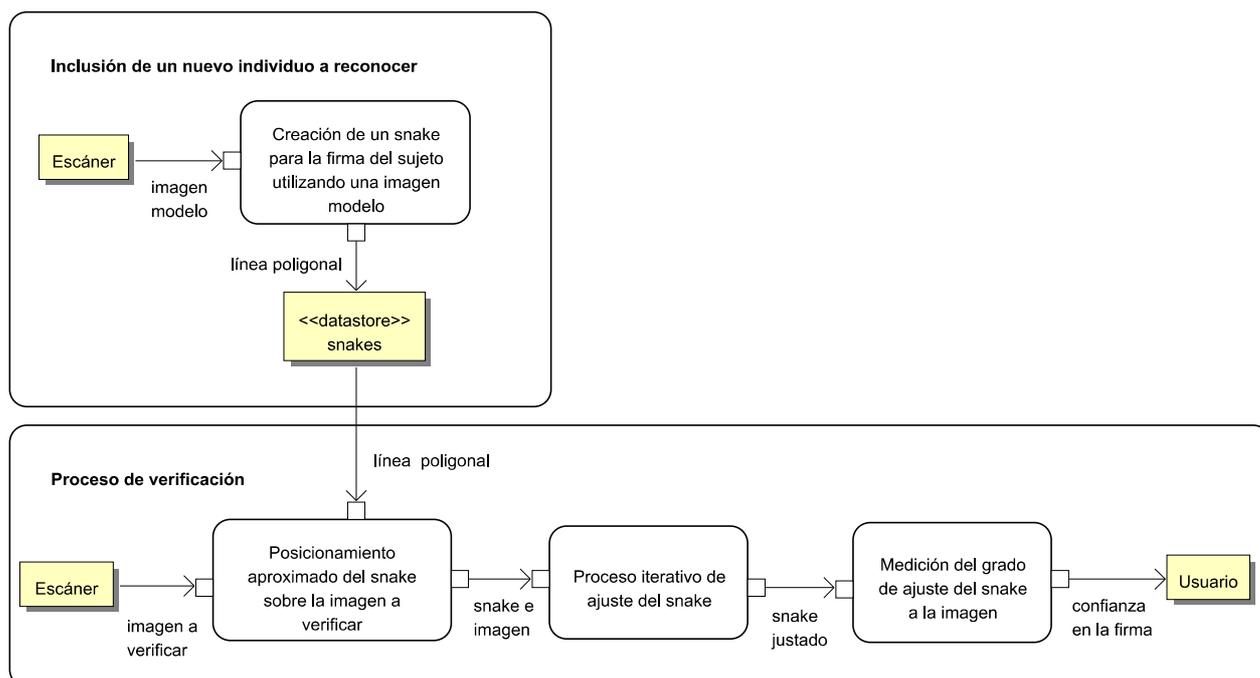


Figura 5.1.- Diagrama de Actividad que describe el uso de *snakes* para la verificación de firmas.

### Generación del *snake* utilizando una firma modelo

El enfoque planteado en este capítulo requiere que previamente al proceso de verificación se disponga de un *snake* de la firma a verificar. La sección 5.2 se dedica al estudio del problema de generación del *snake* a partir de una imagen modelo.

### Convergencia del *snake* sobre una imagen a verificar

Como se explica en la Figura 5.1, para la verificación de una firma que presuntamente pertenece a un individuo se debe posicionar el *snake* de ese individuo sobre la imagen de la firma a verificar. Luego se debe aplicar un algoritmo de convergencia que minimice la energía del *snake*. Durante este proceso iterativo el *snake* se desplazará, serpenteando sobre la imagen, desde su posición inicial hasta una posición de coincidencia, idealmente máxima con la firma a verificar. En este proceso son claves la inicialización del *snake* y la formulación de los términos de energía.

Debido a la naturaleza del algoritmo de convergencia, que suele utilizar descenso de gradiente, es posible que el *snake* caiga en un mínimo local durante su movimiento de ajuste. Por ello, para evitar la caída en algún mínimo local, es importante la determinación de la posición inicial del *snake*. Por otro lado, la formulación exacta de los términos de energía es importante para que el *snake* sea atraído a seguir una trayectoria que le conduzca a un valor bajo de energía. Las secciones 5.3 y 5.4 se dedican a estudiar estos detalles del proceso de convergencia del *snake*.

### Determinación de las características discriminantes

Una vez que el *snake* ha convergido a una posición de mínima energía es necesario medir cómo se ha adaptado a la firma a verificar. Si la adaptación se considera adecuada, se deberá aceptar la firma de la imagen a la que el *snake* se ha ajustado como genuina. En caso contrario, deberá considerarse esa firma como una falsificación.

La sección 5.5 se dedica a estudiar diferentes características discriminantes que se usan para determinar este grado de adaptación

### Ventajas del enfoque basado en *snakes*

El procedimiento descrito de verificación basado en *snakes* aportará las siguientes ventajas:

- Sólo se usará una firma para crear un modelo de *snake* y, por tanto, para construir un sistema de verificación de firmas *off-line*.
- La variabilidad intrapersonal en las firmas de un sujeto no será significativa para el algoritmo de ajuste del *snake*, mientras que la variabilidad interpersonal de las falsificaciones deberá producir valores altos de energía que permitirán su detección.
- Como se ha demostrado en otros trabajos [179][178], el *snake* resultará ser un enfoque robusto frente al ruido con estructura. Mientras que el ruido aleatorio (sin estructura) podría eliminarse mediante un filtrado morfológico previo.

## 5.2 Generación de *snakes* usando una imagen modelo

En esta sección se describe un método que permite obtener un *snake* a partir de la imagen de una firma que se toma como modelo.

Las alternativas consideradas han sido dos:

- El *snake* modelo se obtiene directamente utilizando un dispositivo de captura *on-line*.
- El *snake* modelo se obtiene de manera *off-line* a partir de una firma modelo.

### 5.2.1 Obtención *on-line* del *snake* modelo

Utilizando un dispositivo de captura de tipo tableta digitalizadora se puede obtener el conjunto de puntos que definen el *snake*.

Existe cierta variabilidad en cuanto a los tipos de tabletas que pueden utilizarse para este fin. Con objeto de que el resultado obtenido luego sea utilizable sobre imágenes *off-line* de firmas, sobre las que finalmente deberá operar el sistema, es fundamental garantizar el parecido visual entre el modelo y las firmas que el sujeto suele realizar utilizando elementos convencionales de escritura. Por ello, para no modificar substancialmente la dinámica del firmante, es importante que:

- Exista realimentación visual (*feedback*) para que el firmante, al ver lo que escribe mientras ejecuta la firma modelo, realice las microcorrecciones inconscientes de trayectoria que permitan asegurar el parecido visual con sus firmas habituales.
- Los elementos que intervienen en la captura, como el tipo de puntero o la superficie sobre la que se firma, deben proporcionar una sensación similar a la que habitualmente se obtiene con un papel y un bolígrafo.

La siguiente tabla presenta los tipos de dispositivos de captura *on-line* existentes, indicando sus ventajas e inconvenientes.

Dispositivos	Ventajas	Inconvenientes
<b>Tabletas digitalizadoras con puntero sin tinta</b>	Son los dispositivos más baratos y robustos	No proporcionan realimentación visual, por lo que la firma generada puede no parecerse a la firma que realiza una persona cuando ve lo que escribe
<b>Tabletas digitalizadoras sobre pantalla táctil</b>	Al ver lo que se está escribiendo (gracias al eco proporcionado por la pantalla) permiten al firmante obtener la realimentación visual necesaria para asegurar el parecido con sus firmas	La escritura sobre una pantalla, provoca una sensación diferente (agarre del puntero, tamaño del dispositivo o postura del firmante) que puede modificar el aspecto final de la firma. Además, los dispositivos de este tipo son frágiles
<b>Tabletas digitalizadoras de puntero con tinta</b>	Permiten al firmante una sensación idéntica a la que obtiene cuando realiza la firma con medios convencionales (bolígrafo y papel)	Son los dispositivos más caros y menos robustos

Tabla 5.1.- Ventajas e inconvenientes de los diferentes dispositivos de captura *on-line* de firmas.

### 5.2.2 Generación automática *off-line* de snakes

Inicialmente, la obtención *off-line* de los primeros *snakes*, a partir de imágenes modelo, se ha realizado mediante trazado manual, utilizando para ello una herramienta ad hoc.

En principio puede resultar extraño que el proceso de seguimiento de un trazo *off-line* pueda presentar algún tipo de dificultad. Se ha comprobado que si a cualquier persona se le muestra una firma y se le pide que siga su trazo con un lápiz y sin levantarlo, ésta descubrirá enseguida la manera de realizar la tarea. En la cultura occidental, seguramente comenzará por algún inicio de trazo que se encuentre a la izquierda, mientras que personas de otras culturas (p. ej. árabe) quizás empiecen por la derecha. Desafortunadamente, como todo lo que el ser humano realiza de manera natural y sin seguir un proceso que se preste a una definición concreta, dicha tarea no es un proceso trivial [8].

La experiencia acumulada, tras el trazado manual para las más de 50 imágenes (correspondientes a la primera imagen de cada individuo de las muestras A y B), ha permitido enunciar las siguientes reglas heurísticas respecto a este proceso de creación del *snake* modelo:

1. Se suele comenzar por un punto de comienzo de trazo situado en la parte izquierda de la imagen.
2. Se suele seguir el camino definido por los píxeles activos, sin cambiar de componente conexa hasta que se llega a un punto extremo.
3. El individuo evita realizar dos veces un mismo trazo.
4. Ante las dudas que surgen en algunos cruces entre trazos, se suele seguir la trayectoria que implica un menor cambio de pendiente. A esto lo denominamos *criterio de suavidad*.

Obsérvese el parecido de estas reglas con las propuestas de J. C. Pan y S. Lee [8] comentadas en la sección 2.3.2.

Otro punto de estudio consiste en la decisión relativa al número de puntos que debe tener el *snake*. Primeramente, se probó a utilizar el mínimo número de puntos de control que permitiese a la línea poligonal estar completamente contenida en el trazo de la firma. Se encontró que este enfoque daba lugar a segmentos demasiado largos que luego no se adaptaban correctamente a la firma por ser demasiado rígidos. Así, tras diversos experimentos, para que el *snake* generado sea útil en los procesos de verificación posteriores, se han formulado las siguientes reglas empíricas:

1. Se deben definir los puntos del *snake* a intervalos regulares para no crear segmentos largos y rígidos que luego no se adapten a las variaciones de la firma.
2. El trazado del *snake* debe seguir un orden similar al que siguió el individuo que firmó, con objeto de aprovechar las características de deformación de los *snakes*.

Con el fin de no tener que trazar manualmente un *snake* para cada firma modelo de cada individuo se ha diseñado un algoritmo de generación automática de *snakes* que cumple las reglas descritas.

### **Descripción general del método de generación automática de *snakes***

El proceso de generación automática de *snakes* que se plantea en este trabajo se basa en la idea de utilizar el algoritmo de esqueletizado de firmas, planteado por Doermann y Rosenfeld [181], para realizar seguimiento del trazo. El inicio del recorrido se realiza preferentemente en puntos extremos del *snake*, aunque para evitar que un bucle pueda quedar sin trazar también puede comenzar el trazado en un punto cualquiera. Este proceso de seguimiento se interrumpe cuando se alcanza un punto de cruce o un punto ya visitado. Esto da como resultado un conjunto sobresegmentado de trazos, los cuales se unirán en la etapa siguiente con objeto de generar una única línea poligonal continua que será el *snake*. Esta unión de segmentos se realiza en dos fases: primero, uniendo los segmentos que se tocan por sus extremos, y luego, uniendo los segmentos que no se tocan mediante segmentos fuera de trazo. En aquellos casos en los que dos o más segmentos tengan un extremo común, se unen en primer lugar aquéllos dos cuyos ángulos guarden mayor similitud. Finalmente, se trata de minimizar la longitud de los segmentos fuera de trazo.

La Figura 5.2 representa un esquema general del proceso, y en los siguientes párrafos se describe en detalle cada uno de los pasos del mismo.

El primer paso consiste en realizar un esqueletizado de la imagen de la firma. Se ha utilizado la variación del algoritmo de Zhang y Suen propuesta en la subsección 4.1.1. En este nuevo contexto, este algoritmo plantea un problema: en ocasiones, principalmente en las intersecciones entre trazos, el resultado no parece seguir la línea natural del trazo (véase un ejemplo en la Figura 5.3). Este problema influye en la solución adoptada de unión de segmentos, ya que dificulta la aplicación del criterio de suavidad previamente descrito.

Tras el esqueletizado se crea una lista con los segmentos que componen la firma. Para ello, primero se localizan los puntos finales de dichos segmentos. Estos puntos finales se corresponden con los puntos finales de los trazos y con los puntos de intersección entre dos o más trazos de una firma.

La detección de las intersecciones ya se comentó en el punto 4.3.1 y no presenta mayor problema. Por otro lado la detección de los puntos finales de segmentos en una imagen consiste en la búsqueda de aquellos puntos activos de la imagen esqueletizada que sólo tienen un vecino activo. La Figura 5.4 presenta el resultado de la aplicación de este sencillo algoritmo sobre una imagen.

El algoritmo 5.1 detalla el proceso de creación de la lista de segmentos sobresegmentados. El algoritmo comienza a partir de los puntos extremos de la firma esqueletizada o, en su defecto, a partir de cualquier punto de la misma. En sucesivas iteraciones añade puntos a una variable llamada "segmento\_actual". Cuando el siguiente punto a añadir forma parte de una intersección o es un

punto extremo el “segmento\_actual” se considera completo y se comienza a construir otro. Conforme se recorre la imagen, se van eliminando los puntos activos de la misma, terminando cuando la imagen se queda sin puntos activos.

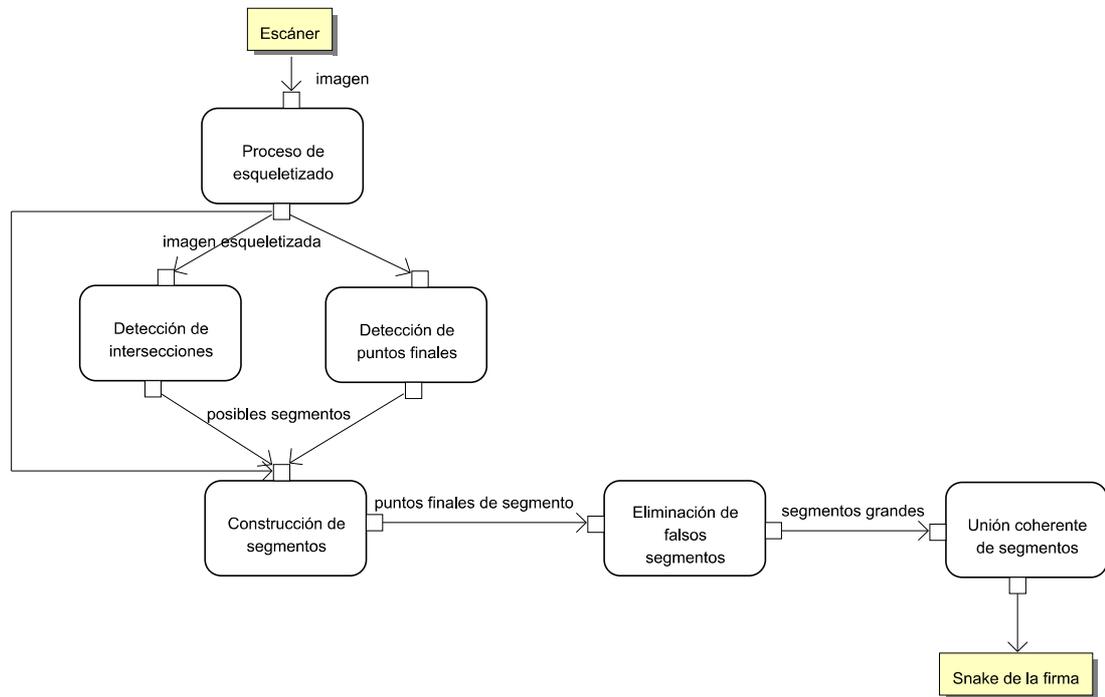


Figura 5.2.- Contexto de la generación automática de *snakes*.



Figura 5.3.- Ejemplo en el que el esqueletizado no sigue de manera natural al trazo.

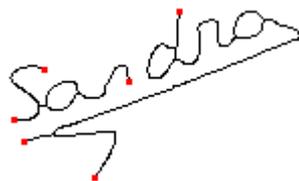


Figura 5.4.- Resultado de la aplicación del algoritmo de localización de puntos extremos sobre una firma.

**Algoritmo 5.1 de sobresegmentación de trazos**

**Nombre:** SobresegmentaTrazos

---

**Entradas:** I : Imagen esquelizada

---

**Salida:** T : Conjunto de trazos

---

```

1 -   while (!I.Vacio())
2 -       cjto_extremos = Extremos(I)
3 -       if (cjto_extremos ≠ ∅)
4 -           i = cjto_extremos.Primerero()
5 -       else
6 -           i = I.PuntoAlAzar()
7 -       segmento_actual = ∅
8 -       fin_trazo = false
9 -       while (!fin_trazo)
10 -           segmento_actual.Añadir(i)
11 -           I.Borrar(i)
12 -           cjto_vecinos = I.VecinosActivos(i)
13 -           if (cjto_vecinos.Tamaño() ≠ 1)
14 -               T.Añadir(segmento_actual)
15 -               fin_trazo = true
16 -           else
17 -               i = cjto_vecinos.Primerero()

```

Para obtener el *snake* tan sólo queda realizar una reconstrucción coherente de la firma a partir de la lista de segmentos que se ha creado. Para ello, se sigue un procedimiento que consiste en ir uniendo los segmentos de la lista según un criterio primero de distancia y luego de suavidad. En consecuencia, entre varios segmentos que se toquen por sus extremos se continúa el recorrido por aquel camino que conlleve un menor cambio de pendiente.

El algoritmo 5.2 describe formalmente el procedimiento de unión de segmentos. En el algoritmo se utiliza una función llamada “Distancia” que debe contemplar los dos criterios de distancia y suavidad. Utilizando esta función, el algoritmo busca los dos trazos más próximos entre sí en el conjunto de trazos de entrada. Una vez encontrados los elimina e inserta el resultado de su unión. El algoritmo termina cuando todos los trazos se han unido en uno solo.

**Algoritmo 5.2 de unión de trazos sobresegmentados**

**Nombre:** UneTrazos

**Entradas:** C : Conjunto de trazos

**Salida:** T : Trazo resultado de la unión de los trazos de entrada

```

1 -      while (C.Tamaño() > 1)
2 -          for x = 1 to C.Tamaño()
3 -              for y = x+1 to C.Tamaño()
4 -                  matriz_dist[x][y] = Distancia(C[x],C[y])
5 -                  [m_x,m_y]= Minimo(matriz_dist)
6 -                  C.Insertar(Unir(C[m_x],C[m_y]))
7 -                  C.Eliminar(C[m_x])
8 -                  C.Eliminar(C[m_y])
9 -      T = C
    
```

**Resultados del proceso de generación automático de *snakes***

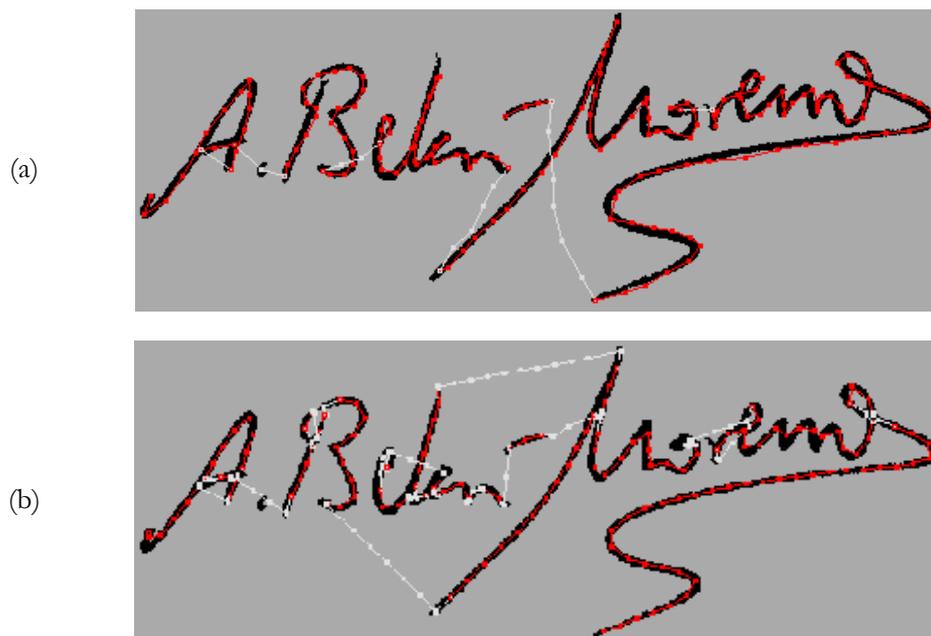
En general, los resultados obtenidos son satisfactorios, si bien el trazado realizado por una persona suele diferir, sobre todo en puntos donde se pierde la suavidad. Se ha comprobado que esto se suele deber a un esqueletizado deficiente. La Figura 5.5 presenta un ejemplo de la aplicación del algoritmo sobre una imagen y la comparación con un trazado realizado manualmente. A pesar de las diferencias, como se comprobará en los experimentos posteriores, éstas no parecen afectar a los resultados del proceso de verificación de firmas.

**5.3 Ajuste de un *snake* a la imagen de una firma**

En esta sección se presenta un método novedoso que permite realizar el ajuste de un Contorno Activo a la imagen bitonal de una firma. El método se basa en una serie de modificaciones a la propuesta original de *snakes* realizada por Kass, Witkin y Terzopoulos [26].

**5.3.1 Uso del algoritmo original de *snakes***

En primer lugar se propuso utilizar directamente el algoritmo original de Kass, Witkin y Terzopoulos [26] sobre las imágenes de las firmas. Sin embargo, dada la naturaleza del problema, pronto hubo que realizar las primeras adaptaciones, las cuales se describen en los siguientes puntos.



**Figura 5.5.-** Ejemplo del resultado de la generación del *snake* a partir de una imagen modelo: (a) mediante trazado manual y (b) mediante trazado automático.

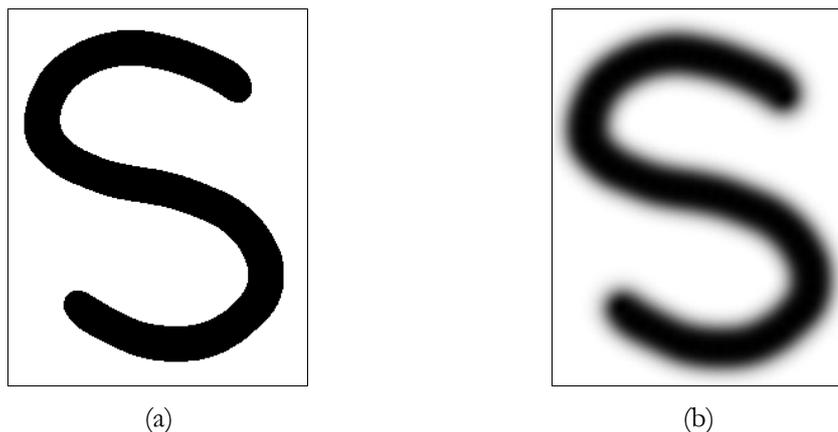
### Primer uso del *snake* sobre un trazo

Como ya se ha señalado, la descripción original de *snakes* utiliza imágenes en niveles de gris para moverse sobre ellas. Gracias a estos niveles de gris se pueden calcular derivadas para encontrar los bordes de las figuras y a la vez obtener un gradiente de aproximación a los objetos. Sin embargo, en el caso de las firmas ya se justificó, en la subsección 2.2.3, el uso de imágenes bitonales. Así pues, la formulación original, basada en el uso de imágenes en niveles de gris para la energía externa, es inadecuada. Por ello, para crear un gradiente de aproximación a la firma similar al obtenido por el algoritmo original, se aplicó un suavizado gaussiano con una ventana amplia (de  $25 \times 25$  píxeles para la muestra A). Además, se favoreció que el *snake* buscara posicionarse sobre los puntos activos de la imagen mediante la asignación de un valor  $w$  muy bajo (en la ecuación A.4), respecto a los valores de  $\alpha$  y  $\beta$  (en la ecuación A.2). Es decir, se forzó al *snake* a buscar los píxeles oscuros, en principio correspondientes a la firma, manteniendo unas fuerzas internas débiles. Sobre las firmas, este ajuste mejoró los resultados de la propuesta original de *snakes* a costa de hacer el *snake* más “maleable”.

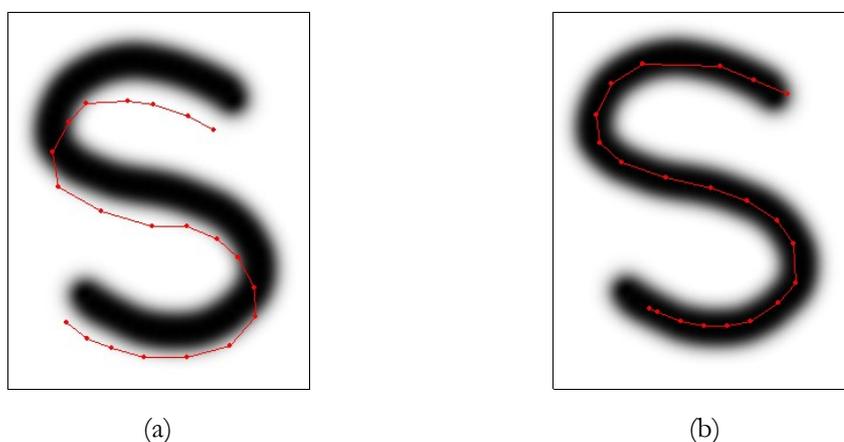
En estos primeros experimentos se utilizaron imágenes simples como la Figura 5.6. Dicha figura presenta una 'S' sobre la que se ha aplicado un suavizado gaussiano. El comportamiento que se esperaba para el *snake* queda reflejado en la Figura 5.7. El *snake* (en rojo), con una forma similar a la del elemento a verificar, se sitúa directamente sobre la imagen en una posición aproximada. La idea inicial consiste en que el *snake*, una vez dispuesto en esta posición inicial, se itere hasta alcanzar un punto de mínima energía. En el momento en que se alcanzase dicho mínimo, se mediría la energía del *snake* para comprobar su situación respecto a un umbral prefijado. Si la energía fuese superior a ese umbral debería rechazarse la firma puesto que la posición de equilibrio no sería la esperada, mientras que si la energía fuese inferior a ese umbral debería aceptarse la firma.

### Problema de la velocidad

Para que el tiempo de cómputo de las múltiples iteraciones del *snake* fuese aceptable se utilizó una optimización del algoritmo de diferencias finitas sobre la propuesta original de Kass y otros. Esta optimización, propuesta por A. A. Amini, T. E. Weymouth y R. C. Jain [182], utiliza técnicas de programación dinámica para minimizar el número de cálculos necesarios en cada iteración. Para ello, impone la limitación de que el cálculo de la energía del *snake* se realice según un orden estricto de los puntos de control, y que el cálculo de la energía asociada a cada punto de control sólo dependa de los puntos de control adyacentes. Esta optimización convierte en polinómica la complejidad del algoritmo, que en su formulación original es exponencial respecto al número de puntos de control.



**Figura 5.6.-** Imagen sintética utilizada en las pruebas: (a) imagen original y (b) tras la aplicación de un filtro gaussiano paso bajo.



**Figura 5.7.-** Ejemplo que muestra cómo se ajusta un *snake* a la figura de una 'S': (a) muestra la posición inicial del *snake*, (b) muestra la posición del *snake* tras iterar hasta converger. El *snake* tiene parámetros  $(\alpha, \beta, \nu) = (0.1, 0.1, -100.0)$

El procedimiento empleado para las primeras pruebas viene detallado por el Algoritmo 5.3, el cual considera dos hipótesis o precondiciones:

- La mayoría de los puntos de la imagen a verificar corresponden a la firma, siendo la imagen de entrada un rectángulo horizontal ajustado a la firma. Esto se deriva de los resultados obtenidos por el algoritmo de segmentación propuesto.

- No existen grandes diferencias entre la inclinación de la línea base de la firma modelo y la de la firma a verificar. Esto es, la imagen de la firma no se encuentra rotada artificialmente, más allá de la variabilidad natural producto de la escritura.

En este algoritmo, la función “Ajustar” consiste en un paso del descenso de gradiente que busca minimizar la energía del *snake*. La función “PosiciónInicial” hace coincidir el centro de masas del *snake* y el centro de masas de la imagen, reescalando el *snake* para que, manteniendo sus proporciones originales, ocupe el máximo espacio posible dentro de la imagen.

#### Algoritmo 5.3 de verificación de firmas mediante snakes

**Nombre:** VerificaConSnake

**Entrada:**  $I_V$  : imagen con la firma a verificar.  
 $I_M$  : imagen modelo.

**Salida:** C : Valor de ajuste del snake a la imagen.

```

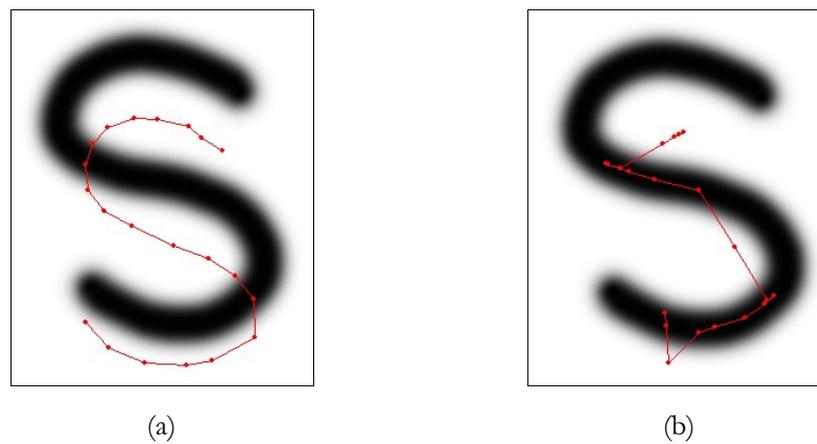
1 - P = UneTrazos(SobresegmentaTrazos((I_M)) //Algoritmo 5.1 y 5.2
2 - P_inicial = PosicionarInicialmente(P, I_V)
3 - P_ajustado = Ajustar(P_inicial, I_V)
4 - while (P_actual ≠ P_anterior)
5 -     P_anterior = P_actual
6 -     P_actual = ajustar(P_actual, I_V)
7 - C = Medir(I_V, P_actual)

```

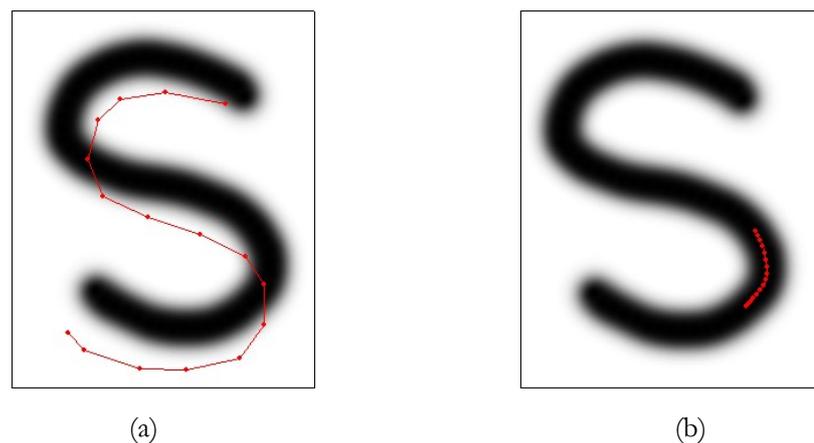
#### Problema de localidad

Los resultados obtenidos por este algoritmo de *snakes* ligeramente modificado no fueron satisfactorios. En primer lugar, se comprobó que, como apuntan varios autores, el algoritmo de convergencia del *snake* cae fácilmente en mínimos locales. Esto motiva que el *snake* rara vez converja a la posición deseada si su inicialización no es lo suficientemente cercana a la firma a verificar (ver ejemplo de la Figura 5.8). Este comportamiento local se deriva de dos hechos: sólo se evalúan un número limitado de vecinos de cada punto de control del *snake*, y la zona de influencia de un punto de la firma sólo llega hasta donde lo haga el filtro paso bajo que se haya aplicado.

Por otro lado, los parámetros para la energía interna del algoritmo original de Kass y otros (elasticidad y rigidez) no se diseñan para conservar la forma inicial del *snake*. El *snake* es un elemento sin memoria que no recuerda su forma inicial. Así, unos valores bajos de  $\alpha$  y  $\beta$  (en la ecuación A.2) lo hacen maleable, mientras que unos valores altos tienden a alinear todos los puntos de control. Debido a ello, en muchas ocasiones, durante el proceso iterativo de minimización de energía el *snake* se deforma totalmente (véase la Figura 5.9). Además, debido a que los puntos de control del *snake* consiguen concentrarse sobre los píxeles oscuros de la imagen y los segmentos entre dichos puntos pueden alinearse y contraerse casi completamente, se obtienen valores de muy baja energía, lo cual invalida, en principio, el uso de la energía como criterio de verificación.



**Figura 5.8.-** Ejemplo que ilustra el problema de la localidad de los *snakes*. Una ligera variación en las condiciones iniciales del *snake* de la Figura 5.7 lo hace converger a una posición no deseada. La Figura (a) muestra la posición inicial del *snake*, y (b) muestra la posición del *snake* tras iterar hasta su convergencia.



**Figura 5.9.-** Ejemplo que ilustra el problema de la ausencia de memoria de los *snakes*. De nuevo una ligera variación en las condiciones iniciales del *snake* de la Figura 5.7 lo hace converger a una posición no deseada. La Figura (a) muestra la posición inicial del *snake* y (b) muestra la posición del *snake* tras iterar hasta converger perdiendo totalmente su forma inicial.

### 5.3.2 Modificaciones adicionales a la propuesta original

Debido a los problemas que se encontraron utilizando la propuesta original de *snakes* se han realizado una serie de modificaciones sobre las funciones de energía, las cuales se describen en las siguientes subsecciones.

### Uso del mapa de potencial para la energía externa

Para resolver la dificultad asociada a la falta de localidad del *snake*, primeramente se pensó en aplicar un filtrado gaussiano con una ventana de mayor tamaño. Pronto se comprendió que lo que se necesitaba era llevar al límite esa idea, es decir, asociar a cada píxel no activo de la imagen su distancia euclídea al píxel activo más cercano. De esta forma, en cualquier punto de la imagen existiría un gradiente hacia los píxeles activos, que por hipótesis son los que pertenecen a la firma. De manera simultánea, mediante una revisión bibliográfica, se descubrió que Cohen y Cohen [183] ya habían propuesto esta idea y la habían denominado *mapa de potencial*. (*potential map*).

Formalmente, sea  $I$  la imagen con la firma a verificar y  $\mathcal{A}$  el conjunto de píxeles activos de  $I$ , el mapa de potencial se define mediante la matriz bidimensional:

$$m_{imagen}(x, y) = \min \{ \sqrt{(x - p_x)^2 + (y - p_y)^2} \mid \forall (p_x, p_y) \in \mathcal{A} \wedge \forall (x, y) \in I \} \quad (5.2)$$

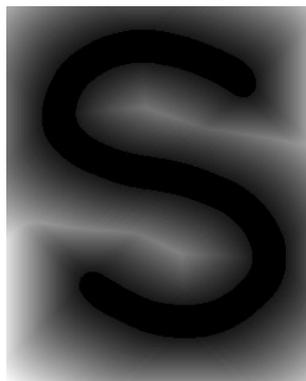
### Algoritmo de cálculo del mapa de potencial bidimensional

La ecuación 5.2 no se debe utilizar directamente para el cálculo de la matriz  $m_{imagen}(x,y)$  debido a la complejidad del algoritmo resultante. Por ello, se propone un algoritmo que reduce la complejidad del cálculo. En el algoritmo propuesto se comienza asignando valor 0 a los píxeles activos y se van asignando paulatinamente valores de distancia a sus vecinos. Así, el valor de distancia para un punto del mapa  $m_{imagen}(x,y)$  es igual al valor ya calculado para alguno de sus vecinos incrementado en una unidad. De esta forma, el algoritmo calcula los valores de distancia partiendo de los píxeles adyacentes a los activos y avanzando en capas hacia los límites de la imagen.

En la Figura 5.10 se aprecia el resultado del cálculo del mapa de potencial para la imagen de la 'S'. En esta figura, el nivel de intensidad asociado a cada punto de la imagen representa la distancia al punto de la firma más cercano. Nótese la diferencia entre dos píxeles adyacentes siempre es la unidad. Utilizando este mapa de potencial el término de la energía externa de la ecuación 5.1 se modifica según:

$$E_{externa}(v_i) = E_{mapa\_potencial}(v_i) = m_{imagen}(v_{i-1}) + m_{imagen}(v_i) + m_{imagen}(v_{i+1}) \quad \forall i = 2 \dots N-1 \quad (5.3)$$

Obsérvese que en esta formulación se tiene en cuenta las energías de los puntos adyacentes a cada punto de control del *snake*. Esto se hace para evitar la inmovilidad de los extremos del *snake*.



**Figura 5.10.-** Resultado del cálculo del mapa de potencial sobre la imagen de la 'S'. El nivel de intensidad de gris, que se ha ecualizado para resaltarlo, indica la proximidad a los píxeles activos.

**Algoritmo 5.4 de creación del mapa de potencial**

**Nombre:** DistanciaEnMapaDePotencial

**Entradas:** I : Imagen bitonal correspondientes a una firma

**Salida:** M : Matriz de enteros de las mismas dimensiones que I

```

1 -      n = 0
2 -      P = []
3 -      M = IniciarACero()
4 -      for each i in I {
5 -          if (i == 0) {
6 -              P.Insertar(i)
7 -          while( p ≠ [] )
8 -              for each p in P
9 -                  M(p) = n
10 -                 n = n + 1
11 -                 V = vecinos(P, I)
12 -                 P = []
13 -                 for each v in V
14 -                     if (M(v) == 0)
15 -                         P.Insertar(v)

```

Las pruebas han permitido comprobar que la introducción del mapa de potencial ayuda a paliar el problema de la localidad. Sin embargo, el problema de la ausencia de memoria del *snake* sigue impidiendo el comportamiento deseado para éste. El *snake*, en la Figura 5.11, partiendo de su inicialización (a), tras 50 iteraciones (b) se aproxima a la solución deseada. Sin embargo, como se puede apreciar en las figuras (c), (d) y (e), el proceso de convergencia continua, fomentado por las fuerzas elásticas de la propuesta original (ecuación A.2). Finalmente, en la figura (f) el *snake* queda concentrado en un punto.

**Propuesta de energía interna con memoria**

Se ha comprobado que el *snake* pierde su forma original durante las sucesivas iteraciones que se producen cuando su posición inicial no está cerca de la forma a la que se ajusta. Con objeto de que esto no ocurra, se ha propuesto una modificación de la energía interna original de Kass y otros. Esta modificación sustituye los factores de elasticidad y rigidez asociados al *snake* por el de elasticidad de un elemento con memoria dando lugar a lo que denominamos *snakes con memoria* (*memory shape snakes*).

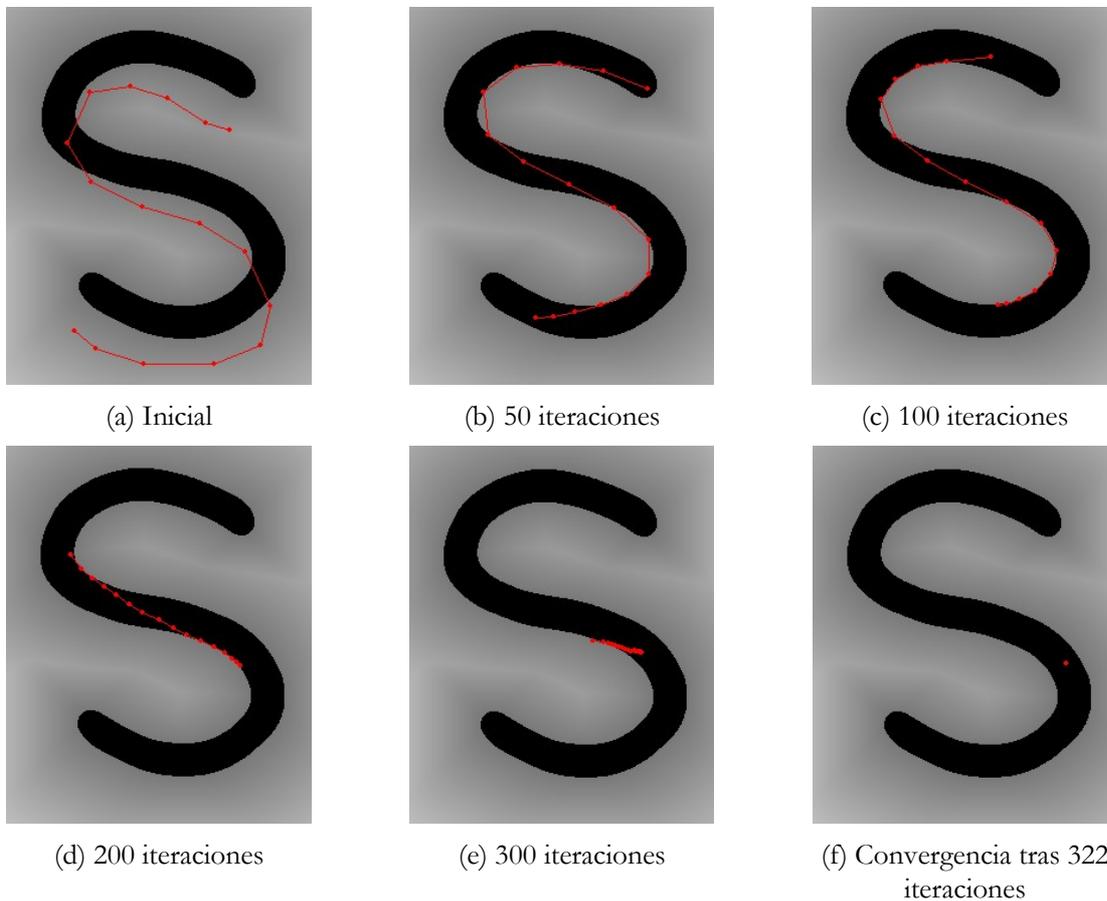
Como el *snake*, en este caso, consiste en una línea poligonal abierta, para preservar su forma basta con mantener constantes, respecto a su inicialización, los ángulos y las proporciones entre cada par de segmentos adyacentes. Sin embargo, el mantenimiento estricto de estos ángulos y estas proporciones daría lugar a un comportamiento tan rígido del *snake* que imposibilitaría adaptación alguna. Para evitar este comportamiento indeseado la energía interna se define como la suma de dos

energías. Una que restringe la variación excesiva de los ángulos entre segmentos del *snake*, y otra que hace lo mismo con la variación de las proporciones. En consecuencia, el término de energía interna queda:

$$E_{interna} = E_{memoria} = E_{ángulo} + E_{propor} \quad (5.4)$$

Debe tenerse cuidado con la definición de estos términos de energía pues, para que siga siendo válido el enfoque de programación dinámica comentado en 5.3.1, el cómputo de dicha energía para cada punto de control no debe depender más que de los puntos de control adyacentes o de valores constantes.

Este tipo de alteraciones en la definición de energía es habitual cuando se trabaja con *snakes*. Así, por ejemplo, el uso de restricciones que consideran el ángulo entre segmentos adyacentes con respecto al ángulo de inicialización del *snake* ya se ha utilizado con éxito en un trabajo de Davison y otros [178]. Sin embargo, no hemos encontrado ninguna referencia al uso de restricciones en las proporciones de los segmentos respecto a su inicialización.



**Figura 5.11.-** Ejemplo de movimiento del *snake* cuando se utiliza un mapa de potencial.

### Restricciones respecto al ángulo entre segmentos adyacentes

A cada vértice de la línea poligonal que define el *snake*, se le puede hacer corresponder el ángulo formado por los dos segmentos precedentes. Nótese que deben excluirse el primer vértice y el último pues no tienen segmento adyacente con el que definir ángulo alguno.

$$\text{ángulo}(v_i) = \arccos\left(\frac{(v_i - v_{i-1}) \cdot (v_{i+1} - v_i)}{\|v_i - v_{i-1}\| \|v_{i+1} - v_i\|}\right) \quad \forall i = 2 \dots N-1 \quad (5.5)$$

En cada instante de tiempo  $t$  la variación del ángulo relativo a cada vértice respecto al ángulo que existía en el momento de inicialización del *snake* viene dado por:

$$\delta = |\text{ángulo}(v_i(0)) - \text{ángulo}(v_i(t))| \quad \forall i = 2 \dots N-1 \quad (5.6)$$

En la formulación de Davison [178] se define la energía de rigidez según la ecuación 5.7 donde  $k_r$  es una constante que permite variar el grado de rigidez. Encontramos que esta formulación no es adecuada para nuestro caso ya que el cambio de ángulo no tiene límite, lo que en principio le permite su adaptación a cualquier figura. Esto se debe a que el *snake* de Davison no persigue verificar una forma, sino adaptarse a ella.

$$E_{\text{rigidez}}(v_i(t)) = k_r \sum_{i=0}^{N-1} \left( \frac{\text{ángulo}(v_i(0)) - \text{ángulo}(v_i(t))}{\text{ángulo}(v_i(0))} \right)^2 \quad \forall i = 2 \dots N-1 \quad (5.7)$$

Por contra, en este trabajo, para restringir la variación del ángulo se define, asociado a cada vértice, un término de energía con una formulación tipo resorte. Esta formulación depende de un parámetro  $k_{\text{ángulo}}$  de elasticidad (la constante del resorte) y de un umbral  $U_{\text{ángulo}}$  que impide distensiones más allá de un límite estricto respecto al ángulo inicial. Formalmente:

$$E_{\text{ángulo}}(v_i(t)) = \begin{cases} \infty, & \text{si } \delta > U_{\text{ángulo}} \\ \delta \cdot k_{\text{ángulo}}, & \text{si } \delta \leq U_{\text{ángulo}} \end{cases} \quad \forall i = 2 \dots N-1 \quad (5.8)$$

siendo la energía total  $E_{\text{ángulo}}$  en un instante  $t$ , el resultado de la suma de la energía correspondiente a cada vértice en ese instante  $t$ . Por lo tanto:

$$E_{\text{ángulo}}(t) = \sum_{i=2}^{N-1} E_{\text{ángulo}}(v_i(t)) \quad (5.9)$$

### Restricciones respecto a las proporciones entre segmentos adyacentes

De nuevo, a cada vértice de la línea poligonal que define el *snake*, se le puede hacer corresponder la proporción existente entre los dos segmentos adyacentes.

$$\text{propr}(v_i) = \frac{\|v_i - v_{i+1}\|}{\|v_{i-1} - v_i\|} \quad \forall i = 2 \dots N-1 \quad (5.10)$$

En cada instante de tiempo  $t$ , la variación de las proporciones relativas a cada vértice respecto a los valores existentes en el momento de inicialización del *snake* viene dada por:

$$\phi = \frac{\text{propr}(v_i(t))}{\text{propr}(v_i(0))} \quad \forall i = 2 \dots N-1 \quad (5.11)$$

Nótese que esta definición no es compatible con que se definan dos vértices sobre el mismo punto. De nuevo, deben excluirse el primer y el último vértice pues no tienen más que un segmento adyacente.

Finalmente, para restringir la variación de las proporciones, asociada a cada vértice, se define otro término de energía con formulación tipo resorte. Esta formulación depende de un parámetro  $k_{prop}$  de elasticidad (la constante del resorte) y un umbral  $U_{prop}$  que impide distensiones más allá de un límite estricto respecto a las proporciones iniciales. Formalmente:

$$E_{prop}(v_i(t)) = \begin{cases} \phi \cdot k_{prop} & , \quad \text{si } U_{prop} - 1 \geq \phi \geq U_{prop} + 1 \\ \infty & , \quad \text{en otro caso} \end{cases} \quad \forall i = 2 \dots N-1 \quad (5.12)$$

siendo la energía total  $E_{prop}$  en un instante  $t$ , el resultado de la suma de la energía correspondiente a cada vértice en ese instante  $t$ :

$$E_{prop}(t) = \sum_{i=2}^{N-1} E_{prop}(v_i(t)) \quad (5.13)$$

### Primeros ensayos con firmas

Los resultados obtenidos tras estas modificaciones fueron satisfactorios. Así, los *snakes* convergen si su localización inicial no está demasiado alejada de la localización deseada. Además, en ausencia de ruido, la localización inicial, basada en la coincidencia de los centros de masas del *snake* y la imagen, es suficiente para obtener un resultado satisfactorio sobre imágenes sencillas.

Estos resultados animaron a realizar pruebas con imágenes de firmas reales, las cuales definen siluetas con mayor complejidad que la considerada en los ejemplos precedentes.

Ya se ha comentado que algunas firmas están compuestas de segmentos inconexos. Esto hace que algunos puntos de control no deban computarse en el cálculo de energía de la ecuación 5.3 debido a que no deben coincidir con el trazo de la firma. Para contemplar esta situación, en el proceso de definición del *snake*, ya se realizó una distinción entre segmentos dentro de trazo y segmentos fuera de trazo. Ahora, además, se cambia la ecuación 5.3 para tener en cuenta este hecho:

$$E_{externa}(v_i) = \begin{cases} E_{mapa\_potencial}(v_i) & , \quad \text{si } v_i(0) \in \mathcal{A} \\ 0 & , \quad \text{si } v_i(0) \notin \mathcal{A} \end{cases} \quad (5.14)$$

siendo  $\mathcal{A}$  el conjunto de píxeles activos de la imagen modelo.

La Figura 5.12 muestra un ejemplo en el que se sitúa el *snake* definido en la Figura 5.8 sobre una imagen a verificar y se itera hasta lograr la convergencia. Al igual que ya se hizo en la Figura 5.5, los segmentos interiores a trazos se representan en rojo y los segmentos fuera de trazo en blanco.

Utilizando la muestra  $\mathcal{A}$  se han obtenido los valores de energía recogidos en la Figura 5.13. En la figura se pueden apreciar las diferencias en las distribuciones de energía para las dos clases: la de firmas a aceptar y la de firmas a rechazar. Realizando pruebas sobre el umbral de energía a partir del cual se deberían aceptar o rechazar las firmas se encuentra un *EER* del 17.8% para un valor de energía de 4.65.

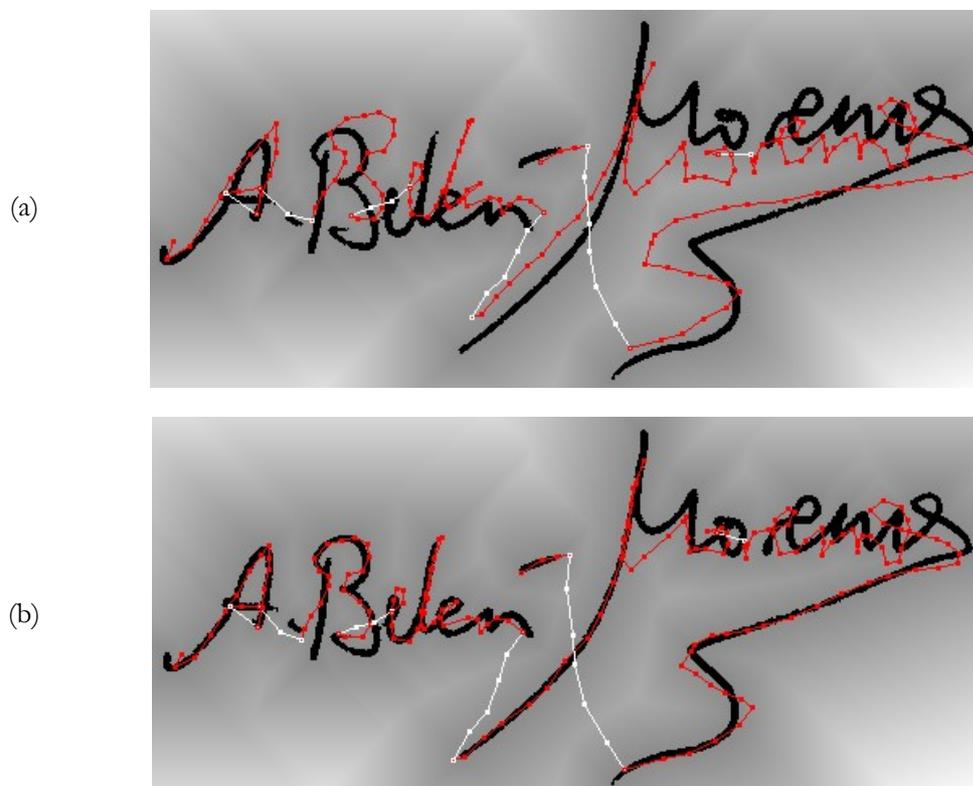
### 5.3.3 Propuesta del uso del ángulo de los trazos como dimensión adicional

A pesar de que los resultados obtenidos por las modificaciones presentadas producen resultados satisfactorios en muchas situaciones, el algoritmo propuesto no es capaz de resolver aquellas en las que el *snake* tiene que atravesar zonas amplias de píxeles activos hasta alcanzar el encaje deseado.

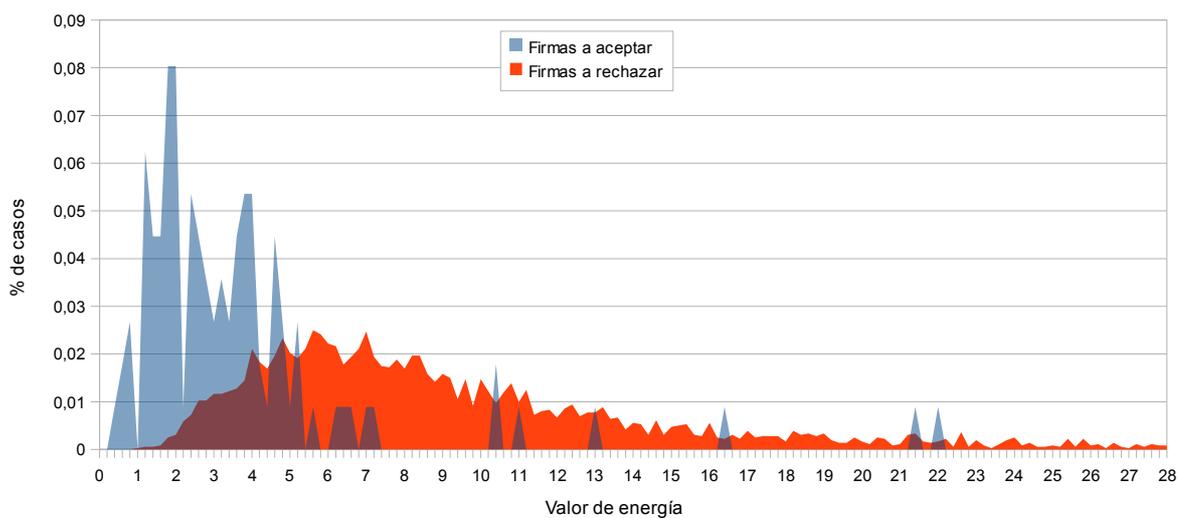
La Figura 5.14 (a) muestra un ejemplo que ilustra este tipo de situaciones. En ella vemos como el *snake* está definido inicialmente como una línea horizontal roja sobre una imagen que tiene dos

líneas que se cruzan: una horizontal (como el *snake*) y otra diagonal. Las posibilidades de comportamiento que se prevén para el algoritmo de *snakes* modificado son dos:

1. El *snake* gira y se ajusta a la línea diagonal.
2. El *snake* se desplaza y finalmente se ajusta a la línea horizontal.



**Figura 5.12.-** Ejemplo de funcionamiento del algoritmo modificado de *snakes*: (a) presenta la posición inicial del *snake* sobre imagen a verificar y (b) una vez que ha convergido tras 20 iteraciones.



**Figura 5.13.-** Energía obtenida para las firmas de la muestra A por el *snake* con memoria.

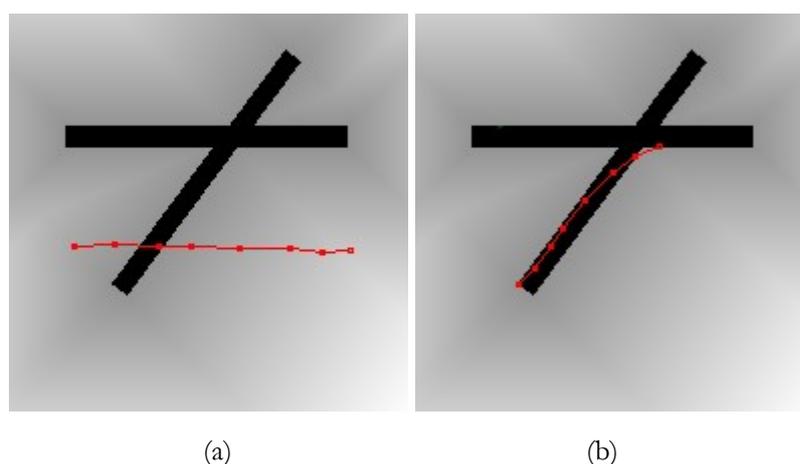
Una de las hipótesis de partida del Algoritmo 5.3 era que el *snake* se encontraba con una inclinación similar a la de la figura con la que se desea encajar. Por ello, el comportamiento deseado sería el que describe la segunda posibilidad. Sin embargo, como se aprecia en la Figura 5.14 (b), el algoritmo de *snakes* modificado encuentra un camino que lo lleva a la primera posibilidad y allí encuentra un mínimo local.

En el caso del ajuste de un *snake* a una firma, esta situación puede encontrarse en, al menos, dos supuestos:

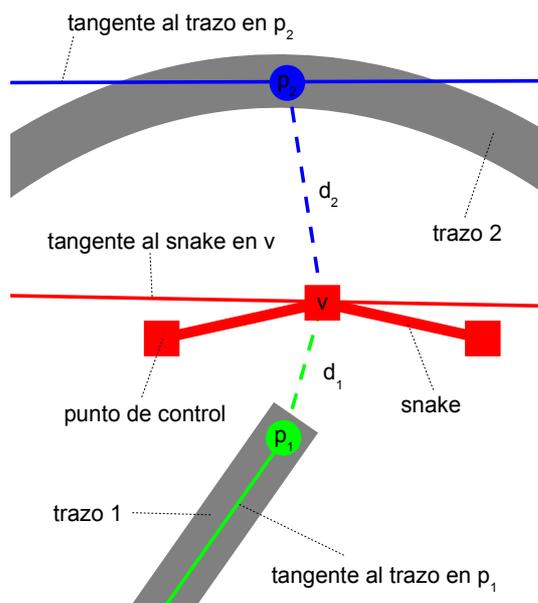
1. Entre el *snake* y el punto deseado de ajuste hay otras partes de la firma debido a una inicialización poco adecuada.
2. Entre el *snake* y el punto deseado de ajuste hay ruido con estructura.

Obsérvese que, en el ejemplo considerado, los segmentos del *snake* tienen una orientación horizontal de aproximadamente  $0^\circ$ . Mientras, algunos de los píxeles activos de la imagen están incluidos en un trazo de orientación  $0^\circ$  y otros en un trazo de orientación  $60^\circ$ . Los puntos de control del *snake* se encuentran más cerca del trazo diagonal que del horizontal (de hecho en ocasiones lo tocan), pero el valor del ángulo asociado al *snake* es igual al valor del ángulo asociado al trazo horizontal. Otra aportación de esta Tesis, consiste en la inclusión del ángulo de la tangente a cada punto de un trazo como una dimensión adicional en la definición de energía del *snake*. De esta forma, en el ejemplo de la Figura 5.14 (a), el *snake* podría encontrar más cercano el trazo horizontal que el inclinado, pues comparte con él el valor de una de sus dimensiones.

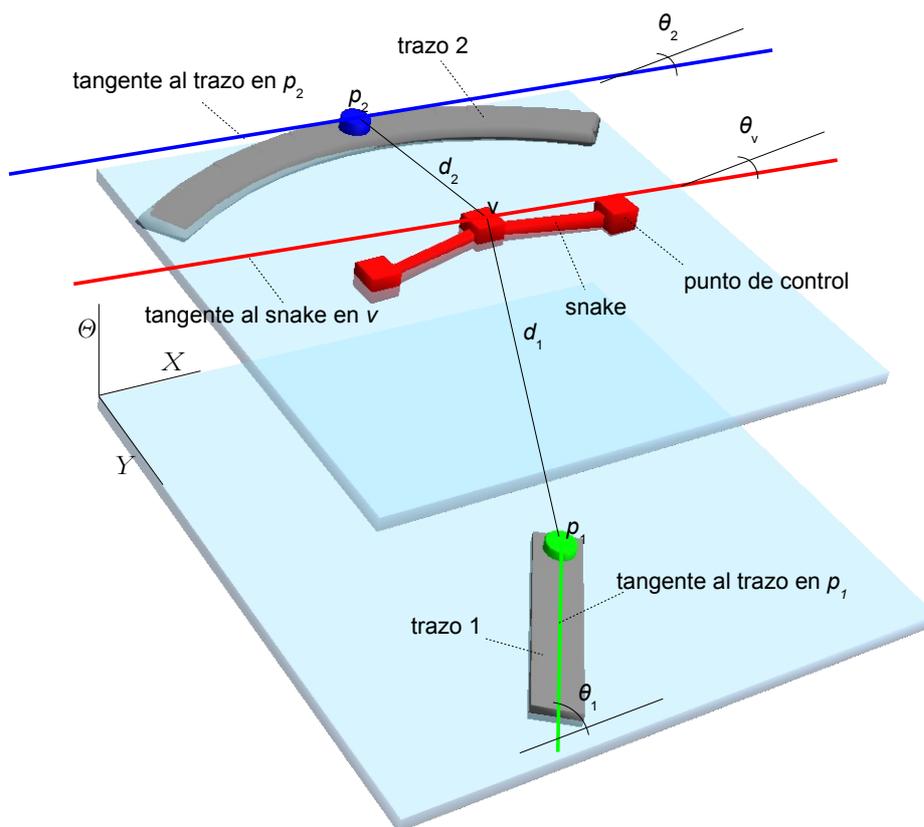
En la Figura 5.15 se presenta un ejemplo en el que un punto de control  $v$  del *snake* se encuentra entre dos trazos de una imagen. Dentro de la imagen, el punto de control  $v$  se halla más cerca de uno de los segmentos. Sin embargo, el *snake* tiene una disposición más parecida a la del otro segmento que se encuentra más alejado dentro de la imagen. En la Figura 5.16 se representa el cambio sobre esta situación que introduce una nueva dimensión que tiene en cuenta el ángulo de la tangente a los puntos del trazo y a los puntos de control. En la misma, se puede apreciar como ahora el punto de control  $v$  se ha alejado del trazo que tiene una disposición contraria al *snake*.



**Figura 5.14.-** Ejemplo de error en el ajuste del *snake*: (a) disposición inicial del *snake* sobre la imagen y (b) posición del *snake* tras finalizar el proceso de convergencia.

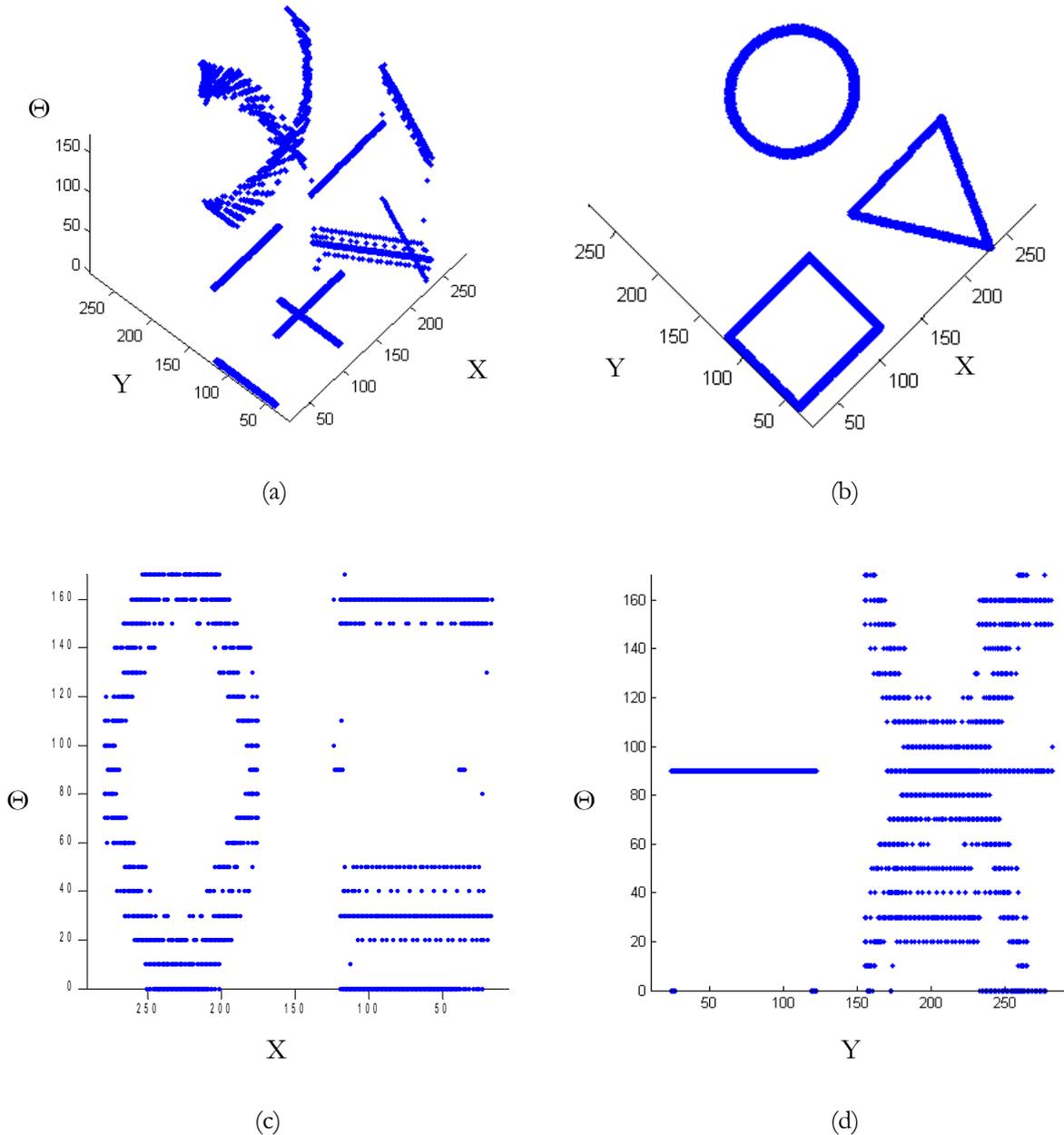


**Figura 5.15.-** Ejemplo de distancia en un mapa de potencial bidimensional. Como sólo tiene en cuenta las coordenadas  $x$  e  $y$ ,  $v$  está más cerca de  $p_1$  que de cualquier otro punto de los trazos, y por ello se crea un gradiente desde  $v$  a  $p_1$ .



**Figura 5.16.-** Ejemplo de distancia en un mapa de potencial que considera el ángulo de la tangente al trazo en cada punto además de las coordenadas  $x$  e  $y$ . Según este mapa,  $v$  está más cerca de  $p_2$  que de cualquier otro punto de los trazos. Esto determina que se cree un gradiente desde  $v$  a  $p_2$ .

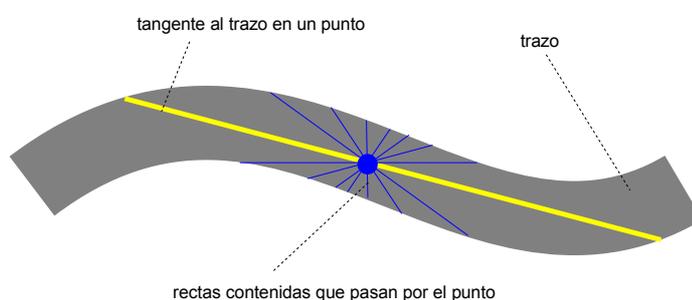
En la Figura 5.17 se presenta, mediante una figura tridimensional axonométrica y algunas de sus proyecciones, un ejemplo de una imagen con varias figuras geométricas. Como el eje vertical corresponde a la dimensión del ángulo tangencial, la proyección (b) corresponde a la imagen original. Aquí se puede apreciar cómo la nueva dimensión introducida supone un incremento del espacio de movimiento, lo cual permitirá al *snake* moverse con mayor libertad, evitando interacciones indeseadas con otros elementos de la imagen que no compartan su orientación.



**Figura 5.17.-** Representación tridimensional de una imagen bidimensional tras la inclusión de la dimensión tangencial: la Figura (a) es la representación axonométrica de la imagen plana (b). El eje vertical corresponde al ángulo, por lo que (b) corresponde a la proyección sobre el suelo de (a). Las Figuras (c) y (d) corresponden a las proyecciones izquierda y derecha de (a).

Así, la idea de incluir el ángulo de la tangente a los trazos como una tercera dimensión, se concreta en los siguientes aspectos:

- A cada píxel activo de la imagen a verificar  $(x, y)$  se le asocia un valor  $\theta_A(x, y)$  igual al ángulo de la recta tangente aproximada al trazo en el que está contenido. Este ángulo se obtiene del segmento que, pasando por el punto  $(x, y)$ , mantiene mayor longitud dentro del trazo de manera ininterrumpida (ver figura 5.18).
- A cada vértice  $v_i$  de la línea poligonal que forma el *snake* se le asocia un valor  $\theta_s(v_i)$  determinado por los ángulos de los segmentos anterior y posterior al vértice. Obsérvese que se sigue respetando la condición impuesta por en el apartado 5.3.1.
- El mapa de potencial pasa de ser una superficie bidimensional  $XY$  a ser un espacio tridimensional  $XY\theta$ , donde  $\theta \in \Theta$  varía de manera discreta entre  $0^\circ$  y  $180^\circ$ . En este espacio, cada punto está etiquetado con la distancia al punto activo más cercano (ver Figura 5.16).



**Figura 5.18.-** Ejemplo de la determinación de la recta tangente a un punto  $p$  del trazo. Se obtiene de manera aproximada mediante la tangente del segmento que pasando por el punto  $p$  mantiene mayor longitud de manera ininterrumpida dentro del trazo.

En el cálculo de un mapa de potencial tridimensional aparecen dos problemas. El primero se deriva de la elección de la escala para la dimensión relativa al ángulo y el segundo se deriva del aumento del tiempo de cálculo respecto al enfoque previo. A continuación se analizan estos problemas.

### Distancia en la dimensión del ángulo tangencial

Se acepta que una imagen digital constituye un espacio métrico donde la distancia entre sus puntos puede ser definida de diferentes formas (por ejemplo con la distancia euclídea). Sin embargo, al introducir la componente relativa a la distancia angular el espacio pierde esta propiedad. Esto se debe a que no es posible establecer un orden entre dos ángulos cualesquiera. Por ejemplo, es evidente que hablando de números naturales se cumple que  $0 < 90 < 180$ , pero no es cierto que  $0^\circ < 90^\circ < 180^\circ$  pues  $0^\circ = 180^\circ$ . Así, se podría establecer que el ángulo  $0^\circ$  sea menor que el ángulo  $90^\circ$ , el cual a su vez sería menor que  $180^\circ$ , pero dos segmentos que forman  $180^\circ$  o  $0^\circ$  están alineados, por lo que corresponden al mismo ángulo.

Obsérvese que en la figura 5.17(a) se produce un salto abrupto entre las dos espirales que forman el círculo. Este salto abrupto desaparecería si se hiciese coincidir el valor superior del ángulo ( $180^\circ$ ) con el inferior ( $0^\circ$ ). Esta operación no se realiza en la figura porque el resultado sería difícil de visualizar, ya que el eje vertical tendría que ser un círculo (y el resultado un hipercilindro).

Con objeto de que la elección de un ángulo con valor  $\theta$  ó  $\theta+180^\circ$  no afecte al cálculo de la distancia entre dos valores, la distancia del ángulo tangencial entre dos valores angulares se ha definido mediante la siguiente ecuación:

$$d_{\text{tangencial}}(\theta_1, \theta_2) = \min(\min(\theta_1, \theta_2) + 180 - \max(\theta_1, \theta_2), \max(\theta_1, \theta_2) - \min(\theta_1, \theta_2)) \quad (5.15)$$

Por otro lado, debe tenerse en cuenta el problema de la escala de la dimensión angular. En el mapa de potencial que sólo tienen en cuenta las coordenadas  $x$  e  $y$  de los puntos activos de la imagen, como ambas coordenadas usan las mismas unidades, no se planteó que una dimensión tuviese más peso que otra. Sin embargo, el ángulo está definido en unidades de diferente tipo a las coordenadas espaciales. Por ello, obligatoriamente se debe elegir una escala que determinará el peso del ángulo en el cálculo de distancias. Si este factor de escala es grande con respecto a las de las otras dimensiones, significará que el ángulo influye mucho en el cálculo de la distancia y por lo tanto en el movimiento del *snake*. Por otro lado, si el factor de escala es pequeño la introducción de esta dimensión adicional apenas afectará al movimiento del *snake*.

### Algoritmo de cálculo del mapa de potencial

El aumento de las dimensiones del *snake* aumenta la complejidad del Algoritmo 5.4 que pasa de  $O(n^2)$  a  $O(n^3)$ , pues la función “Vecinos” de este algoritmo pasa de devolver 8 vecinos a devolver 26. Este cambio hace inviable su aplicación a problemas prácticos que requieran rapidez en la obtención del resultado usando medios computacionales modestos. Por ello, se plantea el Algoritmo 5.5 para el cálculo de distancias, el cual no requiere el cálculo previo de todos los valores del mapa de potencial tridimensional, sino que los calcula de manera eficiente según se precisan.

#### Algoritmo 5.5 de cálculo de distancia en un mapa de potencial 3D

**Nombre:** DistanciaAlMapa3D

**Entradas:** I : imagen con un conjunto A de píxeles activos  
x, y,  $\theta$  : punto p del mapa de potencial

**Salida:** distancia : Distancia de punto p al punto más cercano de A

```

1 -      static
2 -          A = ObtenerPíxelesActivos(I)
3 -          for each a in A
4 -              mapa(a.x, a.y, a. $\theta$ ) = 0
5 -          if (mapa(x, y,  $\theta$ )  $\neq$  null)
6 -              return mapa(x, y,  $\theta$ )
7 -          else
8 -              mapa(x, y,  $\theta$ ) =  $\infty$ 
9 -          for each a in A
10 -              mapa(x, y,  $\theta$ ) = min (mapa(x, y,  $\theta$ ) , Distancia((x, y,  $\theta$ ), (a.x, a.y, a. $\theta$ ))
11 -          distancia = mapa(x, y,  $\theta$ ).
```

Paradójicamente, el cálculo de todos los puntos, del cubo que formaría el mapa de potencial, utilizando este algoritmo sería más costoso que el mismo cálculo usando el Algoritmo 5.3. Sin embargo, como el *snake* en sus iteraciones no recorre todos los puntos del mapa de potencial, el resultado es un incremento efectivo de la velocidad del proceso iterativo de convergencia del *snake*.

Obsérvese que el Algoritmo 5.5 consta de una parte estática que solo se calcular en la primera ejecución, y cuyos resultados se aprovechan en las invocaciones necesarias para calcular la distancia desde cada punto.

### Algoritmo optimizado de cálculo del mapa de potencial

El Algoritmo 5.5 se puede optimizar si, en vez de calcular la distancia del punto  $P$  a todos los píxeles activos  $\mathcal{A}$ , se calculan sólo un subconjunto de dichas distancias. Para ello, se ha procedido a agrupar los píxeles de  $\mathcal{A}$  utilizando una estructura bidimensional (o cuadrícula). De cada cuadro de esta cuadrícula se almacena un representante y el punto contenido en el cuadro que más se desvía del representante (máxima desviación de los puntos contenidos en el cuadro respecto a su representante). Luego, para obtener el potencial de un punto  $P$  se ordenan las cuadrículas según la distancia de cada representante a  $P$ . Finalmente, si la cuadrícula más cercana a  $P$  es  $C$ , se evalúan las distancias de  $P$  a los píxeles de  $C$  y a los píxeles contenidos en las cuadrículas cercanas a  $C$ . Entendiendo como cercanas aquellas cuadrículas cuya desviación, sumada a la distancia a su representante, sea menor que la distancia a  $C$ , más la desviación de  $C$ . Suponiendo que los diferentes grupos tienen en media el mismo número de elementos, esta modificación divide el número de iteraciones del paso 9 del Algoritmo 5.5 entre el número de grupos.

El Algoritmo 5.6 formaliza esta descripción. En él, la función “Distancia” corresponde a la del Algoritmo 5.5.

#### Algoritmo 5.6 optimizado de cálculo de distancia en un mapa de potencial 3D

**Nombre:** DistanciaOptimizadaAlMapa3D

**Entradas:**  $I$  : imagen con un conjunto  $A$  de píxeles activos  
 $x, y, \theta$  : punto  $p$  del mapa de potencial

**Salida:** distancia : Distancia de punto  $P$  al punto más cercano de  $A$

```

1 -      static
2 -          cents = HallarCentroidesRejilla(10,10,I)
3 -          desvs = HallarDesviacionesRejilla(10,10,I)
4 -          cent_candidato = HallarCentroideMasProximo(P,cents,I)
5 -          L = [cent_candidato]
6 -          distancia_candidata = Distancia(cent_candidato,I)
7 -          for each (c in cents)
8 -              if (Distancia(c,p) < distancia_candidata + desvs(cent_candidato))
9 -                  L = c::L
10 -          distancia = ∞
11 -          for each (c in L)
12 -              for each (p_aux in c)
13 -                  distancia = min(Distancia(p_aux,p),distancia_candidata)
    
```

### Cálculo de la energía

El mapa de potencial definido en 5.2 y utilizado en la ecuación 5.3 se modifica según la ecuación:

$$m_{imagen}(v, \theta) = \min \left\{ \sqrt{(v_x - p_x)^2 + (v_y - p_y)^2 + (k_\theta d_{tangencial}(\theta, \theta_A(v_x, v_y)))^2} \right\},$$

$$\forall (p_x, p_y) \in A \wedge \forall (v_x, v_y) = v \in I \wedge \forall 0 \leq \theta < \pi \quad (5.16)$$

siendo  $I$  la imagen con la firma a verificar,  $A$  el conjunto de píxeles activos de  $I$ ,  $\theta_A(x, y)$  el ángulo de la recta tangente aproximada al trazo en el que está contenido cada punto  $(x, y)$  de  $A$ ,  $k_\theta$  una constante de escala, y  $d_{tangencial}$  la función definida en la ecuación 5.15.

Este mapa de energía se utiliza para una nueva definición de energía externa dada por:

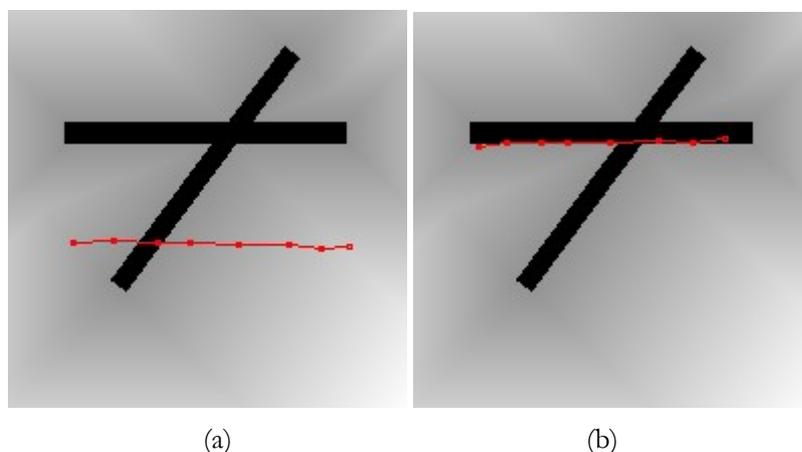
$$E_{externa} = E_{mapa\_potencial} =$$

$$= m_{imagen}(v_{i-1}, \theta_s(v_{i-1})) + m_{imagen}(v_i, \theta_s(v_i)) + m_{imagen}(v_{i+1}, \theta_s(v_{i+1})) \quad (5.17)$$

### Algunos experimentos usando la dimensión del ángulo de las tangentes

Los resultados obtenidos tras la introducción de la dimensión angular en el mapa de potencial fueron positivos. En los experimentos, sobre la muestra A, se usaron varios factores de escala para la dimensión angular, optándose finalmente por  $k_\theta=10$ . Se aprecia que el *snake* se vuelve más robusto en su aproximación a elementos de la imagen con su misma inclinación, reduciéndose la influencia de zonas activas intermedias con inclinaciones diferentes.

Por ejemplo, se puede comprobar la aproximación obtenida usando el mapa de potencial bidimensional (véase la Figura 5.14) con la obtenida con el mapa de potencial tridimensional (Figura 5.19). Se produce una mejora respecto al problema de la localidad, el *snake* se vuelve menos local.

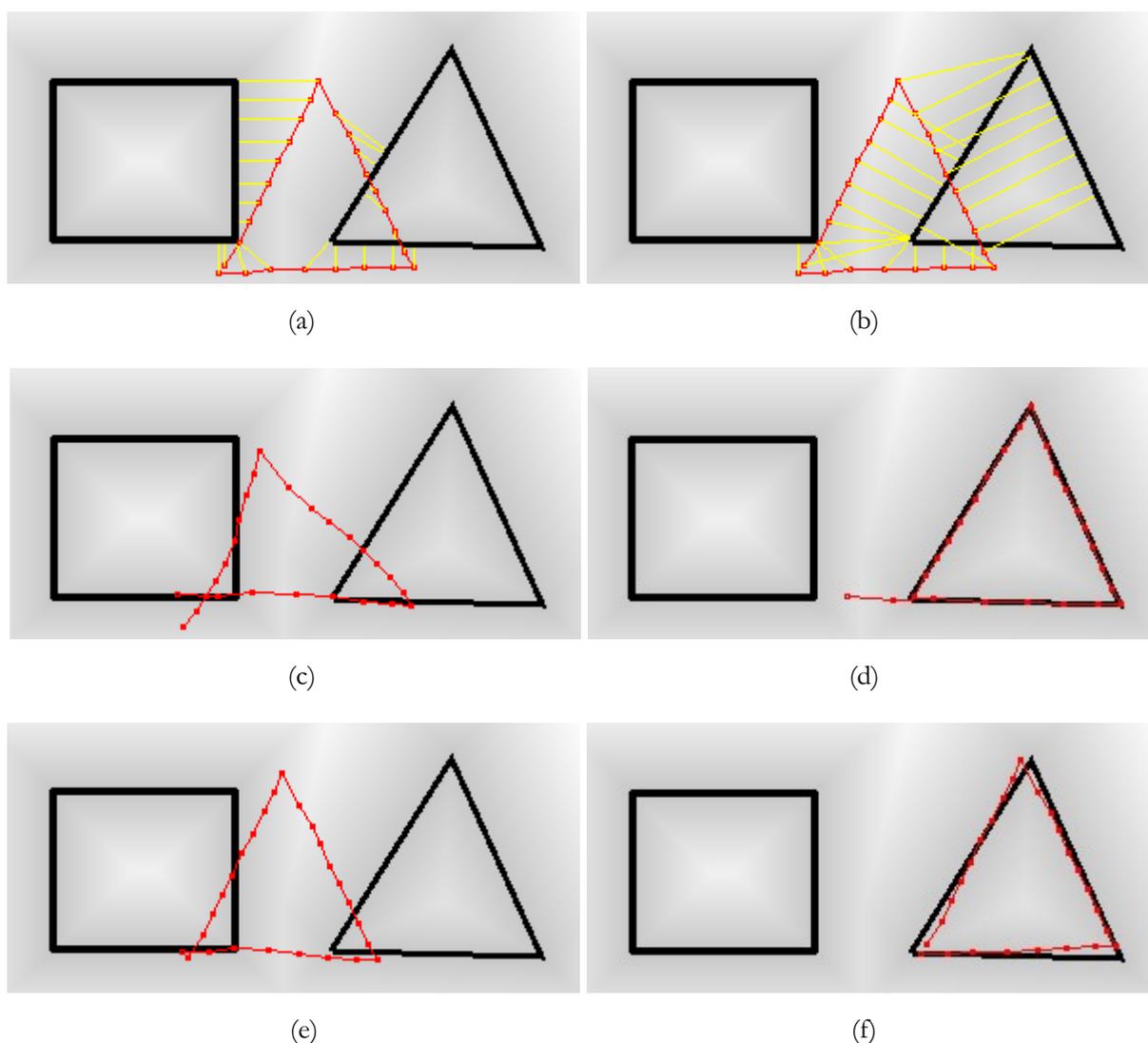


**Figura 5.19.-** Ejemplo de acierto en el ajuste del *snake* cuando se introduce la dimensión angular: (a) disposición inicial del *snake* sobre la imagen, y (b) posición del *snake* tras finalizar el proceso de convergencia.

La Figura 5.20 muestra otro ejemplo donde un *snake*, en rojo y con forma de triángulo, se encuentra posicionado entre la imagen de un cuadrado y la de otro triángulo. En particular, en las Figuras 5.20(a) y (b) se han señalado en amarillo los puntos activos de la imagen más cercanos a cada punto de control del *snake* según las dos definiciones de distancia: una (Figura 5.20(a)) basada en el mapa de potencial bidimensional y otra (Figura 5.20(b)) basada en el mapa tridimensional. Al iterar la

Figura 5.20(a), si se permite un cambio de proporciones del 10% se obtiene el resultado (c) y si se restringe el cambio de proporciones al 1% se obtiene (e). Por otro lado, al iterar la Figura 5.20(b), si se permite un cambio de proporciones del 10% se obtiene (d) y si se restringe el cambio de proporciones al 1% se obtiene (f).

Este ejemplo muestra que el *snake* que utiliza la tercera dimensión, basada en el ángulo de la tangente a los trazos, mejora los resultados respecto al *snake* que utiliza sólo las dos dimensiones espaciales propias de la imagen. Se aprecia que cuando se usa esta tercera dimensión, el *snake* consigue encajar de manera casi óptima sobre la figura con la que guarda parecido (el triángulo). Mientras que cuando el *snake* sólo usa las dimensiones espaciales, el encaje se produce en una posición subóptima, debido al problema de localidad explicado en secciones precedentes. No obstante, se debe señalar que aunque el problema de localidad no se elimina, sí se mejora en un grado considerable.



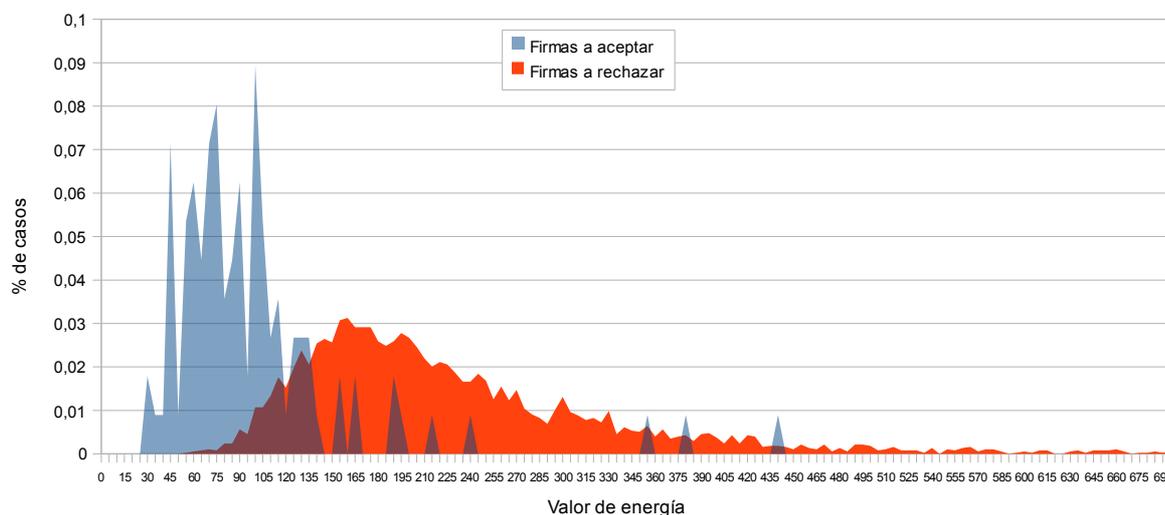
**Figura 5.20.-** Ajuste de un *snake* triangular sobre una imagen con dos figuras geométricas. En (a) y (b) se señalan en amarillo los puntos más cercanos a cada punto de control del *snake* cuando no se utiliza la dimensión del ángulo tangencial y cuando sí se utiliza. En (c) y (e) se itera (a) hasta converger, permitiendo un cambio de las proporciones del 10% y del 1% respectivamente. En (d) y (f) se itera (b) hasta converger, permitiendo un cambio de las proporciones del 10% y del 1% respectivamente.

De nuevo, utilizando la muestra A, se realizan diversas pruebas. En primer lugar se comprueba que el algoritmo logra un mejor ajuste de las firmas que cuando no se utilizaba esta dimensión adicional. La Figura 5.21 presenta un ejemplo que ilustra la mejor aproximación conseguida por los mapas de potencial tridimensionales al ajustar un detalle de una firma.

Por otro lado, la Figura 5.22 presenta los valores de energía obtenidos para la muestra A. Al igual que en la Figura 5.13 se puede comprobar la diferencia en las distribuciones de energía para las dos clases: la de firmas a aceptar y la de firmas a rechazar. Sin embargo, en esta ocasión se aprecia que la diferencia es mayor. Realizando pruebas sobre el umbral de energía a partir del cual se deberían aceptar o rechazar las firmas se encuentra un *EER* del 13.3% para un valor de energía de 131. Así, la introducción de la dimensión angular representa una mejora en el *EER* de 4.5 puntos respecto a la formulación bidimensional de *snakes* (de un 17.8% se pasa a un 13.3%).



**Figura 5.21.-** Detalle de ajuste del *snake* tras el proceso iterativo de convergencia: (a) muestra el resultado utilizando sólo coordenadas bidimensionales, y (b) el resultado añadiendo la dimensión basada en el ángulo de las tangentes.



**Figura 5.22.-** Energía obtenida para las firmas de la muestra A por el *snake* tridimensional.

## 5.4 Formulación borrosa de la energía del *snake*

Se ha comprobado que el ajuste de los parámetros definidos para controlar al *snake* es complejo y costoso. Por un lado, se han requerido múltiples iteraciones de ensayo y error hasta obtener unos valores para dichos parámetros que ofreciesen unos resultados satisfactorios. Por otro lado, resulta complejo explicar el significado de las diferentes variables que gobiernan el modelo.

Con objeto de reducir estos costes y simplificar esa complejidad, se ha realizado una reformulación de las funciones de energía utilizando reglas borrosas (*fuzzy rules*). Otros autores ya han realizado formulaciones borrosas similares de *snakes* con éxito [184][185]. Esta sección describe las modificaciones realizadas para adaptar la definición de energía presentada en la sección precedente.

### Un modelo borroso para la distancia

En la sección 5.3.3 se presentó el ángulo del trazo en cada punto como una dimensión adicional al mapa de potencial de la imagen de la firma. Entonces, se objetó que la medida de la distancia mezclaba dimensiones diferentes (ángulo y distancia espacial), y, por ello, se debía introducir una constante, o peso, que ponderaba la relación entre esas diferentes unidades. El significado de ese peso resultaba oscuro y por ello se elegía mediante prueba y error, buscando que el sistema se comportase de una forma aceptable.

Sin embargo, utilizando un modelo borroso las objeciones a la mezcla de dimensiones parecen diluirse junto con ese parámetro de ponderación. Intuitivamente, dos puntos están cerca si su distancia espacial es pequeña y el ángulo del trazo en el que están contenidos es similar, y están lejos en caso contrario. Si se define de manera precisa qué se considera una distancia espacial pequeña y un ángulo similar, tendremos una formulación equivalente a la presentada en la sección 5.3.3 pero mucho más transparente:

$$\begin{aligned} Si(\text{distancia pequeña}) \wedge (\text{ángulo tangencial parecido}) &\Rightarrow (\text{está cerca}) \\ Si(\text{distancia grande}) \vee (\text{ángulo tangencial diferente}) &\Rightarrow (\text{está lejos}) \end{aligned}$$

Para concretar esta idea intuitiva se sigue un modelo Takagi-Sugeno (TS) [186] de orden cero y se obtiene el siguiente conjunto canónico de 4 reglas:

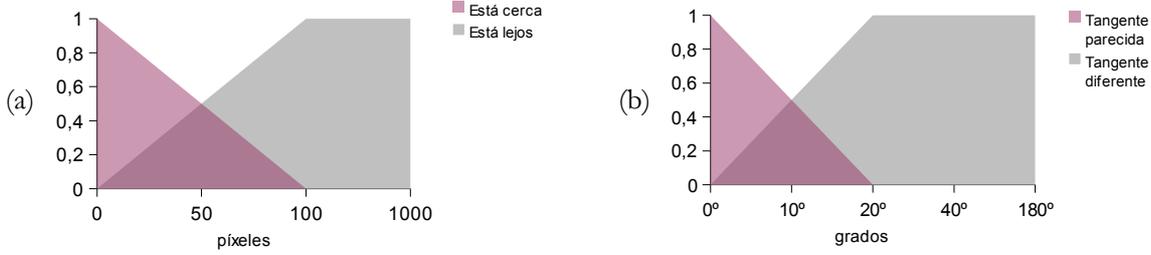
$$\begin{aligned} R_0: Si(e \in \tilde{e}_0) \wedge (\theta \in \tilde{\theta}_0) &\Rightarrow D_{punto} = D_{cerca} \\ R_1: Si(e \in \tilde{e}_1) \wedge (\theta \in \tilde{\theta}_0) &\Rightarrow D_{punto} = D_{lejos} \\ R_2: Si(e \in \tilde{e}_0) \wedge (\theta \in \tilde{\theta}_1) &\Rightarrow D_{punto} = D_{lejos} \\ R_3: Si(e \in \tilde{e}_1) \wedge (\theta \in \tilde{\theta}_1) &\Rightarrow D_{punto} = D_{lejos} \end{aligned} \tag{5.18}$$

donde  $e$  es la menor distancia espacial del punto de control del *snake* evaluado a los puntos activos de la imagen,  $\tilde{e}_0$  es el conjunto borroso “distancia espacial pequeña”,  $\tilde{e}_1$  es el conjunto borroso “distancia espacial grande”,  $\theta$  es el valor absoluto de la diferencia entre ángulo de la tangente al trazo en el punto  $p$  y el ángulo del *snake* en el punto de control evaluado,  $\tilde{\theta}_0$  el conjunto borroso “ángulos de las tangentes similares”,  $\tilde{\theta}_1$  es el conjunto borroso “ángulos de las tangentes distintos”,  $D_{punto}$  es un valor de distancia asociado al punto,  $D_{cerca}$  es el valor asignado a una distancia pequeña, y  $D_{lejos}$  es el valor asignado a una distancia grande. Así, el valor  $D_{punto}$  queda:

$$D_{punto} = \frac{D_{cerca} \cdot \tilde{e}_0(e) \cdot \tilde{\theta}_0(\theta) + D_{lejos} \cdot \tilde{e}_1(e) \cdot \tilde{\theta}_0(\theta) + D_{lejos} \cdot \tilde{e}_0(e) \cdot \tilde{\theta}_1(\theta) + D_{lejos} \cdot \tilde{e}_1(e) \cdot \tilde{\theta}_1(\theta)}{\tilde{e}_0(e) \cdot \tilde{\theta}_0(\theta) + \tilde{e}_1(e) \cdot \tilde{\theta}_0(\theta) + \tilde{e}_0(e) \cdot \tilde{\theta}_1(\theta) + \tilde{e}_1(e) \cdot \tilde{\theta}_1(\theta)} \tag{5.19}$$

donde  $\tilde{e}_0(e)$  es la función de pertenencia de  $\tilde{e}_i$ , y  $\tilde{\theta}_0(\theta)$  es la función de pertenencia de  $\tilde{\theta}_i$  cuando  $i$  toma valores 0 y 1.

Por comodidad, se toma  $D_{cerca}=0$  y  $D_{lejos}=100$ . También, se definen los conjuntos borrosos: “distancia espacial pequeña” y “distancia espacial grande” mediante las particiones unidad de la Figura 5.23(a). Y los conjuntos borrosos “ángulos de las tangentes parecidas” y “ángulos de las tangentes diferentes” mediante las particiones unidad de la Figura 5.23(b).



**Figura 5.23.-** Particiones borrosas relativas a la distancia espacial entre un punto del *snake* y un punto de la imagen (a) y a la similitud de los ángulos de la tangente a un punto del *snake* con la tangente respecto al trazo en un punto de la imagen (b).

Debido a que  $D_{cerca}=0$  y a que existe partición unidad en todos los conjuntos borrosos las ecuaciones 5.18 pueden simplificarse a:

$$D_{punto} = D_{lejos} \cdot (1 - \tilde{e}_0(e) \cdot \tilde{\theta}_0(\theta)) \quad (5.20)$$

### Reglas borrosas para el cálculo de la energía

En realidad, la idea subyacente a los *snakes* con memoria (*shape memory snakes*) presentada en secciones precedentes, es sencilla de adaptar a una formulación con reglas borrosas. Estas reglas deberán reflejar que si el *snake* cambia mucho respecto a su forma inicial o si el *snake* se encuentra lejos de la figura a verificar su energía será alta y, en otro caso, su energía será baja.

*Si* (ángulo cambia poco)  $\wedge$  (proporciones cambian poco)  $\wedge$  (está cerca)  $\Rightarrow$  (energía es pequeña)

*Si* (ángulo cambia mucho)  $\vee$  (proporciones cambian mucho)  $\vee$  (está lejos)  $\Rightarrow$  (energía es grande)

De nuevo, para concretar esta idea se sigue un modelo Takagi-Sugeno de orden cero y se obtiene el siguiente conjunto de ocho reglas resultado de las combinaciones de los tres conjuntos borrosos:

$$\begin{aligned} R_0 &= Si(d \in \tilde{d}_0) \wedge (\alpha \in \tilde{\alpha}_0) \wedge (p \in \tilde{p}_0) \Rightarrow E_{snake} = E_{baja} \\ R_1 &= Si(d \in \tilde{d}_0) \wedge (\alpha \in \tilde{\alpha}_0) \wedge (p \in \tilde{p}_1) \Rightarrow E_{snake} = E_{alta} \\ R_2 &= Si(d \in \tilde{d}_1) \wedge (\alpha \in \tilde{\alpha}_1) \wedge (p \in \tilde{p}_0) \Rightarrow E_{snake} = E_{alta} \\ R_3 &= Si(d \in \tilde{d}_1) \wedge (\alpha \in \tilde{\alpha}_1) \wedge (p \in \tilde{p}_1) \Rightarrow E_{snake} = E_{alta} \\ R_4 &= Si(d \in \tilde{d}_1) \wedge (\alpha \in \tilde{\alpha}_0) \wedge (p \in \tilde{p}_0) \Rightarrow E_{snake} = E_{alta} \\ R_5 &= Si(d \in \tilde{d}_1) \wedge (\alpha \in \tilde{\alpha}_0) \wedge (p \in \tilde{p}_1) \Rightarrow E_{snake} = E_{alta} \\ R_6 &= Si(d \in \tilde{d}_1) \wedge (\alpha \in \tilde{\alpha}_1) \wedge (p \in \tilde{p}_0) \Rightarrow E_{snake} = E_{alta} \\ R_7 &= Si(d \in \tilde{d}_1) \wedge (\alpha \in \tilde{\alpha}_1) \wedge (p \in \tilde{p}_1) \Rightarrow E_{snake} = E_{alta} \end{aligned} \quad (5.21)$$

donde  $d$  corresponde al valor de distancia asociada al punto de control del *snake* evaluado,  $\tilde{d}_0$  es el conjunto borroso “distancia pequeña”,  $\tilde{d}_1$  es el conjunto borroso “distancia grande”,  $\alpha$  corresponde al valor del cambio en el ángulo entre los segmentos adyacentes al punto de control evaluado,  $\tilde{\alpha}_0$  es el conjunto borroso “ángulo pequeño”,  $\tilde{\alpha}_1$  es el conjunto borroso “ángulo grande”,  $p$  corresponde al valor del cambio en las proporciones entre los segmentos adyacentes al punto de control evaluado,  $\tilde{p}_0$  es el conjunto borroso “cambio de proporciones pequeño”,  $\tilde{p}_1$  es el conjunto borroso “cambio de proporciones grande”,  $E_{snake}$  corresponde a la energía asociada al *snake*,  $E_{baja}$  es un valor asociado a una energía pequeña y  $E_{alta}$  es un valor asociado a una energía alta.

Con objeto de reducir la formulación 5.21 podemos generalizar aquella formulación a:

$$R_i = Si(n_0 \in \tilde{n}_{i,0}) \wedge (n \in \tilde{n}_{i,1}) \wedge (n \in \tilde{n}_{i,2}) \Rightarrow E_{snake} = E_i \quad (5.22)$$

donde  $n_0$  es  $d$ ,  $n_1$  es  $\alpha$ ,  $n_2$  es  $p$ , e  $i$  toma valores entre 0 y 7.

Podemos escribir entonces que la energía del *snake* es:

$$E_{snake} = \frac{\sum_{i=0}^7 E_i \prod_{j=0}^2 \tilde{n}_{i,j}(x_j)}{\sum_{i=0}^7 \prod_{j=0}^2 \tilde{n}_{i,j}(x_j)} \quad (5.23)$$

Por comodidad, se toma  $E_{baja}=0$  y  $E_{alta}=100$ . También, se definen los conjuntos borrosos “está cerca” y “está lejos” mediante las particiones unidad de la Figura 5.24(a). Los conjuntos borrosos “ángulo cambia poco” y “ángulo cambia mucho” mediante las particiones unidad de la Figura 5.24(b). Y los conjuntos borrosos “proporciones cambian poco” y “proporciones cambian mucho” mediante las particiones unidad de la Figura 5.24(c).

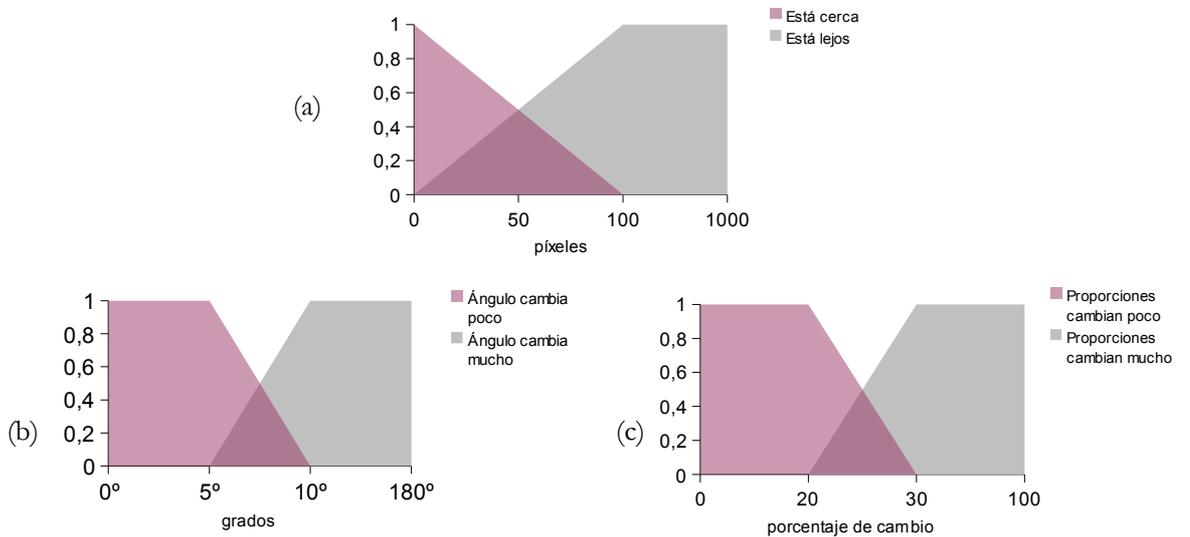


Figura 5.24.- conjuntos borrosos para el cálculo de la energía borrosa de un *snake*.

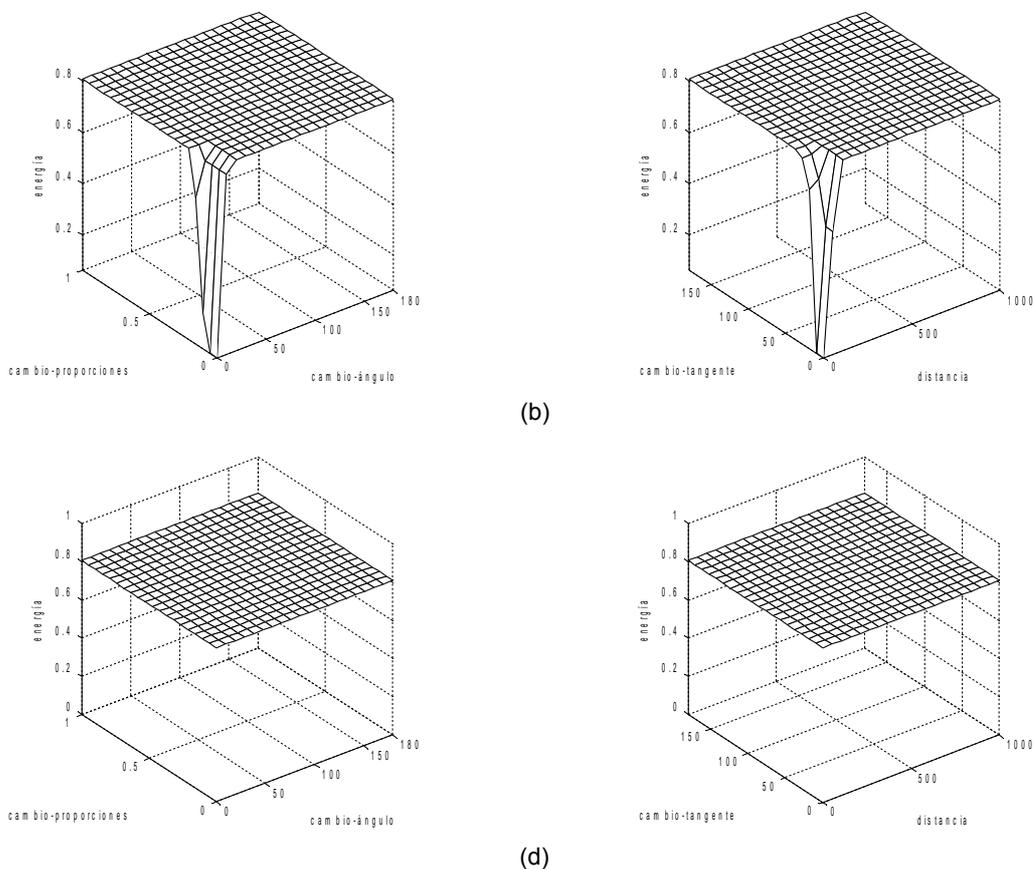
Al ser  $E_{baja}=0$  y al tener partición unidad, la ecuación 5.23 se simplifica a:

$$E_{snake} = E_{alta} \cdot (1 - \tilde{d}_0(d) \cdot \tilde{\alpha}_0(\alpha) \cdot \tilde{p}_0(p)) \quad (5.24)$$

### Experimentos con los *snakes* tridimensionales *fuzzy*

La Figura 5.25 presenta los valores obtenidos de energía por las reglas definidas para diferentes combinaciones de los parámetros de entrada. Obsérvese como la energía siempre es alta, excepto cuando los cambios respecto a la inicialización son pequeños y además están relativamente cerca de un punto con una tangente similar.

Por otro lado, de nuevo, se utiliza la muestra A para realizar pruebas preliminares para comprobar los resultados aportados por la formulación *fuzzy*. En estas pruebas se han obtenido los valores de energía recogidos en la Figura 5.26. De nuevo la clase de las firmas a aceptar y la de las firmas a rechazar aparecen claramente diferenciadas. Las pruebas sobre el umbral de energía a partir del cual se deberían aceptar o rechazar las firmas muestran un *EER* del 12.5% para un valor de energía de 20.99. Este resultado es 0.8 puntos mejor al del enfoque precedente que utilizaba una formulación *crisp*. Esta mejora es atribuible a la mejora en la comprensión de los parámetros de ajuste introducida por la formulación borrosa.



**Figura 5.25.-** Resultados de energía borrosa obtenidos al aplicar las reglas borrosas propuestas. En (a) y (b) se mantienen al mínimo la distancia espacial, mientras que en (c) y (d) se mantiene al máximo.

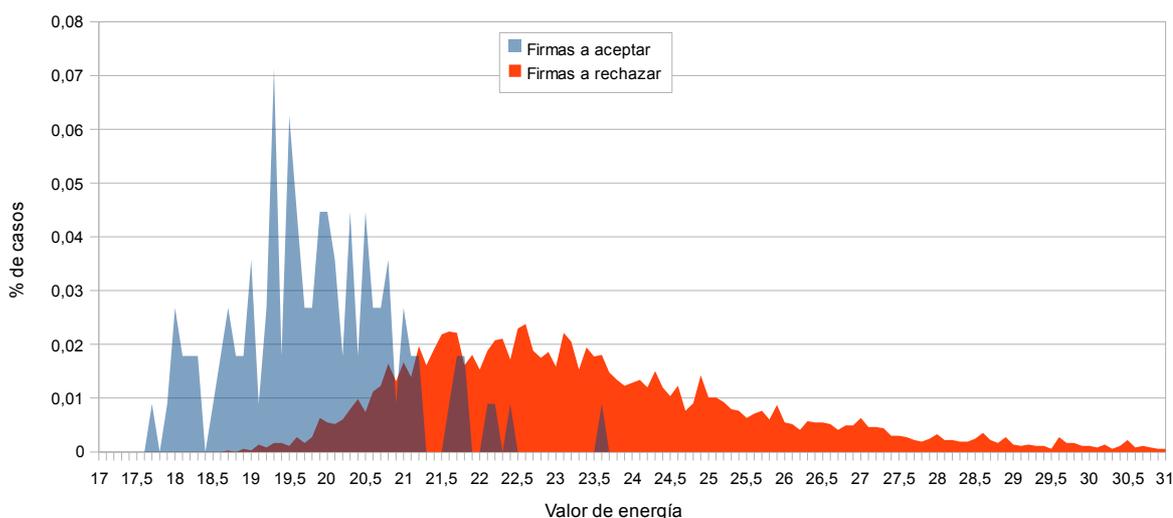


Figura 5.26.- Energía obtenida para las firmas de la muestra A por el *snake fuzzy*.

## 5.5 Características discriminantes para la verificación de firmas mediante *snakes*

En este apartado se proponen un conjunto de características discriminantes que permiten utilizar el resultado del proceso de convergencia de un *snake* sobre la imagen de una firma para la verificación de la misma.

Se ha comprobado que tras un número variable de iteraciones, el *snake* siempre converge a una solución. De acuerdo con el esquema de la Figura 5.1 el siguiente paso consiste en estudiar el resultado del ajuste del *snake* para obtener un valor de confianza en la firma que se está verificando. Este aspecto se puede analizar desde dos perspectivas no excluyentes:

- a) Determinar si la silueta en la que ha convergido el *snake* puede considerarse suficientemente similar al modelo inicial como para que se decida aceptar la firma que se presenta como auténtica.
- b) Determinar si la silueta se ha situado correctamente sobre los puntos de la firma de la imagen que se presenta.

Tras responder a estas preguntas se obtendrán una o varias medidas que se podrán utilizar a modo de características discriminantes para decidir la autenticidad de la firma utilizando un clasificador de patrones. En esta sección se describen las características discriminantes consideradas. Nuevamente, para las diferentes pruebas que se realizan sólo se utiliza la muestra A.

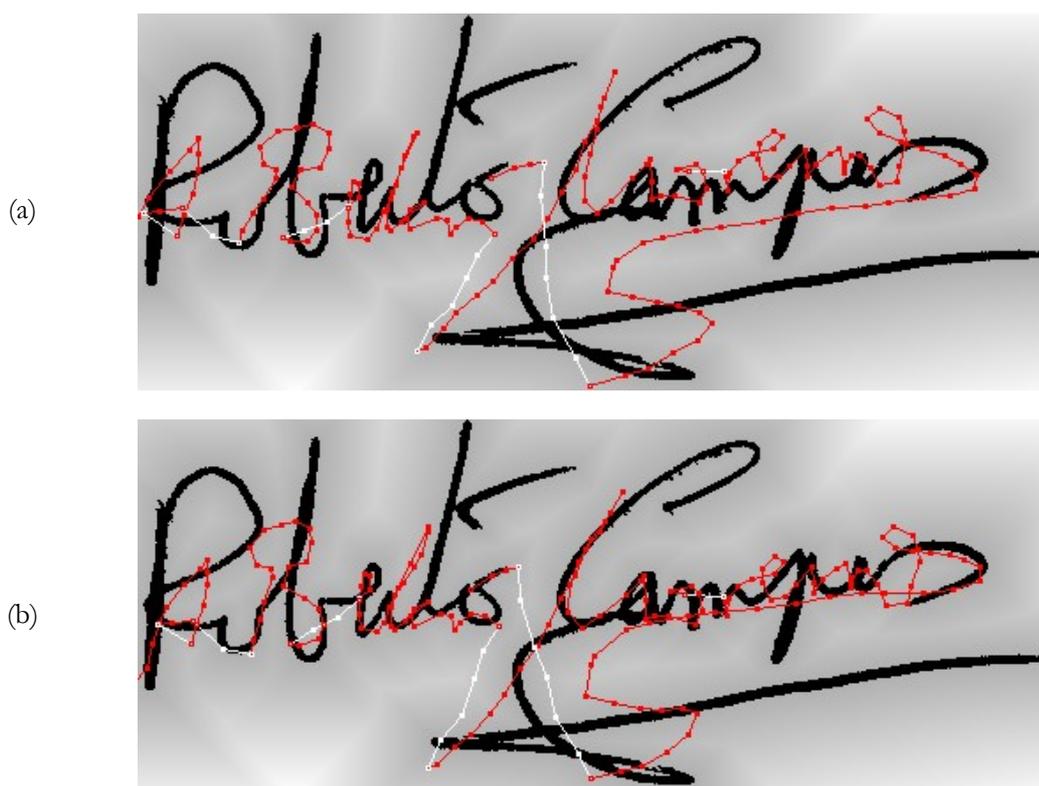
### 5.5.1 Características habituales de ajuste elástico

En la mayoría de los trabajos de reconocimiento de imágenes mediante modelos deformables se proponen soluciones desde la perspectiva (a) del punto anterior. Estas soluciones consisten en medir la similitud entre dos siluetas deformables. Dependiendo del autor se proponen diferentes métodos para realizar esta comparación. Yoshida y Sakoe [187] comparan las diferencias entre los ángulos y las proporciones de segmentos correspondientes entre dos siluetas de textos manuscritos capturados *on-line*. Basri y otros [188] proponen una medida del coste necesario para transformar una silueta en

la otra. Al utilizar estos enfoques para la verificación de firmas con los *snakes* se obtiene como resultado que las siluetas casi siempre coinciden. De hecho, los algoritmos que se proponen funcionan correctamente, ya que en realidad las formas a comparar son las mismas tras las pequeñas modificaciones resultantes del proceso de convergencia. Obsérvese que, en el caso de la verificación de firmas se mide la similitud entre el *snake* antes de comenzar el proceso de convergencia (es decir el modelo almacenado) y el *snake* tras el proceso de convergencia.

Por ejemplo, en la Figura 5.27 se puede apreciar que la silueta del *snake* no cambia significativamente tras el proceso de convergencia, a pesar de que la firma contenida en la imagen a la que trata de adaptarse no se corresponde con la del *snake*. Esto se debe a la propiedad de memoria con la que se ha dotado al *snake*. Cualquier método que compare siluetas, si funciona, debería decidir que la silueta del *snake* en 5.27(b) es igual a la de 5.27(a), por lo que esta medida por sí sola siempre resultará insuficiente como característica discriminante.

Se puede concluir que, para tener éxito al medir la calidad del ajuste del *snake* para la verificación de firmas, el método de comparación que se utilice debe tener en cuenta, no sólo las siluetas del *snake* antes y después del proceso de convergencia, sino también, la imagen sobre la que se ha realizado este proceso. Así, las características no sólo deberán medir la deformación del *snake*, sino que también deberán medir cómo se ha ajustado el *snake* a la imagen de fondo, y cómo la imagen de fondo queda descrita por la posición del *snake* tras su convergencia.



**Figura 5.27.-** Ejemplo del proceso de convergencia del *snake* en un caso de no coincidencia. Se intenta ajustar el modelo del sujeto 1 a la imagen de una firma del sujeto 7. La imagen (a) muestra la inicialización del proceso y (b) muestra el resultado tras el proceso iterativo de convergencia.

### 5.5.2 Energía del *snake* como característica discriminante

La primera característica discriminante que se ha considerado es la propia energía del *snake* obtenida según las ecuaciones (5.1), (5.3) y (5.4). Idealmente, un valor alto de energía significa que los puntos del *snake* se encuentran en posiciones alejadas de las consideradas satisfactorias. Sin embargo, debido a que la energía puede variar de forma considerable dependiendo del tamaño del *snake*, no resulta posible utilizarla sin una normalización previa que considere la ponderación respecto al número de puntos de control.

Otra posibilidad consiste en el uso de la definición borrosa de energía proporcionado descrita en la sección precedente. Esta posibilidad aporta la ventaja de su acotación, pues la suma de las energías borrosas de los puntos de control dividida entre el número de puntos de control siempre produce un valor en el intervalo  $[0,100]$ , mientras que las deformaciones aplicadas sobre las fórmulas de tipo resorte de la formulación no borrosa (ecuaciones 5.8 y 5.12) producen valores no acotados.

Esta característica mide, por un lado, el grado de deformación del *snake*, pues tiene en cuenta las fuerzas internas, pero también considera la posición lograda por el *snake* sobre la firma, debida a las fuerzas externas.

### 5.5.3 Índice de coincidencia como característica discriminante

En las pruebas que se han realizado utilizando la energía de ajuste del *snake* como característica discriminante, se observó que en vez de computar sólo la energía correspondiente a los puntos de control del *snake*, sería conveniente computar la energía externa de todos los píxeles intersectados por los segmentos que unen puntos de control consecutivos. Este cómputo adicional tendría la ventaja de ponderar más exactamente la correspondencia entre los puntos de la silueta del *snake* y los píxeles activos de la imagen.

Este cambio aumenta significativamente el número de puntos en los que hay que calcular la función de energía externa. Debido a este incremento de complejidad, resulta prohibitiva su utilización durante las iteraciones de la etapa de convergencia. Sin embargo, sí es posible y apropiada su aplicación en esta etapa final de análisis de coincidencia.

Para obtener un valor normalizado entre 0 y 1 de coincidencia entre el *snake* y la imagen de la firma a la que se ajusta, se utiliza la siguiente expresión:

$$coincidencia(S) = \sum_{\forall p \in S} c(p) \quad (5.25)$$

donde:

$$c(p) = \frac{m_{imagen}(p)}{k_{fc} I G} \quad (5.26)$$

siendo  $S$  el conjunto de todos los píxeles de la imagen a verificar coincidentes con la silueta del *snake*,  $k_{fc}$  un factor de escala y  $G$  el grosor medio del trazo de la firma en píxeles (ver ecuación 4.1). De nuevo debe tenerse en cuenta que los segmentos de unión entre partes inconexas de la firma no deben ser considerados

### 5.5.4 Índice de distancia como característica discriminante

Sobre la imagen de una firma que posea muchos píxeles activos (el caso límite sería una imagen con todos los píxeles activos) las características descritas hasta ahora devolverán valores de alta coincidencia. Esto se debe a que todos los puntos del *snake* se encontrarán sobre píxeles activos.

Este hecho motivó que se buscara una característica que permitiese la discriminación en este caso. La posibilidad que se ha considerado consiste en medir el porcentaje de píxeles activos totales de la imagen que no se encuentran cerca del *snake*. Este valor, que se ha llamado índice de distancia, se calcula utilizando el mapa de potencial bidimensional  $m_{snake}(x,y)$ , que tiene su origen en el *snake* una vez iterado, y que permite calcular a qué distancia se encuentran los píxeles de la firma respecto a la posición del *snake*. Para obtener un valor normalizado entre 0 y 1 del índice de distancia entre el *snake* y la imagen de la firma a la que trata de ajustarse se utiliza la siguiente expresión:

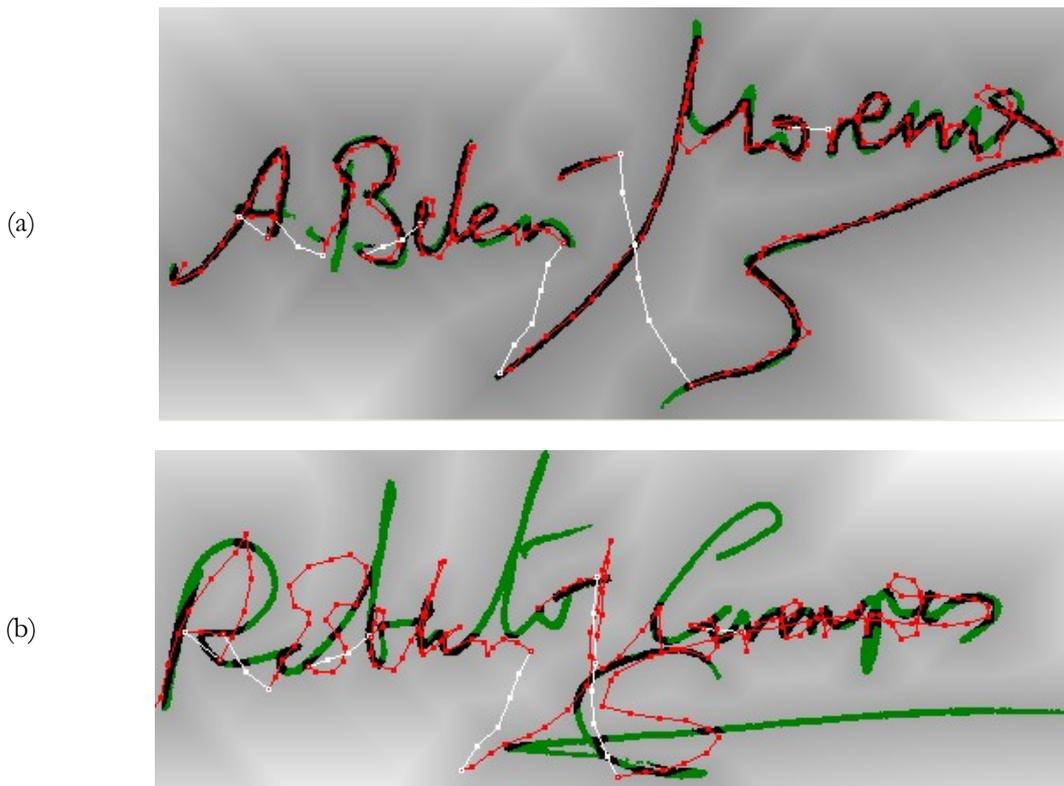
$$distancia(S) = \frac{1}{N} \sum_{\forall p \in \mathcal{A}} d(p) \quad (5.27)$$

donde:

$$d(p) = \begin{cases} 1 & m_{imagen}(p) < G \\ 0 & m_{imagen}(p) \geq G \end{cases} \quad (5.28)$$

siendo  $\mathcal{A}$  el conjunto de todos los píxeles activos de la imagen a verificar,  $N$  la cardinalidad de  $\mathcal{A}$  y  $G$  el grosor medio del trazo de la firma en píxeles (ver ecuación 4.1).

La Figura 5.28 muestra un ejemplo en el que se han coloreado en verde los píxeles activos de la imagen que, tras el proceso de convergencia, se encuentran alejados del *snake*. En definitiva, aquellos puntos  $p$  tales que la ecuación (5.28) devuelve 0.



**Figura 5.28.-** Ejemplos de ajuste del *snake* de una persona a las imágenes de dos firmas. Se han etiquetado en verde los puntos cuyo índice de distancia es 0. La imagen (a) corresponde a la misma persona que originó el *snake*, mientras que la imagen (b) corresponde a otra persona.

El ruido aleatorio, como puede eliminarse fácilmente mediante filtrado morfológico, no afecta al valor de esta característica. Sin embargo, el ruido con estructura, que puede existir en la imagen de la firma tras el proceso de segmentación, es un inconveniente para este índice, pues incrementa su valor al recoger los píxeles que se encuentran alejados de la firma y que en realidad corresponden a ruido.

### 5.5.5 Análisis de las características discriminantes

El estudio de las características descritas sobre la muestra A revela los resultados de la tabla 5.2.

Característica	Fisher	Media		Desviación típica	
		Genuinos	Falsificaciones	Genuinos	Falsificaciones
Energía	1.72	0.20	0.24	0.01	0.03
Coincidencia	1.46	0.71	0.29	0.19	0.28
Distancia	2.53	0.68	0.41	0.11	0.12

**Tabla 5.2.-** Análisis de las características discriminantes calculadas tras la convergencia del *snake*.

Además, se obtienen las siguientes matrices de correlación para las características de energía, coincidencia y distancia.

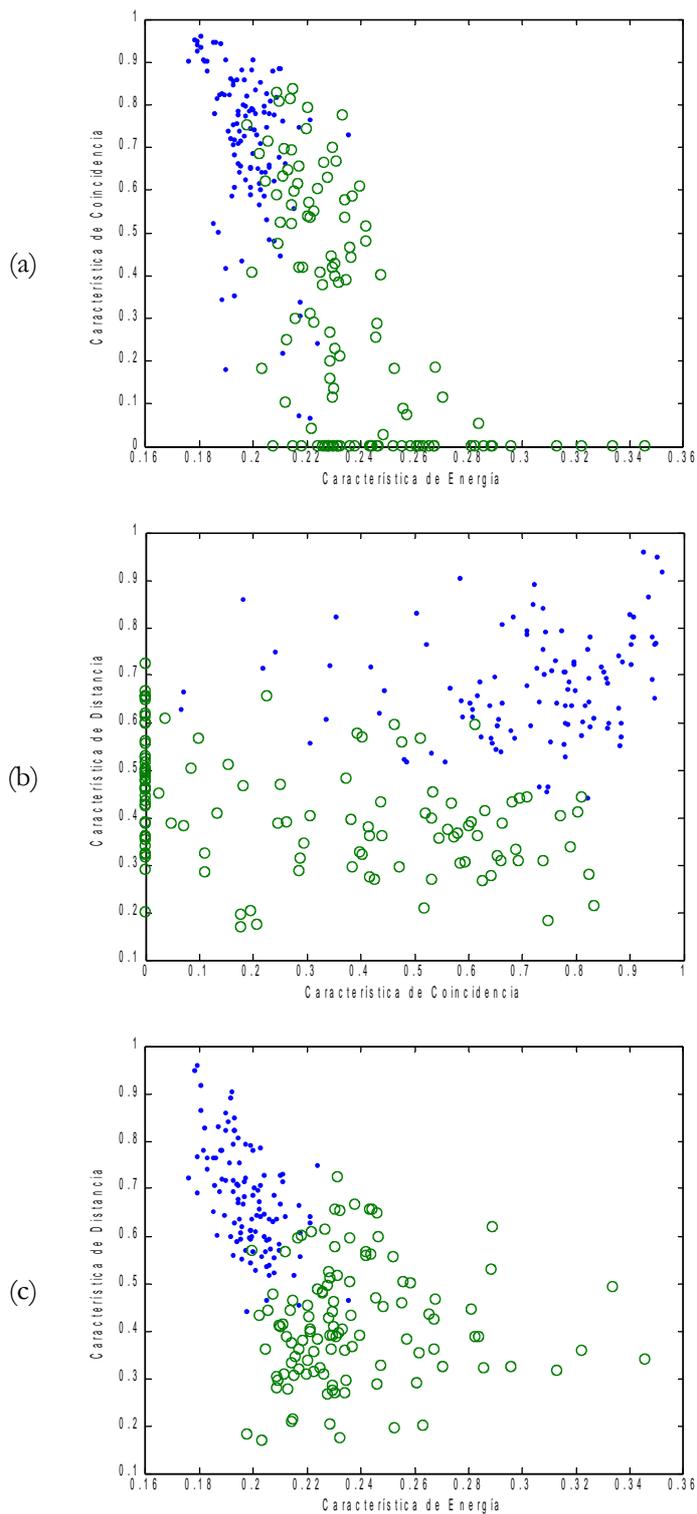
$$c_{genuinos} = \begin{pmatrix} 1 & -0.46 & -0.57 \\ -0.46 & 1 & 0.14 \\ -0.57 & 0.14 & 1 \end{pmatrix} \quad c_{falsificaciones} = \begin{pmatrix} 1 & -0.56 & 0.06 \\ -0.56 & 1 & -0.38 \\ 0.06 & -0.38 & 1 \end{pmatrix} \quad (5.29)$$

El mayor ratio de Fisher del índice de distancia indica que es la característica más discriminante, aunque el análisis de las medias y las desviaciones típicas indica que las diferencias no son muy significativas. Por otro lado, las matrices de correlación muestran que las características gozan de un deseable cierto grado de independencia.

### 5.5.6 Entrenamiento de un clasificador

La Figura 5.29 presenta varias vistas del valor de las características discriminantes, de coincidencia, distancia y energía, para una selección de la muestra A. Esta selección se ha realizado de manera que el número de patrones de ambas clases, patrones a rechazar y patrones a aceptar, sea el mismo.

En la Figura 5.29(b) se aprecia que los patrones a aceptar toman valores altos para el índice de coincidencia y para el de distancia, y los patrones a rechazar toman valores bajos. Respecto al índice de energía se aprecia lo opuesto: valores más bajos para los patrones a aceptar. A priori, parece sencillo construir un clasificador que permita discriminar entre ambos tipos de clase. Es por ello, que se ha diseñado una sencilla Red Neuronal del tipo Perceptrón Multicapa (*MLP*) para realizar dicha clasificación. La entrada de la red viene determinada por las tres características discriminantes existentes. Como lo que se pretende es aceptar o rechazar cada firma, para la salida basta con una neurona que se active o desactive si se debe aceptar o rechazar el patrón. La geometría de las diferentes vistas de la Figura 5.29 (especialmente la vista (b)) parece indicar que una sola capa oculta puede ser suficiente para clasificar correctamente los patrones. La Figura 5.30 presenta un esquema de la red propuesta.



**Figura 5.29.-** Posición de los patrones de la muestra A respecto a sus características discriminantes. Los círculos verdes corresponden a los patrones a rechazar y los puntos azules son los patrones a aceptar. En (a) se comparan las características de energía y coincidencia, en (b) las de energía y distancia, en (c) las de coincidencia y distancia.

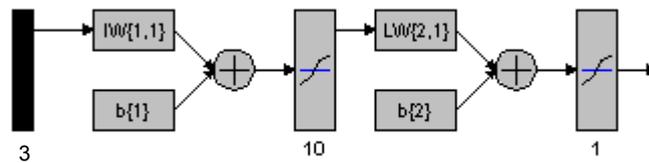


Figura 5.30.- Esquema del MLP utilizado como clasificador.

La Figura 5.31 presenta el error cuadrático medio cometido durante el proceso de entrenamiento de la red utilizando la muestra A. La Figura 5.32 presenta la salida de la red para los diferentes valores de los índices de coincidencia y distancia manteniendo el valor de energía en 0.22. Se puede apreciar que la forma de la superficie se aproxima a la que marcan los patrones de la Figura 5.29(b).

## 5.6 Resultados del sistema de verificación basado en snakes

Una vez entrenado el clasificador se han realizado diversos experimentos sobre las muestras B y E. La Figura 5.33 y 5.34 presentan los resultados obtenidos.

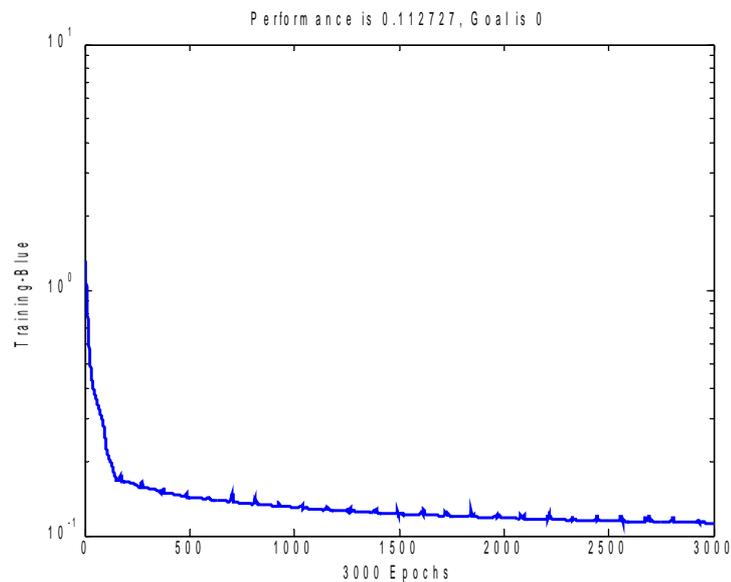
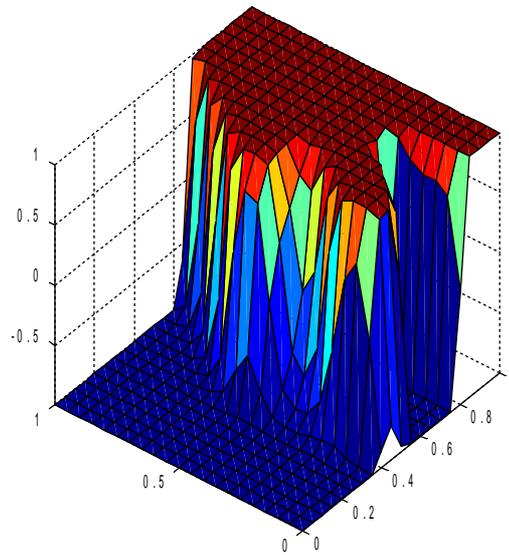
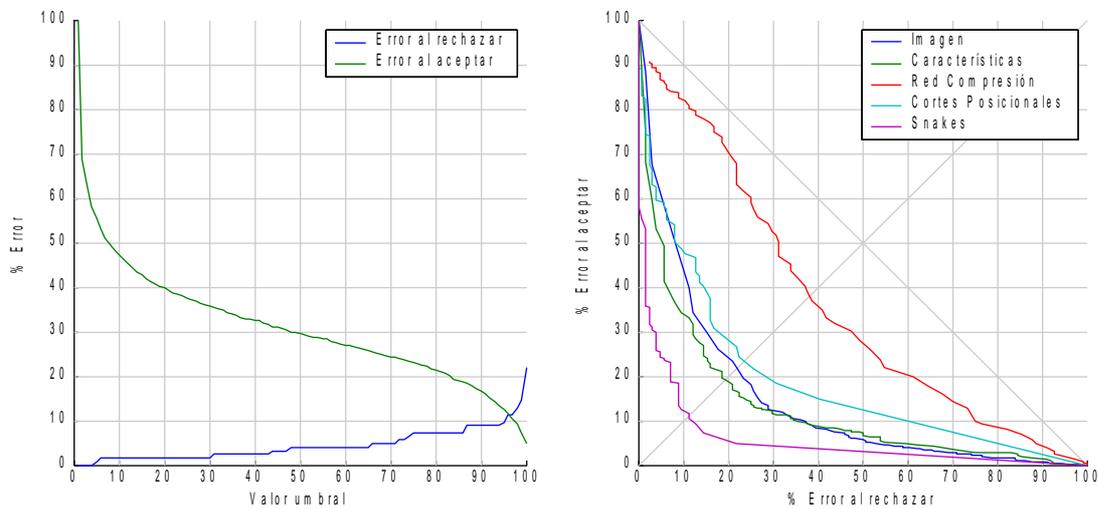


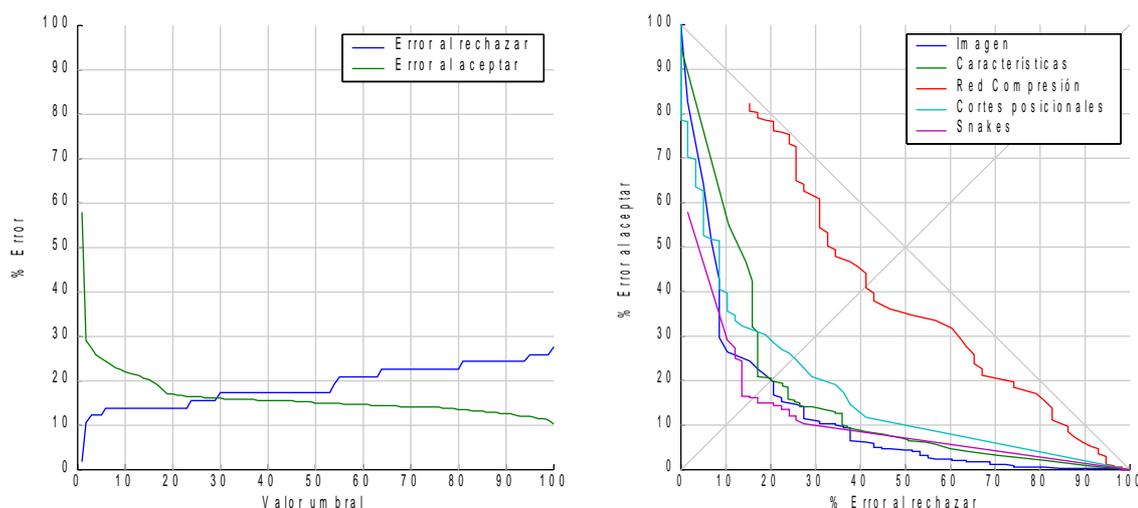
Figura 5.31.- Curva del error cuadrático medio obtenido durante el entrenamiento del MLP de la figura 5.30.



**Figura 5.32.-** Salida del *MLP* de la figura 5.30 una vez entrenada cuando varía el índice de coincidencia y el de distancia y se mantiene la característica de energía con valor 0.22.



**Figura 5.33.-** Curvas de error obtenidas al usar *snakes* tridimensionales con memoria sobre la muestra B: (a) presenta el error al aceptar (*FAR*) y al rechazar (*FRR*), y (b) es una curva *ROC* que superpone el resultado obtenido con: el del método basado en la presentación directa de la imagen a un *MLP*, el basado en características discriminantes globales, el basado en Redes de Compresión y el de Cortes Posicionales.



**Figura 5.34.-** Curvas de error obtenidas al usar *snakes* tridimensionales con memoria sobre la muestra E: (a) presenta el error al aceptar (*FAR*) y al rechazar (*FRR*), y (b) es una curva ROC que superpone el resultado obtenido con: el del método basado en la presentación directa de la imagen a un *MLP*, el basado en características discriminantes globales, el basado en Redes de Compresión y el de Cortes Posicionales.

En la figura 5.34 se puede comprobar que el *EER* sobre la muestra B es aproximadamente del 12%, lo cual mejora de manera considerable los resultados de todos los métodos propuestos en el capítulo precedente. Además, los resultados, en general, también son mejores, sobre todo en cuanto a la tasa de error al rechazar (*FRR*). Este resultado es similar al obtenido por X. Xiaoa y G. Leedhamb [78], aunque este autor usa 6 firmas para el aprendizaje de cada individuo y aquí solo se usa una para este fin.

La Figura 5.34 presenta los resultados sobre la muestra E. Tal y como ocurrió en los todos los casos anteriores el error fue mayor que el obtenido para la muestra B. De todas formas el valor *EER* entorno al 16%, también mejora los resultados de todos los métodos precedentes para esta muestra.

## 5.7 Resumen del capítulo

En este capítulo se ha presentado un método de verificación novedoso basado diferentes modelos novedosos de *snakes*. En general, las propuestas realizadas se basan en una serie de modificaciones sobre el algoritmo original de Kass, Witkin y Terzopoulos [26] realizadas para posibilitar su aplicación al problema de verificación *off-line* de firmas sobre imágenes bitonales. Los resultados obtenidos por el método de verificación basado en *snakes* mejoran los obtenidos por todos los métodos previamente presentados en esta Tesis.

Además, con objeto de disponer de un método de verificación completamente automático, también se ha presentado un método de generación no asistido de los *snakes* a partir de una firma modelo, el cual se ha utilizado en los experimentos.



## Capítulo 6 Conclusiones

En esta Tesis se ha abordado el problema de la verificación *off-line* de firmas manuscritas desde una perspectiva realista: utilizando imágenes bitonales de baja resolución y documentos reales, considerando la velocidad y la escalabilidad del sistema, usando una sola firma modelo por cada firmante y atendiendo al caso de las falsificaciones aleatorias. Primeramente, se ha introducido toda la problemática relativa a esta área de investigación. Luego, se han analizado las diferentes propuestas para la verificación *off-line* de firmas que se pueden encontrar en la bibliografía. Presentado, desde los principales enfoques aplicados clásicamente por la práctica forense, hasta los trabajos actuales basados en técnicas de Visión Artificial.

Como resultados de esta Tesis se ha presentado: una base de datos de firmas, un método novedoso de segmentación de firmas y diferentes propuestas originales que permiten realizar la tarea de verificación *off-line* de firmas de manera automática en un entorno realista. En los siguientes puntos se presentan las conclusiones de cada uno de los trabajos y métodos expuestos.

### 6.1 Conclusiones respecto al estado del arte

Hay áreas de la Percepción Computacional donde existen paradigmas de reconocimientos que están asentados y que son aceptados como efectivos por la comunidad científica. Tal es el caso del reconocimiento del habla (que utiliza los Modelos Ocultos Discretos de Markov [189]), o del reconocimiento de caracteres manuscritos aislados (donde se usan diversos Modelos de Redes Neuronales [12]). Sin embargo, no ocurre esto en el problema de la verificación *off-line* de firmas, en donde no existe una técnica de reconocimiento madura o ampliamente aceptada.

En particular, en la literatura se encuentran pocos trabajos en los que se trate el problema de la localización de la firma en un entorno ruidoso [12]. Se ha visto que el uso de imágenes en color, o en niveles de gris, facilita mucho la segmentación de la firma, mientras que el uso de imágenes bitonales complica la tarea. Sin embargo, se ha comprobado que los requisitos de producción de los entornos reales fomenta el uso de imágenes bitonales. Además, la evidencia de que cualquier persona es capaz de realizar la tarea de segmentación en estas condiciones [23] anima en la búsqueda de una solución automática.

Tampoco suele abordarse en la literatura el problema de la imposibilidad de usar más de una firma por individuo para el aprendizaje del sistema. El problema de la escalabilidad y de la automatización del proceso también es olvidado en muchos trabajos. Aún se encuentran menos trabajos que traten todos estos aspectos simultáneamente con el objeto de proporcionar un sistema operativo en condiciones realistas.

Se ha comprobado que no siempre se sigue la recomendación de Leclerc y Plamondon [41] sobre presentar los resultados de verificación de firmas utilizando los dos tipos de errores (*FAR* y *FFR*) que puede cometer un sistema al verificar una firma. También se ha detectado que, en general, las bases de datos que utilizan los trabajos analizados son de naturaleza privada, lo que dificulta comparar sus resultados. Además, como ya señalaba Plamondon y Srihari en el año 2000 [12], las

bases de datos que se suelen utilizar siguen siendo demasiado pequeñas, en número de firmantes y en número de firmas por firmante, como para obtener resultados estadísticamente significativos.

Finalmente, se puede constatar que, últimamente, la atención se centra en el problema de los falsificadores habilidosos. Sin embargo, en esta área los éxitos presentados son cuestionables por dos motivos. Primero, por la forma de abordar el problema por parte de algunos autores, que entrenan sus sistemas con falsificaciones de los mismos autores que luego intentan desenmascarar. Por otro lado, ya que las fotocopias y el calcado son prácticas que no permiten su discriminación más que con un análisis microscópico, se hace difícil justificar la utilidad práctica del análisis de este tipo de falsificaciones, sobre todo teniendo en cuenta que según varios estudios [19][11] el 99% del fraude corresponde a falsificaciones aleatorias.

## 6.2 Conclusiones respecto a la base de datos de firmas

Se ha creado una base de datos de imágenes de firmas manuscritas y se ha divulgado como de dominio público. Esta base de datos permitirá a los investigadores realizar experimentos sobre verificación *off-line* de firmas y contrastarlos con los resultados de otros autores que también la utilicen.

Las firmas de la base de datos corresponden a 75 individuos diferentes a los que se les ha solicitado colaboración. Algunos individuos, 17 en concreto, han prestado su firma en diversas capturas separadas por dos o más años de diferencia.

La base de datos no sólo contiene imágenes segmentadas de firmas, sino que también contiene firmas sobre tickets de compra simulados, con objeto de que la misma muestra se pueda utilizar tanto para ensayos de verificación como de segmentación de firmas. Como resultado del proceso de captura de datos realizado, se pueden presentar las siguientes conclusiones experimentales:

- El cambio del tamaño de la zona sobre la que se firma influye de manera no trivial en el tamaño de la firmas.
- Las firmas son en general más anchas que altas, con un ratio medio de 1 a 3.
- Las firmas españolas suelen tener pocas componentes conexas (en nuestra muestra nunca más de 10 elementos).

Respecto al proceso de captura de datos, hay que comentar que se ha apreciado cierta reticencia a firmar en algunos individuos a los que se les ha pedido colaboración, pese a que se aseguró el total anonimato de las muestras recogidas y un uso exclusivamente científico. Esta reticencia parece justificada por el valor social de las firmas.

### 6.2.1 Conclusiones a la segmentación automática de firmas

Se ha presentado un método de segmentación que ofrece resultados satisfactorios sobre documentos reales que contienen ruido de diversa procedencia.

Como se ha podido comprobar, debido al volumen de la muestra analizada y a su naturaleza, no ha resultado posible la obtención automática de unos resultados cuantitativos sobre segmentación de firmas, por lo que en una primera etapa de evaluación se ha recurrido a una verificación visual de los resultados. Así, se ha obtenido un 2.70% de error al etiquetar píxeles de ruido como pertenecientes a la firma y un 1.99% de error al etiquetar píxeles de la firma como ruido, respecto al total de píxeles del documento, teniendo en cuenta que la firma comprende un 8.79% de los píxeles del documento.

Por otro lado, para las pruebas de verificación se han utilizado las firmas segmentadas por el método presentado. Estas pruebas han permitido obtener, de manera totalmente automática, valores cuantitativos que corroboran la eficacia del método de segmentación propuesto.

### 6.3 Conclusiones a los métodos de verificación propuestos

Se ha presentado un método de generación de muestra sintética que pretende simular la variación intrapersonal inherente a las firmas de un mismo individuo.

Se han analizado los resultados obtenidos por dos métodos muy utilizados en la literatura de verificación de firmas, uno basado en el uso de un Perceptrón Multicapa (*MLP*) directamente sobre la imagen de la firma y otro basado en la extracción de características discriminantes. Estos métodos se han alterado, mediante el uso de la muestra sintética durante su entrenamiento, para que sólo precisen una firma para aprender la firma de un individuo.

También se han presentado tres enfoques novedosos (junto con diversas variantes de los mismos) para la verificación de firmas, respetando el requisito de utilizar una sola muestra para aprender la firma de un individuo. Estos métodos son: la propuesta basada en Redes de Compresión, el método de Cortes Posicionales y finalmente los diferentes algoritmos basados en el uso de *snakes*.

Se ha cuidado especialmente que todos los métodos presentados sean desasistidos y escalables. Así, ninguno de los métodos descritos requiere intervención humana y el diseño de los mismos no está limitado en cuanto al número de individuos a verificar, y el coste de añadir un nuevo individuo es constante.

Además, se ha estudiado la robustez de los métodos desde dos perspectivas: frente a una muestra limpia y frente a una muestra real procedente de cheques de una entidad bancaria. Esta segunda prueba se ha basado en el método de segmentación de firmas en documentos que también se ha presentado como un resultado de esta Tesis.

La experimentación, sobre cheques bancarios, ha permitido comprobar que, el ruido inherente a las imágenes reales que contienen firmas empeora los resultados de verificación en un 5% aproximadamente, respecto a los resultados obtenidos sobre muestra sin ruido.

#### 6.3.1 Conclusiones respecto a la verificación usando muestra sintética

Por un lado, se ha mostrado que la mayoría de los métodos de verificación de firmas encontrados en la bibliografía precisan de abundante muestra de aprendizaje. Por otro lado, se ha planteado la restricción a este requisito en los sistemas reales, en los que no es posible solicitar varias firmas a cada individuo para el aprendizaje del sistema. Por ello, se ha optado por desarrollar un algoritmo que genera automáticamente una muestra sintética, útil para el aprendizaje, a partir de una única firma, intentando reproducir la variabilidad intrapersonal natural en las firmas de una misma persona. El algoritmo se basa en un análisis cuantitativo que se ha realizado sobre las variaciones de ciertos rasgos que suelen apreciarse entre las firmas de un mismo individuo.

Inicialmente, esta muestra sintética se ha utilizado para entrenar dos clasificadores por cada individuo. El primero está basado en el entrenamiento de una Red de Neuronas, de tipo *MLP*, que utiliza como entrada la propia imagen de la firma una vez normalizada. El segundo clasificador, también basado en el entrenamiento de una red *MLP* para cada individuo, esta vez usa como entrada un conjunto de características discriminantes descritas en la bibliografía.

Con el clasificador que utiliza la imagen directamente se ha obtenido un *Equal Error Rate (EER)* del 22% sobre la muestra sin ruido y del 20% sobre la muestra procedente de documentos bancarios.

Con el clasificador que extrae previamente características discriminantes se ha obtenido un *EER* del 20% sobre ambas muestras. Estos resultados son similares a los reportados por la bibliografía [173], aunque otros autores, al implementar estos métodos usan varias firmas de cada individuo para el aprendizaje, y en los métodos presentados sólo se utiliza una firma junto con el proceso de generación sintética de firmas descrito. Ello permite a esta Tesis extraer la conclusión de que el método de generación de firmas sintéticas cumple el objetivo perseguido de reproducir la variabilidad intrapersonal inherente a las firmas de un mismo individuo.

También se ha utilizado la muestra sintética para entrenar un sistema novedoso de verificación basado en Redes de Compresión. El método consiste en entrenar una Red de Compresión para cada individuo, y utilizar la diferencia entre la entrada y la salida ofrecida por la red como una medida del error al verificar una firma. Los resultados de este método no fueron satisfactorios en comparación con las otras propuestas, obteniendo un *EER* del 38% y del 41% para la muestra sin ruido y la muestra de documentos bancarios respectivamente.

Ante la contradicción entre la potencia observada en algunos experimentos con Redes de Compresión sobre firmas y los pobres resultados ofrecidos por el método presentado, se ha propuesto otro enfoque, también novedoso, que combina las Redes de Compresión con una serie de cortes verticales para obtener una mejor medida de la confianza en una firma al ser verificada. Esta propuesta, que se ha denominado método de Cortes Posicionales, ha obtenido un *EER* del 24% y del 25% respectivamente sobre la muestra sin ruido y la muestra de documentos bancarios. Estos resultados son similares a los obtenidos por los enfoques previos, lo que demuestra la capacidad de las Redes de Compresión para abordar este problema.

### 6.3.2 Conclusiones particulares a la verificación usando *snakes*

Se ha propuesto un método, junto con diferentes variantes del mismo, de verificación de firmas basado en la técnica de ajuste de *snakes* que constituye la principal aportación de esta Tesis. El método, en su fase de aprendizaje, construye un *snake* para cada individuo a partir de una firma modelo. Luego, en su fase operativa, intenta adaptar el *snake* a la firma presente en la imagen a verificar, utilizando diferentes medidas del grado de adaptación para decidir si aceptar o rechazar la firma.

El método propone la realización de una serie de modificaciones a la formulación original de los *snakes* de Kass y otros [26], con objeto de hacerla compatible con el problema de la verificación *off-line* de firmas. Estas modificaciones se han realizado de manera incremental. En primer lugar se han modificado las funciones de energía externa de los *snakes* para que funcionasen sobre las imágenes bitonales de las firmas, utilizando un mapa de potencial que mide la distancia de cada punto del *snake* a los píxeles activos de la imagen a la firma. A continuación, se ha incluido en la formulación de la energía externa una tercera dimensión basada en el ángulo de la tangente aparente a cada punto del trazo de la firma. También, se han modificado las funciones de energía interna, convirtiendo el *snake* en un modelo con memoria de su forma. Además, se ha introducido una formulación borrosa que simplifica la configuración del sistema y mejora los resultados. Finalmente, se han propuesto una serie de características discriminantes que combinadas con una Red de Neuronas se han utilizado para medir el grado de adaptación del *snake* a las firmas a verificar.

Con objeto de conseguir la completa automatización de esta propuesta, se ha desarrollado un método que permite la generación automática del *snake* a partir de la imagen digital de una firma.

El *EER* obtenido por esta propuesta ha sido de un 12% y de un 16% sobre la muestra sin ruido y sobre la muestra de documentos bancarios respectivamente. Estos resultados mejoran

significativamente aquéllos obtenidos por los métodos precedentes, manteniendo todas las restricciones impuestas relativas a las condiciones realistas.

## 6.4 Resumen de publicaciones originadas por esta Tesis

Esta Tesis, en sus diferentes evoluciones, ha dado lugar a varios trabajos publicados en varias revistas y en diversos congresos nacionales e internacionales. A continuación se recogen las publicaciones del autor derivadas de los métodos presentados en esta Tesis, junto con una breve descripción de las mismas.

- En el año 2003, en el congreso internacional “*IEEE Workshop on Neural Networks for Signal Processing*” celebrado en Tolouse se presentó el trabajo “*Robust Off-line Signature Verification using Compression Networks and Positional Cuttings*”. En este trabajo se describían los resultados del método de Cortes Posicionales [79].
- En 2004 en Vigo se presentó un trabajo en el “*2nd COST Workshop on Biometrics on the Internet*” titulado: “*Simplified Stroke-based Approach for Off-line Signature Recognition*”. En este trabajo se evalúan diferentes características para el reconocimiento de firmas [190].
- En el año 2005, dentro del “*Simposio de Reconocimiento de Formas y Análisis de Imágenes*” del 1er congreso nacional CEDI, se presentó un primer trabajo basado en *snakes* titulado “*Verificación Off-Line de Firmas Manuscritas: Una Propuesta basada en Snakes Paramétricos*”. En este trabajo se presentan algunos resultados preliminares y se analizan los problemas inherentes a los *snakes* al aplicarlos a la verificación de firmas (ya que los *snakes* se han aplicado tradicionalmente al problema de segmentación de imágenes) [191].
- En el año 2006, dentro de la 11 Conferencia Internacional IPMU se presentó el trabajo “*A Hybrid Approach using Snakes and Fuzzy Modelling for Off-Line Signature Verification*”. En este trabajo se introduce un modelo borroso preliminar para el método basado en *snakes* [192].
- En el 2006, dentro del “*10th International Workshop on Frontiers in Handwriting Recognition*”, se presentó otro trabajo basado en *snakes* que bajo el título “*Combining Snakes and Neural Networks for Off-Line Signature Verification*” introducen las Redes de Neuronas como clasificador en la etapa final de verificación [123].
- En 2006 se presentó un trabajo resumen titulado “*Verificación off-line de firmas manuscritas*” en el “*Campus Multidisciplinar en Percepción e Inteligencia (CMPI '06)*” [193].
- También en el 2006 se publicó en la revista *Engineering Applications of Artificial Intelligence* un trabajo presentado en el 2005 titulado “*Support Vector Machines versus Multi-layer Perceptrons for Efficient Off-line Signature Recognition*”. En este trabajo se describe el proceso de generación de muestras sintéticas de firmas y se comparan los resultados del enfoque de verificación basado en Redes de Neuronas *MLP* con otro basado en Máquinas de Vectores Soporte (SVM) [194].
- En el año 2007, en el “*7th International Conference on Intelligent Systems Design and Applications*”, se profundiza en el enfoque borroso de los *snakes*, introduciendo además la tercera dimensión angular en el espacio de búsqueda, en un trabajo titulado “*Introducing Fuzziness on Snake Models for Off-Line Signature Verification*” [195].
- Finalmente, en el 2008, la revista *Fuzzy Sets and Systems* ha aceptado y está pendiente de publicación un trabajo titulado “*Fuzzy Shape-Memory Snakes for the Automatic Off-line Signature Verification Problem*” que amplía y formaliza el enfoque borroso de verificación basada en *snakes* presentada en trabajos precedentes.

## 6.5 Futuros trabajos

En esta sección se esbozan futuros trabajos que pretenden dar continuación a la línea de investigación abordada en esta Tesis Doctoral. Dichos trabajos se centran en tareas de ingeniería y tareas de investigación propiamente dicha.

Las tareas de ingeniería se centrarán en la puesta en producción de un sistema que resuelva de manera efectiva el problema de la verificación utilizando una combinación de los métodos propuestos. Los principales problemas existentes, desde este punto de vista, son debidos a los tiempos de ejecución de los métodos propuestos. Por ejemplo, los algoritmos de entrenamiento de una Red de Compresión para cada individuo en un *Pentium 4* a 3Ghzs con 1Gb de memoria, requieren entre uno y dos minutos por firma, y los tiempos de convergencia del *snake* unos 30 segundos en promedio.

Entre las tareas de investigación se propone avanzar en:

- Una propuesta de nuevos métodos de eliminación de ruido con estructura (como sellos o textura). Estos métodos producirían una inmediata mejora en los algoritmos de segmentación, lo que en consecuencia, mejoraría los resultados de verificación.
- El estudio de técnicas de combinación de clasificadores que permitiesen mejorar los resultados de verificación obtenidos.
- La ampliación del conjunto de características discriminantes utilizado en el enfoque basado en características.
- La mejora del sistema de generación de muestra sintética con la introducción de nuevos tipos de transformaciones. En especial, aquéllas de carácter elástico, como por ejemplo alguna transformación que cambie el tamaño de las filigranas de la firma.
- El diseño de un método que no necesite para el aprendizaje una firma modelo capturada en condiciones especialmente controladas. Se trata de poder utilizar como modelo cualquier firma de un individuo, quizás contenida en un documento firmado con presencia de ruido.
- Por último, aunque ya se ha comentado que tiene poco interés, desde un punto de vista económico, el abordaje del problema de los falsificadores habilidosos es un reto interesante desde el punto de vista de la visión artificial. Un primer paso para ello consistiría en ampliar la base de datos para que incluyese imágenes de firmas falsificadas por personas especialmente adiestradas en dicha labor.

## Anexo A Introducción a los *snakes*

Este anexo presenta una breve introducción formal a los *snakes* o *modelos de Contornos Activos*, centrándose en la propuesta original realizada por Kass, Witkin y Terzopoulos en 1988.

### A.1 Los *snakes*

Un *snake* [26] es un tipo de modelo de Contorno Activo [65], que consiste básicamente en un contorno abierto o cerrado que se mueve sobre una imagen a la que trata de adaptarse. Asociado a este Contorno Activo se define una función de energía que tiene una componente interna y otra externa. En la formulación original de Kass y otros [26], la componente interna de la energía incorpora ciertas características de elasticidad y rigidez al contorno. Mientras, la componente externa añade la influencia de la imagen sobre la que se mueve el Contorno Activo. Utilizando estas componentes, un algoritmo iterativo mueve el Contorno Activo sobre la imagen buscando una posición de mínima energía. En este movimiento se aprecia un desplazamiento característico del contorno (serpenteo) que da nombre al *snake* (serpiente en inglés)

Definiendo la posición del *snake* sobre una imagen bidimensional como un conjunto infinito de puntos  $v(s)=(x(s),y(s))$ , donde  $s$  es un parámetro de posición que varía entre 0 y 1, se puede escribir la función de energía del *snake* como:

$$E_{snake} = \int_0^1 E(v(s)) ds = \int_0^1 [E_{inter}(v(s)) + E_{imagen}(v(s)) + E_{cons}(v(s))] ds \quad (A.1)$$

En esta fórmula  $E_{inter}$  representa la energía interna debida a la relación entre los puntos de control del *snake*,  $E_{imagen}$  representa la energía externa relativa a la posición que ocupan los puntos de control del *snake* dentro de la imagen, y finalmente  $E_{cons}$  representa un término de energía asociado a otras condiciones externas opcionales (en la formulación original de Kass y otros no se le da uso alguno).

#### A.1.1 Energía interna

En la formulación original [26] para  $E_{inter}$  se propone la siguiente ecuación:

$$E_{inter} = \frac{1}{2}(\alpha(s)[x'_s(s) + y'_s(s)] + \beta(s)[x''_{ss}(s) + y''_{ss}(s)]) \quad (A.2)$$

En la fórmula,  $x'_s$  e  $y'_s$  representan la derivada primera respecto al parámetro de posición  $s$ . Igualmente,  $x''_{ss}$  e  $y''_{ss}$  representan las derivadas parciales segundas. Por otro lado, los coeficientes  $\alpha(s)$  y  $\beta(s)$  corresponden respectivamente a la importancia de la elasticidad y a la rigidez del *snake*. Cuanto más alto es el valor de  $\alpha(s)$  mayor tendencia tiene el *snake* a contraer su longitud. Por otro lado, cuanto más elevado es el valor de  $\beta(s)$  mas tendencia tiene el *snake* a enderezar su forma (si no es cerrado), o a volverse convexo (si es cerrado), haciéndolo en general más suave y menos anguloso.

Otras formulaciones propuestas posteriormente por Davison y otros [178] tratan de solventar diferentes tipos de problemas relacionados con el comportamiento del *snake* ante situaciones particulares. En general, casi todas las formulaciones se basan en ecuaciones de tipo muelle.

### A.1.2 Energía de la imagen o externa

Asociado a cada punto de la imagen, sobre la que se mueve el *snake*, se define una función de energía llamada externa. Esta función de energía debe asignar valores de menor energía a aquellos puntos donde se desee que se sitúe finalmente el *snake*. Además, se suele dotar a los píxeles de la imagen, sobre la que se ajusta el *snake*, de valores de gradiente que convergen hacia estos puntos de mínima energía, para fomentar el movimiento del *snake* en dicha dirección.

En su formulación original, el algoritmo de *snakes* propuesto por Kass y otros [26], contempla términos que atraen al Contorno Activo hacia las líneas, los bordes y las terminaciones presentes en la imagen. Estos elementos topológicos deben ser previamente segmentados en la imagen sobre la que se sitúa el *snake*. Así, la energía de la imagen se formula como:

$$E_{imagen} = E_{líneas} + E_{bordes} + E_{term} \quad (A.3)$$

Para atraer el *snake* hacia las líneas se propone utilizar la función de la imagen por si misma:

$$E_{líneas} = w \cdot I(x, y) \quad (A.4)$$

Para atraer el *snake* hacia los bordes se propone la función gradiente:

$$E_{bordes} = -|\nabla I(x, y)|^2 \quad (A.5)$$

Finalmente, para atraer el *snake* hacia los puntos de terminación, se propone utilizar la curvatura de las líneas de nivel sobre la imagen suavizada  $C$  mediante un filtro de Gauss.

$$E_{term} = \frac{\partial \theta}{\partial n_{\perp}} \quad (A.6)$$

donde  $\theta = \arctg(Cy/Cx)$  y  $n_{\perp} = (-\sin(\theta), \cos(\theta))$

## A.2 Minimización de la energía del *snake*

En la búsqueda de un mínimo de energía para el *snake* se hacen múltiples simplificaciones. En primer lugar, se discretiza el contorno, de manera que pasa de ser una función continua a estar compuesto de un conjunto finito de  $n$  puntos de control  $\{v_i\}_{i=1}^n$  que forman parte de una línea poligonal o en algunos casos de un *spline*. En segundo lugar, se hacen constantes los coeficientes asociados con las energías internas ( $\alpha(s)$  y  $\beta(s)$ ). Finalmente, se busca el valor de mínima energía mediante un algoritmo iterativo. Éste permite el cálculo de las sucesivas posiciones de los puntos de control del *snake* en el tiempo  $\{v_i(t)\}_{i=1}^n$ , obteniendo finalmente un problema variacional.

Amini y otros [182] proponen el uso de la técnica de programación dinámica para evitar la costosa evaluación de las múltiples posibilidades asociadas a cada movimiento del *snake*, siendo éste el enfoque utilizado por nuestra implementación. En el método propuesto el cálculo de la energía definida en la ecuación A.1 puede descomponerse en etapas sucesivas según la siguiente formulación:

$$E(v_1, v_2, \dots, v_n) = E_1(v_1, v_2) + E_2(v_2, v_3) + \dots + E_{n-1}(v_{n-1}, v_n) \quad (A.7)$$

A su vez, el cálculo de cada término de energía puede descomponerse, utilizando programación dinámica discreta, en una secuencia de funciones de una variable  $s_p$ , donde el conjunto  $\{s_i\}_{i=1}^n$  se obtiene según el siguiente sistema de ecuaciones de recurrencia:

$$\begin{aligned} s_1(v_2) &= \min_{v_1} \{E_1(v_1, v_2)\} \\ s_3(v_4) &= \min_{v_3} \{s_2(v_3) + E_3(v_3, v_4)\} \\ \min_{v_1, \dots, v_n} E(v_1, \dots, v_n) &= \min_{v_{n-1}} \{s_{n-2}(v_{n-1}) + E_{n-1}(v_{n-2}, v_{n-1})\} \end{aligned} \quad (\text{A.8})$$

y finalmente:

$$\min_{v_1, \dots, v_n} E(v_1, \dots, v_n) = \min_{v_{n-1}} \{s_{n-2}(v_{n-1}) + E_{n-1}(v_{n-2}, v_{n-1})\} \quad (\text{A.9})$$

Para poder evaluar no sólo el primer sumando sino toda la ecuación A.2, hay que tener en cuenta que la derivada segunda involucra a un tercer vértice, por lo que la descomposición de energías debe hacerse según:

$$E(v_1, v_2, \dots, v_n) = E_1(v_1, v_2, v_3) + E_2(v_2, v_3, v_4) + \dots + E_{n-1}(v_{n-2}, v_{n-1}, v_n) \quad (\text{A.10})$$

donde:

$$E_{i-1}(v_{i-1}, v_i, v_{i+1}) = E_{img}(v_i) + E_{inter}(v_{i-1}, v_i, v_{i+1}) \quad (\text{A.11})$$

Por ello, la secuencia de funciones  $\{s_i\}_{i=1}^n$  contiene dos variables y para el caso de la función (A.2) tendrá la siguiente forma:

$$s_i(v_{i+1}, v_i) = \min \{s_{i-1}(v_i, v_{i-1}) + \alpha |v_i - v_{i-1}|^2 + \beta |v_{i+1} - 2v_i + v_{i-1}|^2 + E_{ext}(v_i)\} \quad (\text{A.12})$$



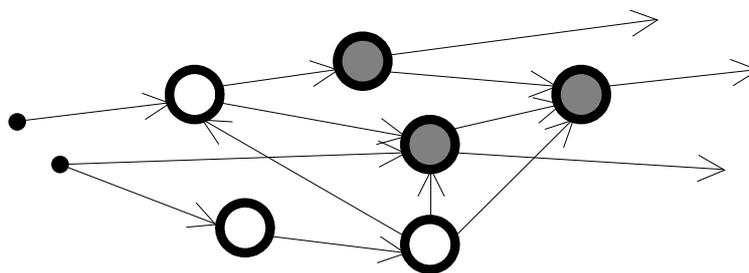
## Anexo B Redes Neuronales: Perceptrones Multicapa y Redes de Compresión

Este anexo realiza una breve descripción de las Redes de Neuronas y sus elementos relacionados, centrándonos a continuación en los tipos de redes usados en esta Tesis Doctoral: los Perceptrones Multicapa y las Redes de Compresión.

### B.1 Introducción a las Redes de Neuronas artificiales

El funcionamiento de las Redes de Neuronas artificiales tiene su base en la interacción de unos elementos, las células nerviosas o *neuronas*, a través de unas conexiones llamadas *conexiones sinápticas*. Tales conexiones tienen su origen en una neurona y su destino en otra neurona. Además, estas conexiones tienen asociadas un valor real variable que determina la influencia de la misma y que se denomina *peso*. Como una misma neurona puede ser destino de varias conexiones sinápticas, las entradas que una neurona recibe se procesan, mediante la función de activación, para definir el *estado de activación* de la célula nerviosa. A partir de la función de activación se genera la salida de la neurona mediante la función de salida<sup>12</sup>.

Se pueden distinguir tres tipos principales de neuronas artificiales (ver Figura B.1): de entrada, de salida y ocultas. Las *neuronas de entrada*, son aquellas por donde se introducen los datos, a partir de los cuales la red producirá la respuesta. Las de entrada son las únicas neuronas cuyo estado de activación se impone directamente desde el exterior. Las neuronas de salida son las que con su estado de activación definen la respuesta de la red ante el estímulo suministrado a su entrada. Puede haber, por último, una serie de neuronas intermedias, llamadas neuronas ocultas, que procesan y propagan información desde la entrada hasta la salida. Como se puede apreciar, por esta descripción que se ha realizado de las Redes de Neuronas, la salida de una red sólo depende: de la entrada, de las conexiones que existan entre las células y de los pesos que éstas tengan asociados.



**Figura B.1.-** Ejemplo de una Redes de Neuronas genérica. Se presentan en negro y más pequeñas las neuronas de entrada y, sombreadas las de salida. El resto son neuronas intermedias u ocultas.

<sup>12</sup> En su definición más general se definen tres funciones asociadas a cada neurona. La función de entrada, la función de activación y la función de salida, aunque como la función de salida suele ser la función identidad en general no se tiene en cuenta. Hay autores que denominan función de transferencia al conjunto formado por la función de activación y la función de salida.

### B.1.1 El proceso de aprendizaje de una red

El cálculo de los pesos que deben poseer las conexiones de una Red de Neuronas para que realice la función que se desea se conoce como aprendizaje. En redes triviales el ajuste de estos pesos se puede hacer manualmente, pero la tarea se vuelve imposible en redes de una complejidad mínima. Se han inventado diferentes algoritmos que realizan esta tarea de manera más o menos automática sobre distintos tipos de redes, aunque todos tienen inconvenientes<sup>13</sup>.

Desde un punto de vista puramente matemático, dejando de lado lo que las entradas o las salidas de la red significan en este contexto, el problema consiste en encontrar el valor de unas incógnitas (los pesos) para que unas funciones (las ecuaciones de las neuronas) obtengan ciertos valores (la salida deseada) cuando se dan unos valores determinados a sus parámetros (las entradas a la red). En caso de no poder obtenerse exactamente el valor buscado se podría pensar en minimizar el error entre la salida deseada y la obtenida. Esto lleva al problema de encontrar el mínimo global de una función cualquiera, problema para el que en general no se conoce solución. Son aplicables sin embargo una infinidad de procedimientos numéricos, basados generalmente en métodos del gradiente, que desgraciadamente sólo permiten obtener soluciones locales a este problema. El algoritmo más popular es el de *aprendizaje por retropropagación*.

#### Propiedad de generalización de las Redes de Neuronas

Durante el entrenamiento las salidas de una Red de Neuronas se aproximan a ciertos valores objetivo. Esto puede bastar para ciertas aplicaciones, pero el mayor interés despertado por las Redes de Neuronas se debe a su capacidad para *generalizar*. Se llama generalizar a la propiedad que le permite generar salidas adecuadas frente a entradas que no se encuentran en el conjunto de entrenamiento y que por tanto la red no ha aprendido previamente. La generalización no siempre es posible. Se destacan tres condiciones necesarias para que pueda conseguirse generalización:

- Que las entradas contengan suficiente información para que se pueda alcanzar las salidas deseadas al menos en el entrenamiento. Si la red no es capaz de aprender algo difícilmente será capaz de generalizar posteriormente.
- Que la función a interpolar sea, en cierto sentido, suave. Es decir, que un pequeño cambio en la entrada debe producir un pequeño cambio en la salida. Esto implica que la función que se emula sea continua hasta la primera derivada. Por ello las salidas de funciones como los generadores de números pseudo-aleatorios o algoritmos como los de encriptación no pueden ser emuladas con una red de neuronas.
- Que el conjunto de entrenamiento sea suficientemente grande. Esto se debe a que la generalización se produce en dos sentidos: interpolando y extrapolando. La interpolación en general da buenos resultados, pero la extrapolación no, por ello se deben tener suficientes casos de entrenamiento de manera que se reduzca la extrapolación.

#### Conjuntos necesarios para entrenar una red

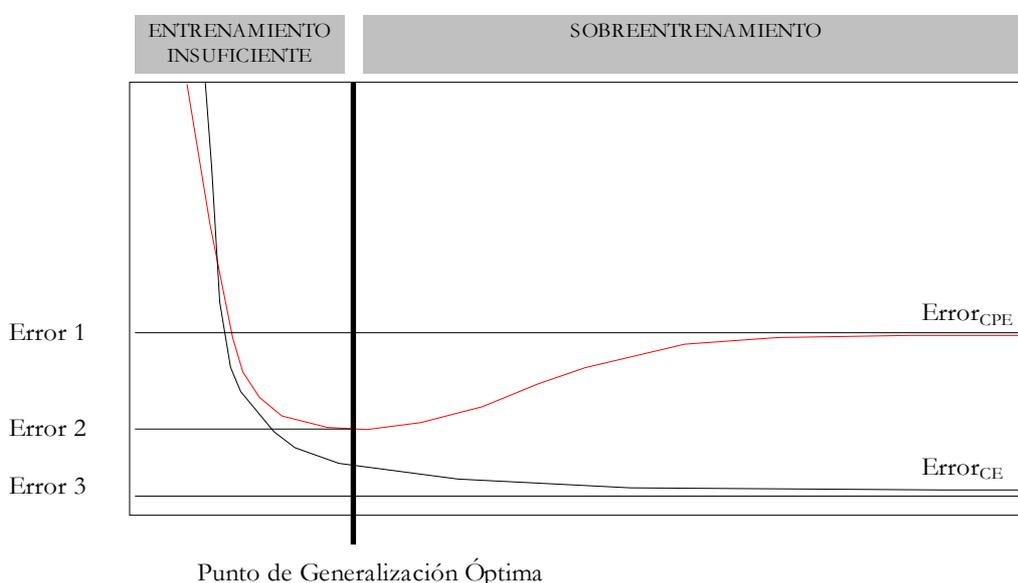
Para el ajuste de los pesos de las Redes de Neuronas, se precisan unos patrones con las entradas a la red y otros con las salidas deseadas frente a esos estímulos, constituyendo lo que se llama conjunto de entrenamiento (CE). Se usará este conjunto para entrenar la red, mediante un procedimiento iterativo, que consiste en variar los pesos buscando minimizar la función de error que se haya determinado.

---

<sup>13</sup> El proceso de aprendizaje de un conjunto de neuronas reales constituye un tema por descubrir para la biología actual.

Opcionalmente se distingue otro conjunto, denominado conjunto de test del entrenamiento (CTE), que se usa para saber en qué momento se debe detener el entrenamiento. Una red sobreentrenada ha memorizado demasiado, y es incapaz de generalizar a nuevos patrones distintos a los del entrenamiento. O de otra forma, un sobreentrenamiento hace que la red ajuste los pesos en exceso, curvando mucho, para ello, la superficie formada por los pesos, de manera que entradas próximas, pero no iguales, obtienen resultados muy diferentes. Mientras que con un entrenamiento menos exhaustivo las entradas próximas darían resultados similares, pues la superficie formada por los pesos es más suave y se produce un proceso de interpolación. El *punto de generalización óptimo* sería aquel a partir del cual la red memoriza en demasía, y antes del cual la red puede mejorar sus resultados con los conjuntos CTE y CE.

Una vez terminado el entrenamiento se debe disponer de un tercer conjunto, llamado conjunto de validación (CV), con el que se evalúa la red. Es importante que CV se use sólo como comprobación final, pero nunca para entrenar la red. Si no, la red aprenderá este conjunto también, y no se sabrá si es capaz de generalizar.



**Figura B.2.-** Ejemplo del entrenamiento de un Red de Neuronas. Se presentan en negro la curva correspondiente al error al entrenar y en rojo la correspondiente al error al validar. El punto de generalización óptimo viene marcado por el momento en el que el error de validación comienza a crecer.

## B.2 Estructura del Perceptrón Multicapa

Cumpliendo el esquema que se ha planteado se pueden construir multitud de modelos. Uno de los más sencillos es aquél en cual el estado de activación de cualquier célula  $j$  es función de la entrada total a la misma, siendo la entrada total el resultado de sumar todas las salidas correspondientes a las diferentes neuronas que son entradas de  $j$  multiplicadas cada una por el peso de la conexión que las une y restándole al total una variable que sólo depende de la neurona  $j$ . Llamaremos a esta variable *umbral* debido a la función inhibidora que puede ejercer. Así la entrada total a la neurona  $j$  toma la forma de la expresión B.1.

$$entrada_j = -umbral_j + \sum_i Salida_i \cdot Peso_{i \rightarrow j} \quad (B.1)$$

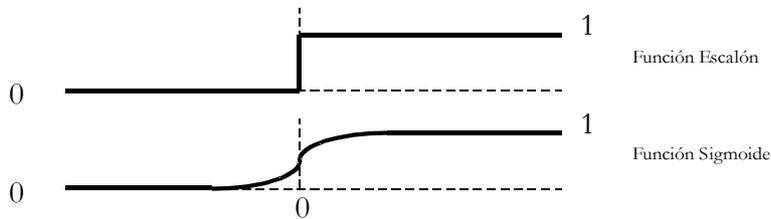
Para simplificar los cálculos suele cambiarse el umbral por una entrada de peso igual a *umbral*, conectando este axón a una neurona de entrada adicional que siempre tendrá estado de activación 1. De esta forma la entrada total a la neurona *j* será:

$$entrada_j = \sum_i Salida_i \cdot Peso_{i \rightarrow j} \quad (B.2)$$

De igual forma, las funciones de activación clásicas son, en el caso discreto, la *Función Escalón* (B.3), y en el caso continuo, la *Función Sigmoide* (B.4). Como se puede ver en la Figura B.3, estas funciones hacen que si la entrada total es negativa la salida de la neurona sea 0 (o próxima a 0) y que si es positiva la salida sea 1 (o próxima a 1).

$$salida_j = \begin{cases} 0 \Leftrightarrow entrada_j < 0 \\ 1 \Leftrightarrow entrada_j \geq 0 \end{cases} \quad (B.3)$$

$$salida_j = \frac{1}{1 + e^{-entrada_j}} \quad (B.4)$$



**Figura B.3.-** Aspecto de las Funciones de Activación Escalón (arriba) y Sigmoide (abajo).

Así, el modelo más sencillo consistirá en una red compuesta por una única neurona, en la que la salida viene determinada por la sigmoide (B.4) de la entrada total. Es sencillo demostrar que este esquema es idéntico al del Clasificador Euclídeo. Así, una red con una sola capa puede separar conjuntos de patrones mediante un hiperplano (ver Figura B.4).

Un paso por encima en complejidad se encuentra una red, carente de neuronas ocultas, compuesta por varias de estas unidades que operan de forma independiente, es decir, que la salida de cualquiera de ellas no influye en el funcionamiento del resto. Esta configuración, conocida como *Perceptrón*, equivale al uso de varios Clasificadores Euclídeos. El *Perceptrón Multicapa* es la red compuesta por varios niveles sucesivos de Perceptrones (ver Figura B.5), de manera que las salidas de las unidades de una capa son entradas de las unidades del nivel siguiente. En el siguiente punto se estudiarán las ventajas que aporta esta estructura multicapa.

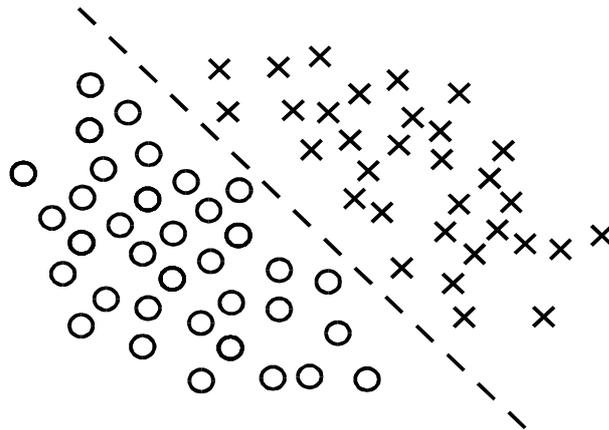


Figura B.4.- Patrones 2D separados por un hiperplano (en este caso una recta).

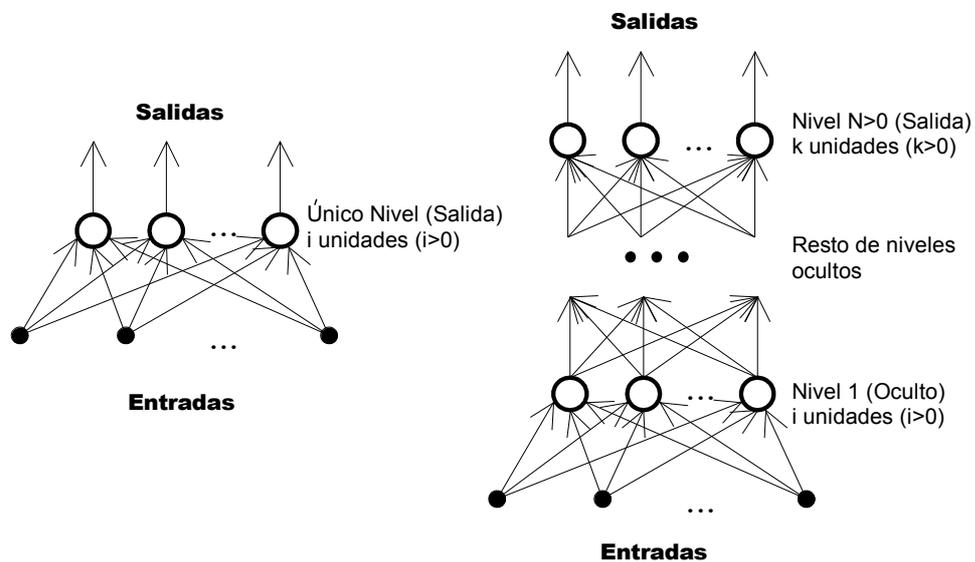


Figura B.5.- Esquemas de los tipos de Perceptrones. A la izquierda el de un Perceptrón y a la derecha el de un Perceptrón Multicapa.

### B.3 Propiedades del Perceptrón Multicapa

El Perceptrón Multicapa tiene una serie de propiedades que lo hacen especialmente interesante al usarlo como clasificador. Éstas son:

- Robustez frente al ruido aleatorio. Lo que conlleva la selección automática de las características importantes dentro del vector de características que es entrada a la red.
- Separación de regiones complejas dependiendo de la estructura de la red. Una red puede separar clases a pesar de que la hipersuperficie de separación entre ambas sea arbitrariamente compleja, o incluso no sea única.
- Capacidad de generalización. Los algoritmos de descenso del gradiente han demostrado en la práctica que obtienen buenas interpolaciones para patrones con los que no se ha entrenado el sistema.

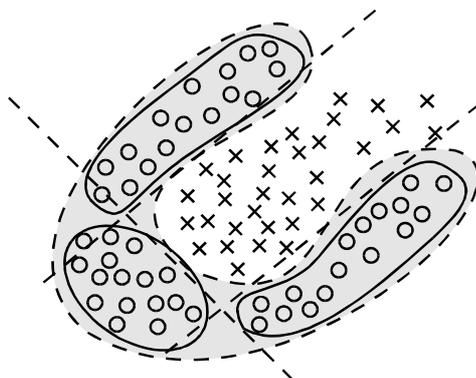
### B.3.1 Selección del número de capas ocultas

Se ha dicho que una red con una neurona de salida y sin unidades ocultas, es equivalente a un Clasificador Euclídeo, siendo capaz de discriminar mediante un hiperplano dos conjuntos de patrones linealmente separables.

Para poder discriminar dos conjuntos sea cual sea la disposición de sus patrones se puede utilizar una red con una sola capa oculta. En los siguientes párrafos se ofrece una demostración informal de este hecho.

Primeramente debe notarse que siempre es posible agrupar los patrones de cualquier clase formando subconjuntos convexos. Estos subconjuntos convexos tienen la particularidad de que pueden separarse linealmente del resto de elementos mediante varios hiperplanos (véase la Figura B.6). Así, un patrón interior a un subconjunto convexo queda clasificado por la intersección de las regiones definidas por varios hiperplanos.

Es fácil demostrar que este primer paso de separación mediante hiperplanos se puede resolver con una red Perceptrón sin unidades ocultas, con una neurona de salida por hiperplano necesario. Cuando, ante un patrón, todas las neuronas de salida de una de estas intersecciones se activan simultáneamente indica que ese patrón pertenece a ese subconjunto convexo.



**Figura B.6.-** Los patrones de una muestra siempre se pueden agrupar en regiones convexas.

El siguiente paso consiste en la unión de los resultados de estos subconjuntos convexos en las clases que les dieron origen. Se puede demostrar que esta tarea la puede realizar una red Perceptrón, sin unidades ocultas y con una neurona de salida por cada clase a discriminar. En particular, una neurona de salida, correspondiente a una clase  $C$ , tendrá un umbral  $H$ . Para que esta neurona se active sólo cuando un patrón corresponda a la clase  $C$ , el umbral  $H$  puede tomarse ligeramente menor al número de hiperplanos necesarios para separar cada uno de los subconjuntos convexos de la clase  $C$ .

Por tanto, concatenando las dos redes descritas, es decir, construyendo una red con una capa oculta y tantas neuronas de salida como clases, se pueden separar regiones con forma cualquiera.

Así, al menos teóricamente, el comportamiento de cualquier red de dos o más capas se puede obtener con una red de sólo una capa oculta. Sin embargo, los algoritmos de aprendizaje que se conocen no garantizan que esta búsqueda tenga éxito. Es más, la mayoría de las veces estos algoritmos obtienen buenos resultados cuando se disponen varias capas ocultas, y malos resultados cuando sólo se dispone una capa oculta.

## B.4 Algoritmos de aprendizaje para el Perceptrón Multicapa

Resumiendo lo visto, un algoritmo para el ajuste de los pesos de una Red de Neuronas artificiales debe minimizar alguna función que esté relacionada con el error cometido en las unidades de salida al presentarle una entrada. Si para cada configuración posible de los pesos de una red se conoce el error que comete a la salida al presentarle una entrada determinada, el error puede verse como una superficie cuya altura depende de las coordenadas de los pesos en que se tenga.

La búsqueda del mínimo de estas superficies ha dado lugar a varios algoritmos basados en su mayoría en métodos de descenso del gradiente. Entre todos destaca, por su simplicidad y buenos resultados, el algoritmo de retropropagación del gradiente, descubierto en 1974 por Werbos y redescubierto de manera independiente en 1985 por Le Cun y Parker. En la base del algoritmo de retropropagación del gradiente está la regla delta.

### B.4.1 La regla Delta

Para el Perceptrón existe un algoritmo, conocido como regla delta o algoritmo *LMS* (*Least Minimum Squares*), que cambia los pesos,  $peso_{ip}$ , en la dirección de máxima pendiente dentro de la superficie del error. Las diferentes componentes del vector de dirección de máxima pendiente vienen determinadas por las derivadas parciales según cada componente de la función error.

$$\Delta peso_{i \rightarrow j} = -\mu \frac{\partial Error}{\partial peso_{i \rightarrow j}} \quad (B.5)$$

En la ecuación anterior se aprecia que se ha añadido un término  $\mu$  cuya función es determinar la velocidad de variación de los pesos, este término se conoce como tasa de aprendizaje. Un valor de  $\mu$  pequeño puede hacer que se caiga fácilmente en mínimos locales, mientras que un valor de  $\mu$  grande puede hacer que se esté siempre oscilando, sin converger nunca. Desgraciadamente el concepto de “grande” y “pequeño” varía según la superficie de error que se esté considerando, la cual a priori es desconocida ya que depende de la muestra que se use para el entrenamiento, por lo que su ajuste constituye un proceso experimental.

### Elementos de la red

De la ecuación B.5 se deduce que se debe disponer de una medida para el error. La medida más común de éste consiste en la suma de las diferencias cuadráticas entre los valores obtenidos y los deseados, en las  $numN$  neuronas de salida, después de presentar cada patrón de entrenamiento. Esta fórmula (B.6) se conoce como error cuadrático medio.

$$Error = \frac{1}{numN} \sum_{N=1}^{numN} (salida_N - deseada_N)^2 \quad (B.6)$$

Donde se ha tomado la salida de una neurona  $j$  como una función  $F$  de la entrada total a la misma (B.2), de acuerdo con la definición de función de activación y función de salida.

$$salida_j = F(entrada_j) \quad (B.7)$$

### Cálculo del incremento del peso

Si se aplica sobre (B.6) la regla de la cadena, se puede escribir:

$$\frac{\partial Error}{\partial peso_{i \rightarrow j}} = \frac{\partial Error}{\partial entrada_j} \frac{\partial entrada_j}{\partial peso_{i \rightarrow j}} \quad (B.8)$$

De la ecuación B.2 deducimos:

$$\frac{\partial entrada_j}{\partial peso_{i \rightarrow j}} = salida_i \quad (B.9)$$

Definimos ahora:

$$cambio_j = \frac{-\partial Error}{\partial entrada_j} \quad (B.10)$$

Así que uniendo las fórmulas B.9 y B.10 se puede reescribir la expresión B.5, que determina la variación de los pesos en su búsqueda del mínimo, como:

$$\Delta peso_{i \rightarrow j} = \mu cambio_j salida_i \quad (B.11)$$

El parámetro  $cambio_j$  se obtiene aplicando de nuevo la regla de la cadena:

$$cambio_j = \frac{-\partial Error}{\partial entrada_j} = \frac{-\partial Error}{\partial salida_j} \frac{\partial salida_j}{\partial entrada_j} \quad (B.12)$$

El segundo término de la última fórmula de la ecuación anterior equivale a derivar respecto de  $entrada_j$  la ecuación B.7.

$$\frac{\partial salida_j}{\partial entrada_j} = F'(entrada_j) \quad (B.13)$$

Si la neurona  $j$  es una neurona de salida, en la ecuación B.6 uno de los términos considerados será la misma neurona  $j$ . En tal caso el primer término de la ecuación B.12 puede calcularse directamente derivando en B.6 respecto de  $salida_j$ . Consecuentemente, la aplicación de esta fórmula para el error se realiza sólo en los casos en que la función de activación de las neuronas es continua y derivable.

$$\frac{-\partial Error}{\partial salida_j} = deseada_j - salida_j \Rightarrow \quad (B.14)$$

$$cambio_j = (deseada_j - salida_j) F'(entrada_j) \quad (B.15)$$

Este algoritmo constituye la regla delta y viene acompañado con un teorema que asegura la convergencia del método siempre que el problema de clasificación tenga solución. Este teorema no asegura, sin embargo, la convergencia hacia el mínimo absoluto de la superficie, sino que el descenso puede ser a un mínimo local de la superficie del error.

### Funciones de activación

Como se desprende de (B.15) la aplicación de la regla delta depende de la derivada de la función de activación. La función de activación más común es la sigmoide, pero también suelen usarse la función tangente hiperbólica y la función identidad.

Si la función de activación fuese la identidad se tiene:

$$salida_j = F(entrada_j) = entrada_j \quad (B.16)$$

Derivando:

$$F'(entrada_j) = 1 \quad (B.17)$$

Si la función es la sigmoide se puede escribir:

$$salida_j = F(entrada_j) = \frac{1}{1 + e^{-entrada_j}} \Rightarrow e^{-entrada_j} = \frac{1}{salida_j} - 1 \quad (B.18)$$

Derivando:

$$F'(entrada_j) = \left( \frac{1}{1 + e^{-entrada_j}} \right) = \frac{e^{-entrada_j}}{(1 + e^{-entrada_j})^2} = salida_j(1 - salida_j) \quad (B.19)$$

Según se ha visto la función de activación debe ser derivable, y si no lo es la regla delta se encuentra con un escollo. Éste se debe a la función de error que se está usando. Tomando otro tipo de función para el error se puede seguir aplicando el algoritmo a funciones no derivables.

Por ejemplo, en el caso de la función escalón  $F'(entrada_j)$  vale 0 en todo  $\mathcal{R}$  menos en el 0 que no existe. Esto hace que (B.13) no pueda calcularse. La solución es hacer que no sea necesario su cálculo, por ejemplo haciendo que el error tenga tal forma que  $cambio_j$  pueda calcularse directamente como la derivada del error respecto de la entrada. Para ello se define  $m$  como aquel valor umbral tal que, si la entrada es mayor que él entonces la salida es 1, y si la entrada es menor entonces la salida es 0. Definiendo el error para una neurona determinada con una función derivable respecto de la entrada como (B.27), se puede obtener (B.28) que permite usar la regla delta con la función escalón.

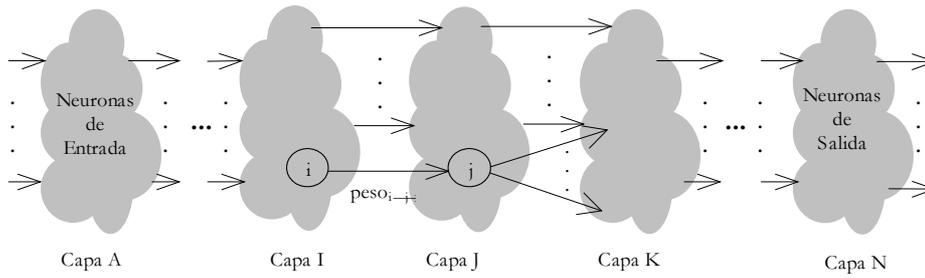
$$error = \begin{cases} 0 & \text{si la salida es correcta} \\ \frac{1}{2} \cdot (m - entrada_j)^2 & \text{si la salida es 0 y debería ser 1} \\ -\frac{1}{2} \cdot (m - entrada_j)^2 & \text{si la salida es 1 y debería ser 0} \end{cases} \quad (B.20)$$

$$cambio_j = \frac{-\partial Error}{\partial entrada_j} = \begin{cases} 0 & \text{si la salida es correcta} \\ \frac{entrada_j}{2} - m & \text{si la salida es 0 y debería ser 1} \\ -\frac{entrada_j}{2} + m & \text{si la salida es 1 y debería ser 0} \end{cases} \quad (B.21)$$

## B.4.2 Generalización de la regla Delta

El algoritmo de retropropagación supone una generalización de la regla delta de manera que pueda aplicarse a Perceptrones Multicapa. En estas redes las fórmulas (B.15) y (B.19) permiten el cálculo del incremento de los pesos para las neuronas de la última capa. Sin embargo, en las neuronas intermedias no se conoce el valor deseado para su salida, por lo que estas fórmulas no se pueden aplicar directamente.

La idea central de la retropropagación está en que el valor deseado para una neurona de una capa intermedia  $J$  (ver Figura B.7) puede derivarse del valor deseado para las neuronas de la siguiente capa  $K$ . La aplicación de esta idea de manera recursiva nos conducirá a la última capa para la que sí se conoce el valor deseado a la salida.



**Figura B.7.-** Esquema general de una conexión entre dos neuronas en un Perceptrón Multicapa. Se presenta el caso de una conexión cualquiera, que une la neurona  $i$  de la capa intermedia  $I$  y la neurona  $j$  de la capa intermedia  $J$ .

En un Perceptrón Multicapa, el error cometido por la red se puede escribir como una función  $E$  de las entradas a cualquier capa intermedia de la red, sin más que ir sustituyendo en (B.6) los términos *salida* por sus correspondientes de (B.7). Así para un nivel  $J$  cualquiera, de un Perceptrón Multicapa, en el que se encuentran las neuronas  $1, 2, \dots, j, \dots, numJ$  se puede expresar el error en función de las entradas a este nivel.

$$Error = E(entrada_1, entrada_2 \dots entrada_j \dots entrada_{numJ}) \quad (B.22)$$

Teniendo en cuenta la definición previa y usando la regla de la cadena se puede simplificar el primer término de la ecuación B.11 para cualquier neurona de la red.

$$\frac{-\partial Error}{\partial salida_j} = - \sum_{K=1}^{numK} \frac{\partial Error}{\partial entrada_K} \frac{\partial entrada_K}{\partial salida_j} \quad (B.23)$$

$$= - \sum_{K=1}^{numK} \frac{\partial Error}{\partial entrada_K} \cdot \frac{\partial \left( \sum_{J=1}^{numJ} peso_{J \rightarrow K} \cdot salida_J \right)}{\partial salida_j} \quad (B.24)$$

Realizando la derivada y usando el resultado de la ecuación B.12 se llega a:

$$= - \sum_{K=1}^{numK} \frac{\partial Error}{\partial entrada_K} \cdot peso_{J \rightarrow K} = \sum_{K=1}^{NumK} cambio_K \cdot peso_{j \rightarrow K} \quad (B.25)$$

Sustituyendo esta ecuación y la B.13 en la B.12 se obtiene:

$$cambio_j = F'(entrada_j) \cdot \sum_{K=1}^{numK} cambio_K \cdot peso_{j \rightarrow K} \quad (B.26)$$

Esta fórmula nos permite conocer el factor *cambio* para una capa siempre que se conozca el factor *cambio* para la capa siguiente. Este proceso tiene fin en la última capa de la que se conoce el valor cambio gracias a B.15. Constituye por tanto un método constructivo para ir calculando el incremento de los pesos que minimiza el error según B.6, y que exige que la función de activación de cada neurona sea derivable respecto de la entrada total a la misma.

Una vez conocidos estos resultados, se puede resumir el Algoritmo de Retropropagación en los pasos que se detallan en el listado adjunto.

- Algoritmo de Retropropagación del Gradiente -

```

1 -      Iniciar los pesos de las conexiones de la red con valores aleatorios
      pequeños.
2 -      Presentar uno de los conjuntos de entrada de la muestra a la red y
      calcular la salida que se obtiene.
3 -      Si no coincide la salida obtenida con la deseada ajustar los pesos
      como sigue:
4 -       $j = N$  ( $N$  es el número de neuronas)
5 -       $i = j$ 
6 -      while ( $j \neq 0$ )
7 -          while ( $i \neq 0$ )
8 -               $\Delta \text{ peso}_{i \rightarrow j}(t+1) = \mu \cdot \text{cambio}_j(t) \cdot \text{salida}_i(t)$ 
9 -               $\text{peso}_{i \rightarrow j}(t+1) = \text{peso}_{i \rightarrow j}(t) + \Delta \text{ peso}_{i \rightarrow j}(t+1)$ 
10 -              $i = i - 1$ 
11 -          $j = j - 1$ 
12 -
13 -          $\text{cambio}_j(t) = F'(\text{entrada}_j(t)) \cdot (\text{deseado}_j(t) - \text{estado}_j(t))$ 
14 -          $\text{cambio}_j(t) = F'(\text{entrada}_j(t)) \cdot \sum_k (\text{cambio}_k(t) \cdot \text{peso}_{j \rightarrow k}(t))$ 
15 -
16 -      Tomado  $k$  los valores de todas las neuronas por encima de la neurona  $j$ .
17 -      Si han coincidido todas las salidas obtenidas con las esperadas
      después de presentar todos las entradas de nuestro conjunto de
      entrenamiento terminar. En otro caso hacer  $t = t + 1$  e ir al paso 2.
      Como es posible que el proceso no logre obtener siempre la salida
      deseada frente al patrón de entrada, siempre se puede detener el
      proceso cuando alcance un valor de error que se considere aceptable.

```

En esta forma los pesos se modifican tras cada iteración constituyendo la variante *on-line*. También podrían sumarse cuando han pasado todos los conjuntos de entrada en lo que sería la variante *batch*. Obsérvese que  $\text{cambio}_j$  es una medida del error cometido en la neurona  $j$ . En este algoritmo este error depende del tipo de neurona. Si la neurona  $j$  es de salida,  $\text{cambio}_j$  tendrá en cuenta la diferencia entre el valor deseado y el obtenido:

### B.4.3 Las Redes de Compresión

Las Redes de Compresión se basan en un sistema autoasociativo lineal que fue expuesto por primera vez por Anderson y otros [196] y por Kohonen [175] de manera independiente.

#### La memoria autoasociativa

Sea una imagen dada como un vector columna  $x_k$  de dimensión  $p$ , donde  $p$  es el número de píxeles de la imagen. Por comodidad, además, asumiremos que tales vectores se han normalizado de manera que:

$$x_k^T x_k = 1 \quad (\text{B.27})$$

Si deseamos almacenar  $n$  imágenes en la memoria, se puede comenzar por representar el conjunto total de imágenes como una matriz  $M$  de dimensión  $p \times n$ , donde cada elemento  $x_{i,k}$  corresponderá a la intensidad del píxel  $i$  de la imagen  $k$ .

Por otro lado la memoria autoasociativa se representará como una matriz  $A$  de dimensión  $p \times p$ , donde el elemento  $a_{i,i'}$  representa la covarianza existente entre los píxeles  $i$  e  $i'$ . Esta matriz se conoce como matriz de correlación y de acuerdo con la nomenclatura descrita se puede expresar la matriz  $A$  como:

$$A = \sum_k x_k x_k^T = X X^T \quad (\text{B.28})$$

Construida la matriz  $A$  ya tenemos la memoria autoasociativa. Ahora, para obtener la imagen  $k$ , almacenada en la memoria, sólo se ha de posmultiplicar la matriz  $A$  por la imagen  $x_k$ . Se obtiene así  $\hat{x}_k$ , que corresponde a la estimación de la imagen  $k$  dada por la memoria.

$$\hat{x}_k = A x_k \quad (\text{B.29})$$

Cuando todos los vectores almacenados en la memoria sean ortogonales  $x_k$  será igual a  $\hat{x}_k$ . Por el contrario, si los vectores no son ortogonales, la memoria añadirá algún tipo de ruido al patrón correcto. En estos casos, métodos iterativos para la obtención de la matriz  $A$ , como la regla de Widrow-Hoff [197] obtienen mejores resultados. La calidad de esta reconstrucción puede medirse con el coseno del ángulo entre los vectores  $x_k$  y  $\hat{x}_k$ .

### Componentes principales contenidas en la memoria

Sea  $A$  una matriz semidefinida positiva. Podemos, según el teorema espectral, expresarla como el producto de 3 matrices, una de ellas diagonal.

$$A = \sum_r \lambda_r u_r u_r^T = U \Lambda U^T \quad (\text{B.30})$$

Donde  $U$  es la matriz de los autovectores  $u_r$  de  $A$ ,  $\Lambda$  la matriz diagonal positiva de los autovalores  $\lambda$  y  $R$  el rango de  $A$  (es decir, el número de autovectores con autovalores distintos de cero).

De (B.29) y (B.30) se puede deducir:

$$\hat{x}_k = \sum_r \lambda_r s_{k,r} u_r \quad (\text{B.31})$$

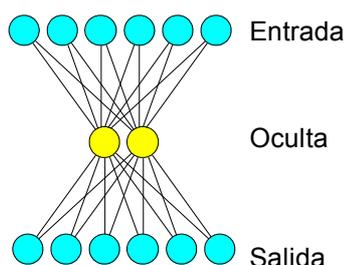
donde  $s_{k,r}$  son las proyecciones del vector  $x_k$  sobre los autovectores de  $A$ :

$$s_{k,r} = u_r^T x_k \quad (\text{B.32})$$

Lo importante de esta situación es la interpretación de los autovalores, ya que se corresponden con la varianza direccional de las proyecciones sobre los autovectores. Así, la magnitud de cada autovalor determina la cantidad de varianza representada por su correspondiente autovector. Un mayor valor del autovalor implica que el autovector asociado explica más varianza de la muestra. Además, si los autovectores son ortogonales, la cantidad de varianza explicada por un conjunto de autovectores es igual a la suma de sus autovalores.

### Redes de compresión

Una de las propiedades de las Redes de Neuronas consiste en que, a través del entrenamiento, en las unidades ocultas se construye una representación interna de la entrada para que se produzcan las salidas deseadas. Para un patrón de dimensión  $p$ , se puede disponer una red con  $p$  unidades de entrada,  $p$  unidades de salida y  $h$  unidades ocultas. Si se entrena para obtener a la salida el mismo patrón que se presenta a la entrada, queda claro que se construye una representación del patrón en las unidades ocultas.



**Figura B.8.-** Formato de una Red de Compresión.

Si además el número de neuronas de  $h$  es menor que el de  $p$ , se está obligando a la red a codificar la entrada en un subespacio de dimensión menor, que conserva, sin embargo, la mayor parte de la información. Es decir, se fuerza a la red a capturar las regularidades y redundancias presentes en los patrones, extrayendo un conjunto de características menos correlacionadas que las originales (que serían todos los puntos la imagen). Por esto, este tipo de Redes de Neuronas se conocen como Redes de Compresión, Redes codificadoras o Redes Autocodificadoras [174].

Si las unidades ocultas son lineales en definitiva lo que tenemos es un polinomio con tantas incógnitas como pesos tenga la red. La mejor solución al problema de representar la información de la imagen en un polinomio de este tipo corresponde a unos valores de los pesos que minimizan el error cuadrático medio [174]. Como hemos visto, esta solución es la que se obtiene con una memoria autoasociativa. Pero también una Redes de Neuronas, entrenada con el algoritmo de retropropagación, trata de minimizar esta suma. En particular Baldin y Hornik han demostrado [176] que la superficie del error para este problema no tiene mínimos locales y tiene un único mínimo global que coincide con  $\mathcal{A}$  (solución de la memoria autoasociativa). Por tanto al iterar una Redes de Neuronas de Compresión, con retropropagación, sólo podemos aproximarnos a la misma solución que se obtiene con el análisis de componentes principales. Además, se ha demostrado que estas Redes Neuronales, en sus  $h$  unidades ocultas, aproximan las  $h$  componentes principales de la imagen que poseen mayores autovalores [198].



## Anexo C Diagramas UML de Actividad

El *Lenguaje Unificado de Modelado* (*Unified Modelling Language* o *UML*) es un lenguaje estándar para describir el software. Los Diagramas de Actividades son uno de los cinco tipos de diagramas *UML* que se utilizan para el modelado de los aspectos dinámicos de los Sistemas de Información [199].

Un Diagrama de Actividad es fundamentalmente un diagrama de flujo que muestra la secuencia de control entre diferentes acciones, mostrando tanto la concurrencia como las bifurcaciones de control. Los Diagramas de Actividades pueden servir para visualizar, especificar, construir y documentar la dinámica de una sociedad de objetos, o pueden emplearse para modelar el flujo de control de una operación.

La ejecución de una actividad se descompone en la ejecución de varias acciones individuales cada una de las cuales pueden cambiar el estado de un objeto o pueden comunicar mensajes a otros objetos.

### C.1 Elementos de un Diagrama de Actividad

Los elementos básicos de un diagrama de actividad son:

- Las *acciones y nodos de actividad*
- *Objetos de valor*
- *Flujos de control y de objetos*

En las siguientes secciones se resumen brevemente las características de estos elementos.

#### C.1.1 Acciones y nodos de actividad

Los sucesos que describe un Diagrama de Actividad ocurren en las acciones y en los nodos de actividad. Cuando el suceso es atómico, no puede descomponerse, es una acción. Mientras que, cuando el suceso puede descomponerse en sucesos más elementales es un nodo de actividad o actividad.

Tanto las acciones como los nodos de actividad se representan mediante un rectángulo con las esquinas redondeadas (véase la Figura C.1) y un texto descriptivo en su interior. En el caso de las acciones el texto se corresponde a una operación simple o a una expresión, mientras que en los nodos de actividad el texto se corresponde al nombre de otra actividad.

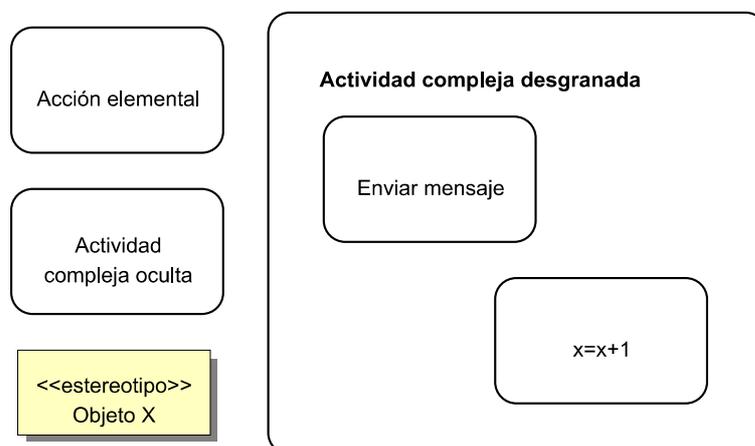
En general, una acción puede verse como un caso particular de una actividad. Por simplicidad, en adelante, se hablará exclusivamente de actividades, aunque todo sería igualmente aplicable a las acciones.

#### C.1.2 Objetos de valor

En los Diagramas de Actividad se pueden representar instancias de los objetos involucrados mediante rectángulos (véase la Figura C.1) En el interior de estos rectángulos se describe mediante

texto la naturaleza del objeto y su valor. Además se puede etiquetar, mediante un texto entrecomillado, que se denomina estereotipo, categorías especialmente importantes de objetos (como por ejemplo los almacenes).

Los objetos de un Diagrama de Actividad pueden corresponder a elementos producidos, consumidos o utilizados por una actividad.



**Figura C.1.-** Nodos de un Diagrama de Actividad. Los rectángulos redondeados corresponden a acciones o a actividades, mientras que los rectángulos normales corresponden a objetos. Obsérvese que se suelen permitir licencias como sombreados o colores.

### C.1.3 Flujos de control y de objetos

En los diagramas de Actividad se distinguen dos tipos de flujos: los flujos de control y los flujos de objetos.

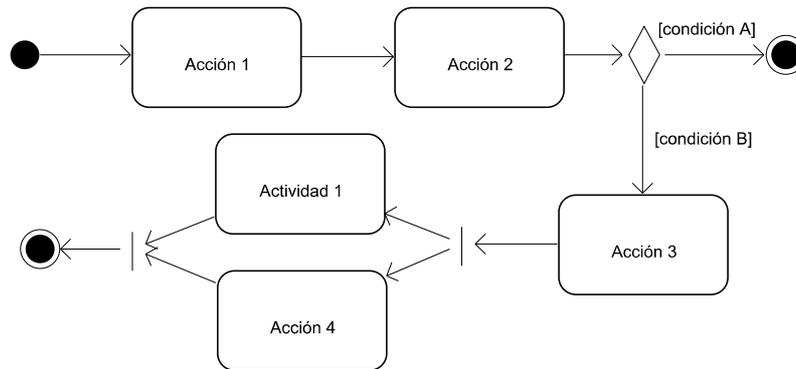
#### Flujos de control

Los flujos de control dentro de un Diagrama de Actividad marcan el orden en el que se pasa de una actividad a otra. Los flujos de control se detallan mediante flechas que unen las cajas de las actividades (véase la Figura C.2).

Es posible especificar el principio de un flujo de control mediante un círculo relleno. Igualmente, el fin de un flujo de control se puede especificar mediante un círculo relleno concéntrico a una circunferencia exterior.

Dentro de los flujos de control es posible especificar bifurcaciones mediante rombos huecos. Al rombo llega una flecha de flujo de control y salen tantas flechas como bifurcaciones se produzcan. Acompañando a cada flecha de salida del rombo se escribe un texto entre corchetes, que se denomina *guarda*, con la condición que determina que se siga ese camino.

También es posible especificar flujos concurrentes en los Diagramas de Actividades. Los puntos en los que el control se divide o se une en múltiples hilos se denominan *barras de sincronización* y se representan mediante un segmento horizontal o vertical de trazo grueso.



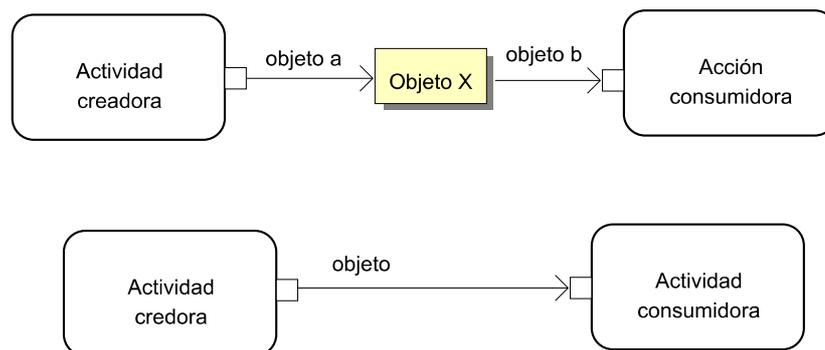
**Figura C.2.-** Ejemplo de flujo de control. El punto relleno muestra el punto de inicio de flujo de control, los puntos concéntricos son puntos de fin del flujo, el rombo es una bifurcación y las barras verticales son puntos de sincronización donde se crean o se unifican hilos separados de ejecución.

### Flujos de objetos

Cuando en un Diagrama de Actividad se involucran objetos es preciso detallar qué actividad los produce o consume. Así, cuando un objeto recibe una flecha de una actividad, significa que el objeto es producido por ella. Por otro lado, cuando existe una flecha de un objeto a una actividad significa que la actividad consume dicho objeto.

A partir de la versión 2.1 de UML, es posible añadir una pequeña caja, denominada *pin*, al contorno de una actividad para indicar de manera abreviada que esa actividad crea o consume un objeto. En este caso, sobre la flecha se escribe un texto que describe al objeto involucrado (véase la Figura C.3).

Finalmente, es posible enlazar dos objetos mediante una flecha para indicar que los objetos interactúan entre sí.



**Figura C.3.-** Ejemplo de flujo de objetos. Arriba la “actividad creadora” crea un objeto *a*, se lo envía al *objeto X* y éste a su vez genera un objeto *b* que lo consume la “actividad consumidora”. Abajo la “actividad creadora” crea un objeto que lo envía a una “actividad consumidora”.



## Anexo D Bibliografía

- [1] R. Plamondon, G. Lorette, "Automatic signature verification and writer detection: the state of the art", *Pattern Recognition*, núm. 22, vol. 2, págs. 107-131, 1989.
- [2] M. A. Ismail, Samia Gad, "Off-line arabic signature recognition and verification", *Pattern Recognition*, págs. 1727-1740, 2000.
- [3] R. M. Bolle, J. H. Connell, S. Pankanti, N. K. Ratha, A. W. Senior, "Guide to Biometrics", Springer, 2003.
- [4] J. Fierrez, J. Ortega, J. González, "Reconocimiento de firma escrita", *Tecnologías biométricas aplicadas a la seguridad*, págs. 201-222, 2004.
- [5] K. Han y I. K. Sethi, "Handwritten signature retrieval and identification", *Pattern Recognition Letters*, vol. 17, págs. 83-90, 1996.
- [6] C. Y. Suen, Q. Xu, L. Lam, "Automatic recognition of handwritten data on cheques - Fact or fiction?", *Pattern Recognition Letters*, vol. 20, págs. 1287-1295, 1999.
- [7] N. Gorski, V. Anisimov, E. Augustin, O. Baret, S. Maximov, "Industrial bank check processing: the A2iA CheckReader", *International Journal on Document Analysis and Recognition*, vol. 3, págs. 196-206, 2001.
- [8] S. Lee y J.C. Pan, "Offline tracing and representation of signatures", *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, págs. 679-680, 1991.
- [9] M. E. Munich, P. Perona, "Visual Signature Verification using Affine Arc-length", *Proceedings of the IEEE Computer Society, Conference on Computer Vision and Pattern Recognition*, págs. 180-186, 1999.
- [10] J. I. Sanz, A. Ruiz, L. Gavete, C. Manzano, J. Herranz, "Reconocimiento y verificación on-line de firmas", *VIII Encuentros de Geometría Computacional : Castelló de la Plana*, págs. 249-258, 1999.
- [11] H. Weiping, Y Xiufen, W. Kejun, "A Survey of Off-Line Signature Verification", *Proceedings of the 2004 International Conference on Intelligence Mechatronics and Automation*, págs. 536-541, 2004.
- [12] R. Plamondon, S. N. Srihari, "On-line and off-line handwriting recognition: a comprehensive survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, núm. 1, vol. 22, págs. 63-84, 2000.
- [13] D. Maravall, "Reconocimiento de formas y visión artificial", RA-MA Editorial, 1993.
- [14] J. D. Hobby, "Using Shape and layout information to find signatures, text, and graphics", *Computer Vision and Image Understanding*, vol. 80, págs. 88-110, 2000.
- [15] I. Pavlidis, N. Papanikolopoulos, "Application of deformable structures to signature identification", *Proceedings of the International Conference on Digital Signal Processing*, vol. 2, págs. 26-28, 1995.

- [16] I. Pavlidis, R. Mavuduru, N. Papanikolopoulos, "Off-line recognition of signatures using revolving active deformable models", IEEE International Conference on Systems, Man, and Cybernetics, Humans, Information and Technology, págs. 771-776, 1994.
- [17] S. Impedovo, G. Dimauro, G. Pirlo, "Algorithms for Automatic Signature Verification", Handbook of Character Recognition and Document Image Analysis, vol. 23, págs. 605-621, 1997.
- [18] M. Ammar, Y. Yoshida, T. Fukumura, "Off-line preprocessing and verification of signatures", International Journal of Pattern Recognition and Artificial Intelligence, núm. 4, vol. 2, págs. 589-602, 1987.
- [19] L. L. Lee, M. G. Lizarraga, N. R. Gomes, A. L. Koerich, "A prototype for brazilian bankcheck recognition", International Journal of Pattern Recognition and Artificial Intelligence, núm. 4, vol. 11, págs. 549-569, 1997.
- [20] B. Fang, C.H. Leung, Y. Y. ang, P. C. K. Kowok, K. W. Wong, "Offline signature verification with generated training samples", IEE Proceedings on Vision, Image and Signal Processing, núm. 2, vol. 149, págs. 85-90, 2002.
- [21] XG. You, B. Fang, ZY. He, YY. Tang, "Similarity measurement for off-line signature verification", Advances In Intelligent Computing, PT 1, Proceedings Lecture Notes In Computer Science 3644, págs. 272-281, 2005.
- [22] Y. Mizukami, M. Yoshimura, H. Miike I. Yoshimura, "An off-line signature verification system using an extracted displacement function", Pattern Recognition Letters, vol. 23, págs. 1569-1577, 2002.
- [23] V. E. Ramesh, M. Narasimha Murty, "Off-line signature verification using genetically optimized weighted features", Pattern Recognition, vol. 32, págs. 217-233, 1999.
- [24] J. L. Esteban, "Procesamiento automático de Cheques", Informe Técnico, págs. 1, 1999.
- [25] R.C. Gonzalez, R.E. Woods, "Digital Image Processing", Addison-Wesley, págs. Cap. Morfología, 2007.
- [26] M. Kass, A. Witkin, D. Terzopoulos, "Snakes: Active contour models", International Journal of Computer Vision, págs. 321-331, 1988.
- [27] J.D. Villalain, L. M. Muñoz, V. Moya, "Presente de la Peritación: el Peritaje Caligráfico", Revista Española de Medicina Legal, núm. 10-11, págs. 58-66, 1977.
- [28] J. Picchia, "Tratado de Documentoscopia. La falsedad documental", La Roca, Buenos Aires, págs. 36, 1993.
- [29] F. Mendez, "Documentoscopia", Estudios de Policía Científica, División de Formación y Perfeccionamiento de la D.G.P., Ministerio del Interior, págs. 14, 1994.
- [30] K. Franke, M. Koppen, "A Computer-based System to Support Forensic Studies on Handwritten Documents", International Journal on Document Analysis and Recognition, Springer Berlin / Heidelberg, núm. 4, vol. 3, págs. 218-231, 2001.
- [31] F. de Antón y Barbera, F. Méndez, "Análisis de textos manuscritos, firmas y alteraciones documentales", Ciencia Policial, Tirant lo Branch, 2005.
- [32] F. Álvarez, "Identificación de firmas y grafometría", Asociación Profesional de Peritos Calígrafos de Cataluña, 1993.
- [33] Scheneickert y Meyer, "Zur Frage der Graphometria", , 1915.
- [34] E. Locard, "Traité de criminalistique", Lyon, Payoy, 1936.

- [35] C. Champod, "Edmond Locard - Numerical Standards & "Probable" Identifications", *Journal of Forensic Identification*, núm. 2, vol. 45, págs. 136-163, 1995.
- [36] M. Ras, "Lo que sabemos de Grafopatología", Gregorio del Toro Editor, 1968.
- [37] A. Posada, "Grafología y Grafopatología", Ed. Paraninfo, Madrid, 1977.
- [38] N. Tallent, "On Individualizing the Psychologists's Clinical Evaluation", *Journal of Clinical Psychology*, vol. 14, págs. 243-244, 1958.
- [39] B. Beyerstein, F. Dale, "The Write Stuff: Evaluation of Graphology - The Study of handwriting Analysis", Prometheus Books, 1992.
- [40] E. J. R. Justino, A. EL Yacoubi, F. Bortolozzi, R. Sabourin, "An Off-Line Signature Verification System Using Hidden Markov Model and Cross-Validation", *Proceedings of the ICDAR 99*, vol. 2, págs. 859-869, 2000.
- [41] F. Leclerc, R. Plamondon, "Automatic signature verification the state of the art 1989-1993", *International Journal of Pattern Recognition and Artificial Intelligence*, núm. 3, vol. 8, págs. 643-660, 1994.
- [42] J. R. Parker, "Algorithms for image processing and computer vision", Wiley, 1997.
- [43] L. Bottou, P. Haner, P. G. Howard, "High Quality Document Image Compression with DjVu", *Journal of Electronic Imaging*, núm. 3, vol. 7, págs. 410-425, 1998.
- [44] J. L. Esteban, "Captura Masiva de Datos de Documentos", *Aplicaciones de la Visión Artificial y la Biometría Informática*, págs. 79-90, 2005.
- [45] R. N. Nagel, "Computer screening of handwritten signatures: a proposal", *Computer Science Centre. University of Maryland, College Park*, págs. TR-220, 1973.
- [46] R.N. Nagel, A. Rosenfeld, "Steps towards handwritten signature verification", *Proceedings of 1st International Joint Conference on Pattern Recognition*, págs. 59-65, 1973.
- [47] M. Ammar, "Identification of fraudulent Japanese signatures from actual handwritten documents: a case study", *Proceedings of International Workshop on Frontiers in Handwriting*, págs. 369-374, 1991.
- [48] H. Lv, WY. Wang, C. Wang, Q. Zhuo, "Off-line Chinese signature verification based on support vector machines", *Pattern Recognition Letters*, núm. 15, vol. 26, págs. 2390-2399, 2005.
- [49] I. Yoshimura, M. Yoshimura, "Off-line verification of Japanese signature after elimination of background patterns", *International Journal of Pattern Recognition and Artificial Intelligence*, núm. 3, vol. 8, págs. 693-708, 1994.
- [50] F. Leclerc, R. Plamondon, "Automatic signature verification the state of the art 1989-1993", *International Journal of Pattern Recognition and Artificial Intelligence*, núm. 3, vol. 8, págs. 643-660, 1994.
- [51] G. Pirlo, "Algorithms for Signature Verification", *Fundamentals in Handwriting Recognition*, Springer, págs. 435-454, 1994.
- [52] R. Sabourin, "Off-line signature verification: recent advances and perspective", *Brazilian Symposium on Document Image Analysis (BSDIA)*, págs. 84-98,, 1997.
- [53] S. Impedovo, G. Dimauro, M. G. Lucchese, R. Modugno, G. Pirlo, "Recent Advancements in Automatic Signature Verification", *9th International Workshop on Frontiers in Handwriting Recognition*, págs. 179-184, 2004.

- [54] R.C. Gonzalez, R.E. Woods, "Digital Image Processing 3<sup>a</sup> Edition", Addison-Wesley, págs. Cap. Morfología, 2007.
- [55] M. Ammar, Y. Yoshida, T. Fukumura, "Structural description and classification of signature images", Pattern Recognition, núm. 7, vol. 23, págs. 697-710, 1990.
- [56] R. Bajaj, S. Chaudhury, "Signature verification using multiple neural classifiers", Pattern Recognition, núm. 1, vol. 30, págs. 1-7, 1997.
- [57] D. A. Mighell, T. Wilkinson, J. Goodman, "Backpropagation and its application to handwritten signature verification", Advances in Neural Information Processing Systems, págs. 340-347, 1989.
- [58] R. Sabourin, J.P. Drouhard, E.S.Wah, "Shape Matrices as a Mixed Shape Factor for Off-Line Signature Verification", International Conference on Document Analysis and Recognition (ICDAR), págs. 661-664, 1997.
- [59] C. F. Lam, D. Kamins, "Signature recognition throug spectral analisys", Pattern Recognition, núm. 1, vol. 22, págs. 39-44, 1989.
- [60] G. Dimauro, S. Impedovo, G. Pirlo, A. Salzo, "A multi-expert sign. verification system for bankcheck processing", Series in Machine Perception and Artificial Intelligence, vol. 28, págs. 365-382, 1997.
- [61] Canan Senol y Tülay Yildirim, "Signature Verification Using Conic Section Function Neural Network", ISCIS 2005 - Springer-Verlag Berlin Heidelberg, págs. 524-532, 2005.
- [62] M. Ammar, T. Fukumura, "A new effective approach for off-line verification of signatures by using pressure features", Proceedings of the 8th International Conference of Pattern Recognition, págs. 566-569, 1989.
- [63] H. Srinivasan, S. N. Srihari, M. Beal, "Signature verification using Kolmogorov-Smirnov statistic", Proceedings International Graphonomics Society Conference (IGS), págs. 152-156, 2005.
- [64] N. A. Murshed, R. Sabourin, F. Bortolozzi, "A cognitive aproach to off-line signature verification", International Journal of Pattern Recognition and Artificial Intelligence, vol. 11, págs. 801-825, 1997.
- [65] H. Zhiqiang, H. Chongzhao, "Force fiel analysis snake: an improved parametric active contour model", Pattern Recognition Letters, vol. 26, págs. 513-526, 2005.
- [66] M. Ammar, Y. Yoshida, T. Fukumura, "Feature extraction and selection for simulated signature verification", Computer Recognition and humman production of handwriting, págs. 61-76, 1988.
- [67] R. Sabourin, M. Cheriet, G. Genest, "An extended shadow-code based approach for off-line signature verification: Part-IEvaluation of The Bar Mask Definition", 12th International Conference on Pattern Recognition ICPR, Jerusalem, Israel, vol. 2, págs. 450-453, 1994.
- [68] I. W. McKeague, "A statistical model for Signature Verification", Journal of the American Statistical Association, núm. 469, vol. 100, págs. 231-241, 2005.
- [69] M. Ammar, Y. Yosheda, T. Fukumura, "Off-line preprocesing and verification of signatures", International Journal of Pattern Recognition and Artificial Intelligence, págs. 589-602, 1988.
- [70] H. Cardot, M. Revenu, B. Victorri, M.J. Revillet, "A static signature verification signature system based on a cooperating neural networks architecture", International Journal of Pattern Recognition and Artificial Intelligence, núm. 3, vol. 8, págs. 679-692, 1994.
- [71] N. Papamarkos, H. Baltzakis, "Off-line signature verification using multiple neural network classification structures", Proc. IEEE International Conference on Digital Signal Processing (DSP),

págs. 727-730, 1997.

[72] E. Özgündüz, T. Sentürk, M. E. Karşlıgil, "Efficient off-line Verification And Identification Of Signatures by Multiclass SVM", Computer Analysis of Images and Patterns, 11th International Conference (CAIP), vol. 3691/2005, págs. 799-805, 2005.

[73] R. Sabourin, R. Plamondon, "Segmentation of Handwritten Signature Images Using the Statistic of Directional Data", ICPR, págs. 282-285, 1988.

[74] F. Nouboud, "Handwritten Signature Verification: A Global Approach", Fundamentals in Handwriting Recognition, Springer, págs. 455-459, 1994.

[75] M. Dehghan, K. Faez, M. Fathi, "Signature verification using shape descriptors and multiple neuralnetworks", Proceedings of IEEE, IEEE Region 10 Annual Conference, Speech and Image Technologies for Computing and Telecommunications (TENCON), vol. 1, págs. 415-418, 1997.

[76] B. Fang, C.H. Leung, Y. Y. ang, P. C. K. Kowok, K. W. Wong, "A smoothness index based approach for off-line signature verification", Proceedings of the 5th International Conference on Document Analysis and Recognition ICDAR99, págs. 785-787, 1999.

[77] K. Huang, H. Yan, "Signature verification using fractal transformation", Proceedings of the 15th International Conference on Pattern Recognition (ICPR), vol. 2, págs. 851 - 854, 2000.

[78] X. Xiaoa, G. Leedhamb, "Signature verification using a modified Bayesian network", Pattern Recognition, vol. 35, págs. 983-995, 2002.

[79] J. F. Vélez, A. Sanchez y B. Moreno, "Robust Off-line Signature Verification using Compression Networks and Positional Cuttings", IEEE Workshop on Neural Networks for Signal Processing , NSSP, Toulouse, págs. 627- 636, 2003.

[80] J.J. Igarza, I. Hernaez, I. Goirizelaia, "Off-line signature recognition based on dynamic methods", Proceedings of the SPIE - International Society for Optical Engineering - Biometric Technology for Human Identification II, págs. 336-343, 2005.

[81] R. Plamondon, G Lorette, R. Sabourin, "Automatic Processing of Signature Images: Static Techniques and Methods", Computer Processing Of Handwriting, págs. 49-64, 1990.

[82] J.P. Drouhard, R. Sabourin, M. Godbout, "A NN approach to off-line sign. verifi. using directional PDF", Pattern Recognition, núm. 3, vol. 29, págs. 415-424, 1996.

[83] R. Sabourin, G. Genest, F. J. Preteux, "Off-line signature verification by local granulometric size distributions", IEEE Transactions on Pattern Analysis and Machine Intelligence, núm. 9, vol. 19, págs. 976-988, 1997.

[84] J. K. Guo, D. Doermann, A. Rosenfeld, "Off-Line Skilled Forgery Detection Using Stroke and Sub-Stroke Properties", Proceedings of the 15th International Conference on Pattern Recognition (ICPR), vol. 2, págs. 2355-2358, 2000.

[85] C. Quek, R.W. Zhou, "Antiforgery: a novel pseudo-outer product based fuzzy NN driven signature verification system", Pattern Recognition Letters, vol. 23, págs. 1795-1816, 2002.

[86] V. Krishna Madasu, M. Hafizuddin, M. Yusof, M. Hanmandlu y K. Kubik, "Off-Line Signature Verification and Forgery Detection System Based on Fuzzy Modeling", Artificial Intelligence - Springer-Verlag Berlin Heidelberg, págs. 1003-1013, 2003.

[87] M. K. Kalera, S. Srihari, A. Xu, "Off-line signature verification and identification using distance statistics", International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), núm. 7, vol. 18, págs. 1339-1360, 2004.

- [88] J.J. Igarza, I. Hernaez, I. Goirizelaia, "Static signature recognition based on left-to-right Hidden Markov Models", *Journal of Electronic Imaging*, núm. 4, vol. 14, págs. 043015, 2005.
- [89] R. Plamondon, R. Sabourin, "Preprocessing of handwritten sign. from image gradient analysis", *International Conference on Pattern Recognition*, págs. 576-579, 1986.
- [90] R. Sabourin, R. Plamondon, G. Lorette, "Automatic processing of signatures: Static techniques and methods", *Computer Processing of Handwriting*, págs. 49-63, 1990.
- [91] G. Congedo, G. Dimauro, S. Impedovo, G. Pirlo, "Off-Line Signature Verification by Fundamental Components Analysis", *Progress in Image Analysis and Processing III, Proceedings of the 7th International Conference on Image Analysis and Processing*, vol. 20-22, págs. 331-337, 1993.
- [92] Q. Yingyong, B. Hunt, "Signature verification using global and grid features", *Pattern Recognition*, núm. 12, vol. 27, págs. 1621-1629, 1994.
- [93] Q. Yingyong, B. R. Hunt, "A Multiresolution Approach to Computer Verification of Handwritten Signatures", *IEEE Transactions on Image Processing*, núm. 6, vol. 4, págs. 870-874, 1995.
- [94] K. Huang, H. Yan, "Off-line signature verification based on geometric feature extraction and neural network classification", *Pattern Recognition*, núm. 1, vol. 30, págs. 9-17, 1997.
- [95] P. S. Deng, H.Y.M. Liao, C. W. Ho, H.R Tyan, "Wavelet based off line handwritten signature verification", *Computer Vision and Image Understanding*, núm. 3, vol. 76, págs. 173-190, 1999.
- [96] G. Cavalcanti, R. Doria, E. Carvalho, "Feature Selection For Off-Line Recognition of Different Size Signatures", *IEEE International Workshop on Neural Networks for Signal Processing*, págs. 355-364, 2002.
- [97] A. Zimme, Lee Luan Ling, "A Hybrid On-Off Line Handwritten Signature Verification System", *7th International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1, págs. 424- 428, 2003.
- [98] R. N. Nagel, A. Rosendfeld, "Computer detection of freehand forgereis", *IEEE Transaction Computer*, vol. 26, págs. 895-905, 1977.
- [99] M. J. Revillet, "Signature Verification on Postal Cheques", *Proceedings of International Conference on Document Analysis and Recognition (ICDAR)*, págs. 767-773, 1991.
- [100] R. Sabourin, R. Plamondon, G. Lorette, "Offline identification with handwritten signature images: survey and perspective", *Structured Image Analysis*, págs. 219-234, 1992.
- [101] N. G. See, O. H. Seng, "A neural network approach for off-line signature verification", *Proceedings of IEEE International Conference on Computer, Communication, Control and Power Engineering*, págs. 770-773, 1993.
- [102] R. Sabourin, R. Plamondon, L. Beaumier, "Structural Interpretation of Handwritten Signature Images", *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, núm. 3, vol. 8, págs. 709-748, 1994.
- [103] R. Sabourin, G. Genest, "An extended-shadow-code based approach for off-line signature verification. II. Evaluation of several multi-classifier combination strategies", *Proceedings of the Third International Conference on Document Analysis and Recognition ICDAR*, vol. 1, págs. 197, 1995.
- [104] R.W. Zhou, C. Quack, "An Automatic fuzzy neural network driven signature verification system", *IEEE International Conference on Neural Networks*, págs. 1034-1039, 1996.

- [105] E. A. Fadhel, P. Bhattacharyya, "Application of a steerable wavelet transform using NN for signature verification", *Pattern Analysis & Applications*, págs. 185-195, 1999.
- [106] H. Baltzakisa, N. Papamarkos, "A new signature verification technique based on a two-stage Neural Network classifier", *Engineering Applications of Artificial Intelligence*, vol. 14, págs. 95-103, 2001.
- [107] Kai Huang, Hong Yan, "Off-line signature verification using structural feature correspondence", *Pattern Recognition*, vol. 35, págs. 2467-2477, 2002.
- [108] B. Kovari, Z. Kertesz y A. Major, "Off-Line Signature Verification Based on Feature Matching", *Intelligent Engineering Systems*, 2007. INES 2007. 11th International Conference on, págs. 93-97,, 2007.
- [109] W.F. Nemcek, W.C. Lin, "Experimental Investigation of Automatic Signature Verification", *IEEE Transactions Systems, Man and Cybernetics*, págs. 121-126, 1974.
- [110] E.R. Brocklehurst, "Computer Methods of Signature Verification", *Journal Forensic Science Society*, págs. 445-457, 1985.
- [111] T. Wilkinson, J. Goodman, "Slope histogram detection of forged handwritten signatures", *Proceedings of the SPIE - International Society for Optical Engineering*, págs. 293-304, 1990.
- [112] G. Krishnan, D. Jones, "Machine verification of traced signatures", *Proceedings of the SPIE - International Society for Optical Engineering*, págs. 563-572, 1991.
- [113] M. Ammar, "Elimination of skilled forgeries in off-line systems: a breakthrough", *Proceedings of IEEE Conference of Pattern Recognition and Artificial Intelligence*, págs. 415-418, 1992.
- [114] V. A. Shapiro, I. S. Bakalov, "Static signature verification as a dynamic programming problem", *Proceedings in the 6th International Conference Handwriting Drawing ICOHD*, págs. 219-221, 1993.
- [115] R. Plamondon, "Special Issue on Automatic Signature Verification", *International Journal of Pattern Recognition and Artificial Intelligence*, núm. 3, vol. 8, págs. 641-811, 1994.
- [116] R. Sabourin, G. Genest, F. Preteux, "Pattern Spectrum as a Local Shape Factor for Off-Line Signature Verification", *Proceedings of International Conference on Pattern Recognition (ICPR)*, vol. 3, págs. 43-48, 1996.
- [117] De Oliveira, C. Kaestner, Bortolozzi, R. Sabourini, "Generation of signatures by deformations", *Advances in Document Image Analysis, First Brazilian Symposium, BSDIA'97*, págs. 283-298, 1997.
- [118] B. M. Herbst, H. Coetzer, "On an Off-line Signature Verification System", *Proceedings of the 9th Annual South African Workshop on Pattern Recognition*, págs. 39-43, 1998.
- [119] J.N. de Gouvea Ribeiro, G.C. Vasconcelos, "Off-line signature verification using an auto-associatorcascade-correlation architecture", *International Joint Conference on Neural Networks (IJCNN)*, vol. 4, págs. 2882-2886, 1999.
- [120] B. Fang, Y. Y. Wang, C. H. Leung, K. W. Tse, Y. Y. Tang, P. C. K. Kwok, Y., "Off-line signature verification by the analysis of cursive strokes", *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, núm. 4, vol. 15, págs. 659-673, 2001.
- [121] M. Hanmandlu, V.K. Madasu, S. Madasu, "Neuro-Fuzzy Approaches to Signature Verification", *2nd National Conference on Document Analysis and Recognition (NCDAR-2003)*, págs. 164-171, 2003.

- [122] J. Coetzer, B. M. Herbst, J. A. du Preez, "Offline Signature Verification Using the Discrete Radon Transform and a Hidden Markov Model", *EURASIP Journal on Applied Signal Processing*, vol. 4, págs. 559-571, 2003.
- [123] J. F. Vélez, A. Sanchez, B. Moreno, J. L. Esteban, "Combining Snakes and Neural Networks for Off-Line Signature Verification", *Proceedings of the Tenth Internacional Workshop on Frontiers in Handwriting Recognition*, págs. 499-504, 2006.
- [124] H. Coetzer y R. Sabourin, "A Human-Centric Off-Line Signature Verification System", *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Vol. 1., 2007.
- [125] F. Lamarche, R. Plamondon, "Segmentation and feature extracton of handwritten signature patterns", *Procedings 7th International Conference on Pattern Recognition*, págs. 756-759, 1984.
- [126] M. Ammar, "Progress in verification of skillfully simulated handwritten signatures", *International Journal of Pattern Recognition and Artificial Intelligence*, núm. 1&2, vol. 5, págs. 337-351, 1991.
- [127] S. Lee, J.C. Pan, "Offline tracing and representation of signatures", *IEEE Transactions on Systems, Man. and Cybernetics*, núm. 4, vol. 22, págs. 755-771, 1992.
- [128] H. Cardot , M. Revenu , B. Victorri, M-J. Revillet, "An artificial neural networks architecture for handwritten signature authentication", *SPIE Intelligent Information Systems*, págs. 633-644, 1993.
- [129] J. K. Guo, D. Doerman, A. Rosenfeld, "Local correspondence for detecting random forgeries", *Proceedings of the 4th International Conference on Document Analysis and Recognition (ICDAR)*, págs. 319 - 323, 1997.
- [130] Y.Mizukami,M.Yoshimura, H.Miike, I. Yoshimura, "An off-line signature verification system using an extracted displacement function", *Proceedings of the Fifth International Conference on Document Analysis and Recognition*, págs. 757-760, 1999.
- [131] E. Justino, F. Bortolozzi, R. Sabourin, "Off-line signature verification using HMM for random, simple and skilled forgeries", *Proceedings of the Sixth International Conferenceon Document Analysis and Recognition*, págs. 1031-1034, 2001.
- [132] D. Chen, X. Zhang, K. P. Chan, "Adaptative weighting algorithm on off-line signature verification", *Proceedings of the 3rd IASTED International Conference, Visualization, Imaging, and image processing*, págs. 710-715, 2003.
- [133] Qiao, Yu Liu, Jianzhuang Tang, Xiaoou, "Offline Signature Verification Using Online Handwriting Registration", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR '07)*, págs. 1-8, 2007.
- [134] R. I. Phelps, "A holistic approach to signature verification", *Proceedings of the Sixth International Conference of Pattern Recognition*, págs. 1187, 1982.
- [135] R. Sabourin, M. Cheriet, G. Genest, "An extended shadow-code based approach for off-line signature verification", *2nd Conference on Document Analysis and Recognition IAPR*, Ibraki, Japan, págs. 1-5, 1993.
- [136] S. Djeziri, F. Nouboud, R. Plamondon, "Extraction of signature from check background based on a filiformity criterion", *IEEE Transactions on Image Processing*, núm. 10, vol. 7, págs. 1425-1438, 1998.
- [137] J. L. Camino, C. M. Travieso, C. R. Morales, M. A. Ferrer, "Signature Classification by Hidden

- Markov Model", Proceedings of the 33rd Annual 199 International Carnahan Conference, págs. 481-484, 1999.
- [138] B. Fang, C.H. Leung, Y. Y. ang, P. C. K. Kowok, K. W. Wong, "Off-line signature verification by the tracking of feature and stroke positions", Pattern Recognition, vol. 36, págs. 91-101, 2003.
- [139] M. Hanmandlu, M. Hafizuddin M.Yusofb,V. K. Madasu, "Off-line signature verification and forgery detection using fuzzy modling", Pattern Recognition, vol. 38, págs. 341-356, 2005.
- [140] E. Frías-Martínez, A. Sánchez y J. F. Vélez, "Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition", Engineering Applications of Artificial Intelligence, núm. 6, vol. 19, págs. 693-704, 2006.
- [141] A. Piyush Shanker, A. N. Rajagopalan, "Off-line signature verification using DTW", Pattern Recognition Letters, vol. 28, págs. 1407-1414, 2007.
- [142] Y. Xuhua, T. Furuhashi, K. Obata, Y. Uchikawa, "Study on signature verification using a new approach to genetic based machine learning", IEEE International Conference on Systems, Man and Cybernetics, Intelligent Systems for the 21st Century, vol. 5, págs. 4383 - 4386, 1995.
- [143] L. L. Lee, M. G. Lizárraga, "An off-line method for human signature verification", Proceedings of the International Conference of Pattern Recognition, págs. 195-198, 1996.
- [144] R. Sabouring,R. Plamondon, G. Lorette, "Offline identification with handwritten signature images: survey and perspective", Structured Image Analysis, págs. 219-234, 1992.
- [145] K. Ueda, "Investigation of Off-Line Japanese Signature Verification Using a Pattern Matching", 7th International Conference on Document Analysis and Recognition (ICDAR), vol. 1, págs. 951-955, 2003.
- [146] J. Fasquel, C. Stolz, M. Bruynooghe, "Real-time verification of handwritten signatures using a hybrid opto-electronical method", Proc. 2nd ISPA IEEE Conference, págs. 552-557, 2001.
- [147] R. Plamondon, G. Lorette, "Desinging an automatic signature verifier: problem definition and system description", Computer Processing of Handwriting, págs. 3-20, 1990.
- [148] B. Fang, Y.Y. Tang, "Improved class statistics estimation for sparse data problems in offline signature verification", IEEE Transactions on Systems, Man and Cybernetics, Part C, núm. 3, vol. 35, págs. 276-286, 2005.
- [149] I. K. Sethi, K. Han, "Use of local structural association for retrieval and recognition of signature images", Proceodings of the SPIE - International Spciety for Optical Engineering, vol. 2420, págs. 125-136, 1995.
- [150] N. A. Murshed, F. Bortolozzi, R. Sabourin, "Off-Line Signature Verification, Without a Priori Knowledge of Class w: A New Approach", Proceedings of International Conference on Document Analysis and Recognition (ICDAR), vol. 1, págs. 191-196, 1995.
- [151] G. Rigoll, A. Kosmala, "A Systematic Comparison Between On-Line and Off-Line Methods for Signature Verification with Hidden Markov Models", International Conference on Pattern Recognition (ICPR), págs. 1755-1757, 1998.
- [152] M. Ammar, Y. Yoshida, T. Fukumura, "A New Effective Approach for Off-line Verification of Signatures by Using Pressure Features", Procedings of 8th International Conference on Pattern Recognition, págs. 566-569, 1986.
- [153] R. Sabourin, R. Plamondon, "Progress in the field of automatic handwritten signature verification systems using grey level images", Procedings of International Workshop on Frontiers in Handwriting, págs. 1-12, 1990.

- [154] M. Ammar, "Performance of parametric and reference pattern based feature in static signature verification: a comparative study", Proceedings of the International Conference of Pattern Recognition, págs. 646-648, 1990.
- [155] Young Woon Woo, Soowhan Han y Kyung Shik Jang, "Off-Line Signature Verification Based on Directional Gradient Spectrum and a Fuzzy Classifier", PSIVT 2006 - Springer-Verlag Berlin Heidelberg, págs. 1018-1029, 2006.
- [156] M.A. Ferrer, J.B. Alonso, C.M. Travieso., "Offline geometric parameters for automatic signature verification using fixed-point arithmetic", IEEE Transactions on Pattern Analysis and Machine Intelligence, núm. 6, vol. 27, págs. 993-997, 2005.
- [157] E. J. R. Justino, F. Bortolozzi, R. Sabourin, "The Interpersonal and Intrapersonal Variability Influences on OffLine Signature Verification Using HMM", roc. XV. Brazilian Symposium on Computer Graphics and ImageProcessing (SIBGRAPI), págs. 197-201, 2002.
- [158] M. A. Ferrer, J. B. Alonso y C. M. Travieso, "Offline Geometric Parameters for Automatic Signature Verification Using Fixed-Point Arithmetic", IEEE Transactions on Pattern Analysis And Machine Intelligence, núm. 6,págs. 993-997,, 2005.
- [159] R. Plamondon , G. Lorette, G. Poulin, "On the intrinsictheoretical and practical difficulties of designing an automaticsignature verification system", Int. Symp. OnHandwriting and Computer Applic., Montreal, págs. 154-156, 1987.
- [160] B. Fang, Y.Y. Tang, "Reduction of Feature Statistics Estimation Error For Small Training Sample Size In Off-Line Signature Verification", Intenational Conference on Biometric Authentication ICBA04, págs. 526-532, 2004.
- [161] Y. QI, B. Hunt, "Signature verification using global and grid features", Pattern Recognition, núm. 12, vol. 27, págs. 1621-1629, 1994.
- [162] S. Barua, "Neural Networks and their application to computer security", Proc. SPIE, págs. 735-742, 1992.
- [163] E. Justino, Yacoubi, F. Bortolozzi, R. Sabourin, "An Off-Line sign. verif. system using HMM and graphometric features", 4th IAPR International Workshop on Document Analysis Systems, págs. 211-222, 2000.
- [164] E. Justino, F. Bortolozzi, R. Sabourin, "A comparison of SVM and HMM classifiers in the off-line signature verification", Pattern Recognition, vol. 26, págs. 1377-1385, 2005.
- [165] S. Larrea, M. A. Ferrer, C. T. González, F. Vargas, J. B. Alonso, "Eliminación del fondo de un cheque bancario", Terceras Jornadas de Reconocimiento Biométrico de Personas, 2006.
- [166] X. YE, M. Cheriet y C. Y. Suen, "A Generic System To Extract And Clean Handwrittendata From Business Forms", Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, págs. 63-72, 2000.
- [167] K. Ueda, H. Maegawa y K. Matsuo, "Automatic Extraction of Filled-In Items from Bank-Check Images", Lecture Notes in Computer Science: Document Analysis Systems VI, vol. 3163/2004, págs. 225-228, 2004.
- [168] V. K. Madasu y B. C. Lovell, "Automatic Extraction of Signaturas From Bank Cheques and Other Documents", Digital Image Computing: Techniques and Applications (DICTA 2003), págs. 591-600, 2005.
- [169] Jin-Yong Yoo, Min-Ki Kim, Sang Yong Han y Young-Bin Kwon, "Line Removal and Restoration of Handwritten Characters on the Form Documents", Proceedings of the 4th

- International Conference on Document Analysis and Recognition, págs. 128-131, 1997.
- [170] S. H. Kim, S. H. Jeong, H. K. Kwag, "Line removal and character restoration using bag representation of form images", Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, págs. 43-52, 2000.
- [171] S. Cha y S. N. Srihari, "Convex Hull Discriminant Function and Its Application to Writer Identification Problem", Proceedings 5th Joint Conference on Information Sciences (JCIS2000), Atlantic City, NJ, págs. 13-16,, 2000.
- [172] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns", Communications of the ACM, vol. 27, pags. 236-239, 1984.
- [173] Rasha Abbas, "Backproagation Networks Prototype for Off-line Signature Verification", Department of Computer Science, RMIT, 1994.
- [174] D. Valentin, H. Abdi, A.J. O'Toole, G.W. Cottrell, "Connectionist Models of Face Processing: A Survey", Pattern Recognition, núm. 9, vol. 27, págs. 1209-1230, 1994.
- [175] T. Kohonen, "Associative Memory: A system Theoretic Approach", Springer, 1977.
- [176] P. Baldi y K. Hornik, "Neural Networks and Principal Components Analysis: Learning From Examples Without Local Minima", Neural Networks, núm. 1, vol. 2, págs. 53-58, 1989.
- [177] C. G. Looney, "Neural Networks for Patterns Recognition", Oxford University Press, 1997.
- [178] N.E. Davison, H. Eviatar, R.L. Somorjai, "Snakes simplified", Pattern Recognition, vol. 33, págs. 1651-1664, 2000.
- [179] P. Radeva, J. Serrat, E. Marti, "A snake model based segmentation", Proceedings on the International Conference in Computer Vision (ICCV), págs. 816-821, 1995.
- [180] R. Nilanjan, C. Bhabatosh, D. Jyotirmay, "A fast and flexible multiresolution snake with a definite termination criterion", Pattern Recognition, vol. 34, págs. 1483-1490, 2001.
- [181] D.S. Doermann y A. Rosenfeld, "The Interpretation and Reconstruction of Interfering Strokes", Frontiers in Handwriting Recognition III, págs. 41-50, 1993.
- [182] A. A. Amini, T. E. Weymouth, R. C. Jain, "Using dynamic programming for solving variational problems in vision", IEEE Transactions on pattern analysis and machine intelligence, núm. 9, vol. 12, págs. 855-866, 1990.
- [183] L.D. Cohen, I. Cohen, "Finite-element methods for active contour models and balloons for 2-D and 3-D images", IEEE Trans. PAMI, núm. 11, vol. 15, págs. 1131-1147, 1993.
- [184] F. Howing, D. Wermser, L.S. Dooley, "Sixth Intl. Conf. on Image Processing and its Applications", , vol. 2, págs. 627-630, 1997.
- [185] S. G. Bueno, A. Martínez-Albalá and A. Adán, "Fuzzy-Snake Segmentation of Anatomical Structures Applied to CT Images", Lecture Notes in Computer Science, vol. 3212, págs. 33-42, 2004.
- [186] J. M. Mendel, "Fuzzy Logic Systems for Engineering: A Tutorial", Proceedings of The IEEE núm 3, vol. 83,, 1995.
- [187] K. Yoshida y H. Sakoe, "Online handwritten character recognition for a personal computer system", , 1982.
- [188] R. Basri, L. Costa, D. Geiger y D. Jacobs, "Determining the Similarity of Deformable Shapes", , 1998.
- [189] L. Rabiner, B-H. Juang, "Fundamentals of Speech Recognition", Prentice Hall Signal

Processing Series, Part 1, págs. 321-386, 1993.

[190] A. Perez, A. Sanchez y J.F. Velez, "Simplified Stroke-based Approach for Off-line Signature Recognition", Proc. 2nd COST Workshop on Biometrics on the Internet: Fundamentals, Advances and Applications, págs. 89-94, 2004.

[191] J.F. Vélez, A. Sánchez, A.B. Moreno y J.L. Esteban, "Verificación Off-Line de Firmas Manuscritas: Una Propuesta basada en Snakes Paramétricos", Simposio de Reconocimiento de Formas y Análisis de Imágenes, AERFAI - I CEDI, 2005.

[192] J.F. Vélez, A. Sánchez, A.B. Moreno y J.L. Esteban, "A Hybrid Approach using Snakes and Fuzzy Modelling for Off-Line Signature Verification", The 11th IPMU International Conference, 2006.

[193] J. Vélez, A. Sánchez, A.B. Moreno y J.L. Esteban, "Verificación off-line de firmas manuscritas", Campus Multidisciplinar en Percepción e Inteligencia (CMPI '06), vol. 2, págs. 512-523, 2006.

[194] E. Frías-Martínez, A. Sánchez, J. Vélez, "Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition", Engineering Applications of Artificial Intelligence, núm. 6, vol. 19, págs. 693-704, 2006.

[195] J.F. Vélez, A. Sánchez, A.B. Moreno y J.L. Esteban, "Introducing Fuzziness on Snake Models for Off-Line Signature Verification", Seventh International Conference on Intelligent Systems Design and Applications (ISDA 2007), págs. 843-848, 2007.

[196] J. A. Anderson, J. W. Sil, J. W. Silverstein, S. A. Ritz y R. S. Jones, "Distinctive Features, categorical perception and Probability Learning: Some applications of neural model", Psych. Rev., núm. 84, págs. 413-451, 1977.

[197] R. O. Duda y P. E. Hart, "Pattern Classification and Scene Analysis", Academic Press, 1984.

[198] G.W. Cottrell, P. Munro y D. Zipser, "Learning Internal Representations form Gray-Scale Images: An Example of Extensional Programming", Proc. 9th Annual Conference of the Cognitive Science Society, págs. 462-473, 1987.

[199] G. Booch, J. Jacobson y I. Rumbaugh, "El Lenguaje Unificado de Modelado", Addison Wesley, 2007.

## Anexo E Índice alfabético

### A

Acciones, 175  
Adosamientos, 16  
Aleatorios, 21  
Alteraciones fraudulentas, 18  
Alteraciones no fraudulentas, 17  
Aprendizaje, 162

### B

Banda luminiscente, 19  
Barras de sincronización, 176  
Base-line, 92  
Batch, 171  
Binarización, 27  
Binarización adaptativa, 28  
Binarización global, 28  
Bounding box, 88  
Brisados, 18  
BSL, 92

### C

Calcadores, 21  
Calco, 18  
Captura, 4  
Características de clase, 13  
Células nerviosas, 161  
Coeficiente de Fisher, 93  
Compresores con pérdida, 29  
Condiciones realistas, 7  
Conjunto de aprendizaje, 51  
Conjunto de entrenamiento, 162  
Conjunto de test, 51  
Conjunto de test del entrenamiento, 163  
Conjunto de validación, 163  
Cortes Posicionales, 104

Criterio de suavidad, 113  
Cuantización, 25

### D

Depósitos, 18  
Diagrama de Actividad, 9  
Diferencias interpersonales, 20  
Diferencias intrapersonales, 20  
Digitalización, 4  
Dinámico, 4  
Distribuciones de Tamaño Granulométrico, 41  
Documentoscopia, 14  
DPI, 24

### E

EER, 23  
Energía de rigidez, 125  
Energía externa, 109  
Energía interna, 109  
Enlodamientos, 16  
Entrada total, 163  
Entradas sinápticas, 161  
Equal Error Rate, 23  
Error cuadrático medio, 167  
Error de tipo I, 22  
Error de tipo II, 23  
Esqueletizado, 77  
Estado de activación, 161  
Estático, 4  
Expertos, 21

### F

Factor caligráfico, 13  
False Acceptance Rate, 22

False Rejection Rate, 22  
 Falsificación mediante sistemas mecánicos, 19  
 Falsificación por composición, 19  
 Falsificadores aleatorios, 6, 21  
 Falsificadores calcadores, 21  
 Falsificadores hábiles, 6, 21  
 Falsificadores simples, 21  
 Falsificadores simulados, 21  
 Falso negativo, 6  
 Falso positivo, 6  
 FAR, 22  
 Fludo de objetos, 175  
 Flujos de control, 175  
 Fondos de seguridad, 19  
 FRR, 22  
 Función de activación, 161  
 Función de salida, 161  
 Función de transferencia, 161  
 Función Escalón, 164  
 Función Sigmoide, 164

## G

GDS, 42  
 Generación automática de muestra, 8  
 Generalizar, 162  
 Global Descriptor String, 42  
 Grafología, 19  
 Grafopatología, 19  
 Grafoscopia, 14  
 Grafotecnia, 14  
 Grafoterapia, 19  
 Grid, 41  
 Grosor medio del trazo, 77  
 GSD, 41  
 Guarda, 176

## I

Identificación de firmas, 5  
 Imágenes bitonales, 25  
 Imágenes en color real, 25  
 Imágenes en niveles de gris, 25  
 Imágenes RGB, 25  
 Imitación por asimilación, 18  
 Imitación servil, 18  
 Inclinación, 16

Interpersonales, 20  
 Intrapersonales, 20

## L

Lenguaje Unificado de Modelado, 175  
 Límite inferior, 93  
 Límite superior, 93  
 Línea base, 92  
 LL, 93  
 LMS, 167  
 Lower-limit, 93

## M

Mapa de potencial, 122  
 Método anastasiográfico, 15  
 Método escopométrico, 15  
 Método grafológico, 15  
 Método grafonométrico, 15  
 Método grafonómico, 15  
 Método gramatomórfico, 15  
 Método morfológico caligráfico, 15  
 Método segnalético-descriptivo, 15  
 Método sinalético, 15  
 Modelos de Contornos Activos, 8, 157  
 Muestra A, 58  
 Muestra B, 58  
 Muestra C, 58  
 Muestra D, 59  
 Muestra E, 60  
 Muestra F, 61  
 Muestreo, 24

## N

Neuronas, 161  
 Neuronas de entrada, 161  
 Neuronas de salida, 161  
 Neuronas intermedias, 161  
 Neuronas ocultas, 161  
 Nodos de actividad, 175

## O

Objetos de valor, 175

OCON, 83  
Off-line, 4  
On-line, 4, 171

## P

Papeles permeables, 19  
Películas plásticas, 19  
Perceptrón, 164  
Perceptrón Multicapa, 164  
Peso, 161  
Potential map, 122  
Proporciones, 16  
Pseudociencias, 19  
Punto de cruce, 90  
Punto de generalización óptimo, 163  
Puntos de control, 109  
Puntos por pulgada, 24  
Punzón, 18

## R

Rachas de píxeles, 29  
Receiver Operating Characteristic), 23  
Reconocimiento de firmas, 5  
Recuperación de firmas, 5  
Redes Autocodificadoras, 97, 173  
Redes codificadoras, 97, 173  
Redes de Compresión, 97, 173  
Redundancia en la codificación, 28  
Redundancia en la representación, 29  
Redundancia visual, 29  
Regla Delta, 167  
Resolución espacial, 24  
Resolución radiométrica, 24  
Respuesta química, 19  
Retropropagación, 162, 167

ROC, 23  
Ruido blanco, 5  
Ruido con estructura, 5  
Runlengths, 29

## S

Segmentación de firmas, 5  
Simples, 21  
Simulados, 21  
Snake, 109  
Snakes, 8, 157  
Snakes con memoria, 123  
Sobrentrenamiento, 163

## T

Tamaño, 16  
Tasa de aprendizaje, 167  
Tasa de Falsa Aceptación, 22  
Tasa de Falso Rechazo, 22  
Trayectoria, 16

## U

UL, 93  
Umbral, 163  
UML, 175  
UML2, 9  
Universo de trabajo, 51  
Upper-limit, 93

## V

Velocidad, 16  
Verificación de firmas, 5