UNIVERSIDAD POLITECNICA DE MADRID

**ESCUELA TECNICA SUPERIOR DE INGENIEROS
DE TELECOMUNICACION**

TESIS DOCTORAL

**TRATAMIENTO Y REPRESENTACION
DE
ESTRUCTURAS GRAFICAS**

Autor: Guillermo Cisneros Pérez

Director: Narciso García Santos

Mayo 1986

**Tratamiento y Representación
de
Estructuras Gráficas**

**Autor: Guillermo Cisneros Pérez
Director: Narciso García Santos**

**E.T.S.I.Telecomunicación
Universidad Politécnica
28040-Madrid**

Todos los hombres tienen naturalmente el deseo de saber. El placer que nos causan las percepciones de nuestros sentidos son una prueba de esta verdad. Nos agradan por sí mismas, independientemente de su utilidad, sobre todo las de la vista. En efecto, no sólo cuando tenemos intención de obrar, sino hasta cuando ningún objeto práctico nos proponemos, preferimos, por decirlo así, el conocimiento visible a todos los demás conocimientos que nos dan los demás sentidos. Y la razón es que la vista, mejor que los otros sentidos, nos da a conocer los objetos, y nos descubre entre ellos gran número de diferencias.

Aristóteles

A mis padres

Esta tesis ha sido desarrollada en dos partes fundamentales:

- Tratamiento y representación de estructuras gráficas 2-D, llevada a cabo en la Subdirección de Investigación Aplicada del Departamento de Investigación y Desarrollo de Telefónica, fundamentalmente encaminada a la consecución de un sistema de transmisión de firmas y mapas por línea telefónica, aunque ha dado frutos en otros campos muy diversos.
- Tratamiento y representación de estructuras gráficas 3-D, llevada a cabo en el Centro Científico IBM-Madrid, encaminada a la obtención de un sistema de representación de propósito general, apto para aplicaciones que no resuelven los paquetes de software de que actualmente se dispone en el mercado por deficiencias en sus algoritmos.

Madrid, Mayo de 1986.

Resumen

Se presenta un conjunto de algoritmos para tratamiento y representación de estructuras gráficas en 2-D y 3-D. Mientras que la parte de 2-D va encaminada a la consecución de una forma de transformación de imágenes en gráficos y viceversa, la parte de 3-D se orienta hacia la concepción de un sistema completo de representación.

La transformación de imágenes en gráficos se ha efectuado mediante una operación de vectorización. La vectorización lleva algunos pasos previos, tales como el realce de imagen, la segmentación, la extracción opcional de bordes y el adelgazamiento, y pasos posteriores, tales como el suavizado. Se han revisado todos los experimentos sobre estas técnicas, y se han diseñado algoritmos nuevos que superan los fallos que en se encuentran en los antiguos. El núcleo ha sido la creación de un nuevo algoritmo de vectorización muy eficiente, que es capaz de reducir las situaciones particulares al caso general y no se le ha observado ningún fallo. Todas las operaciones realizadas en los algoritmos propuestos son del tipo lógico, lo que acelera el proceso y lo hace realizable en hardware. El suavizado se realiza empleando coordenadas paramétricas y operaciones vectoriales, lo que evita problemas mal condicionados y aumenta su eficiencia. Así, esta forma de transformación de imágenes en gráficos es muy superior en calidad a todas las demás.

El sistema de representación 3-D suple las deficiencias de los algoritmos publicados hasta la fecha. El núcleo es el diseño de un algoritmo de líneas ocultas que resuelve todas las penetraciones y autopenetraciones de las estructuras multiformes que aparecen en el dibujo, generando incluso las intersecciones entre superficies. El algoritmo trabaja perfectamente en los vértices agudos, lo que le hace ampliamente superior a todos los existentes hasta la fecha. Todas las operaciones se realizan vectorialmente, eliminando el uso de determinantes que normalmente conduce a problemas mal condicionados. Las coordenadas homogéneas son sustituidas por coordenadas paramétricas. Para la realización de este algoritmo son necesarias operaciones primitivas de corte entre estructuras en los dibujos. A tal efecto se ha desarrollado también un conjunto de algoritmos con operaciones vectoriales, que simplifica los cálculos respecto a las fórmulas corrientes geometría analítica, además de aumentar la exactitud. Los mismos algoritmos se emplean para la extracción de curvas de nivel de las estructuras multiformes, lo cual reduce el espacio ocupado por el paquete gráfico.

Abstract

A set of algorithms for processing and displaying 2-D and 3-D computer graphic structures is introduced. The 2-D section has as objective an optimum way for transforming images into graphics and vice versa, while the 3-D one is oriented for creating a complete system of computer graphics displays.

The transformation of images into graphics has been carried out by means of a vectorization operation. The vectorization has some preprocessing steps, such as the image enhancement, the segmentation, the optional edge extraction and the thinning, and some postprocessing steps, such as the smoothing. All the experiments about these techniques have been checked, and a set of new algorithms overcoming the failures found in the previous ones has been developed. The core is the creation of a new algorithm for vectorization, very efficient, able to reduce the particular situations to the general case and it has not been found any errors. All the operations performed in the proposed algorithms are logical, therefore allowing an easy hardware implementation. The smoothing operation is computed using parametric coordinates and vectorial operations, avoiding ill-conditioned problems and increasing its efficiency. So, this way of transforming images into graphics yields a higher quality than the previous ones.

The 3-D display system removes the deficiencies of all the algorithms published until now. The core is the design of a new algorithm for hiding lines, able to solve all the penetrations and self-penetrations of the non-uniform structures displayed at the picture, furthermore generating the intersections and self-intersections among the surfaces. The algorithm works optimally in the sharp vertices environments, and this feature makes it very superior to all others until now. All the operations are carried out with vectors, removing the use of determinants which usually lead to ill-conditioned problems. The homogeneous coordinates are replaced by parametric ones. Some primitive operations of generating intersections among the structures of the picture are needed for implementing this algorithm. Therefore, a set of algorithms with vectorial operations has been developed too, simplifying the computer work with respect to the usual formulae of analytic geometry, as well as increasing the accuracy. The same algorithms are applied to the extraction of level curves of the non-uniform structures, reducing the storage requirements for the software graphic set.

Agradecimientos

En este punto el autor desea hacer público su agradecimiento a todas las personas que han hecho posible en alguna forma que estos trabajos llegasen a feliz término.

A mi director de tesis y, por encima de todo, gran amigo, Narciso García Santos, por su constante e imprescindible apoyo en momentos decisivos y por sus inestimables ayudas y comentarios.

Al Departamento de Investigación y Desarrollo de Telefónica, donde ha sido desarrollada la parte de estructuras 2-D, y, sobre todo, a la Subdirección de Investigación Aplicada donde en concreto se ha realizado y a los delineantes que hicieron las figuras. En especial a Gonzalo Gutiérrez Aguado, fotógrafo del Departamento, por su interés en que los revelados y positivados de fotografías y diapositivas se hiciesen con la mayor calidad y rapidez posibles.

Al Centro Científico IBM-Madrid, donde ha sido realizada la parte de estructuras 3-D, y, en especial, a José López Navalón por sus comentarios y a María Teresa Farrás Tarragona, operadora del Centro, por su paciencia infinita de los viernes por la tarde durante tanto tiempo.

Al Laboratorio de Física de la E.T.S. de Ingenieros de Telecomunicación de Madrid, y en especial a Claudio Aroca y a Pedro S. Sánchez, persona esta última muy entrañable para mí, por las fotografías de los dominios magnéticos utilizadas.

A mi grande y antiguo amigo José Jacobo Storch de Gracia y Asensio, por su ayuda en el momento de realizar las fotografías y por las fotos de elementos arqueológicos que aparecen y que él me facilitó.

A mis padres, por su inestimable colaboración general, desde el principio de mis estudios hasta el momento de la coronación de la carrera con esta tesis, alentando constantemente en los momentos de desfallecimiento, y a los que cualquier agradecimiento por mi parte quedaría pequeño con cualesquiera palabras que lo intentase describir. En particular, por todas las ideas que me han aportado sobre la estética y composición de las figuras que aparecen en este escrito.

Tabla de Materias

1.0 Introducción	1
2.0 Tratamiento de estructuras 2-D	5
2.1 El problema de la vectorización	5
2.2 Descripción de un proceso de vectorización	5
2.3 El proceso propuesto de vectorización	6
2.3.1 Mejora de imagen	7
2.3.2 Extractores de bordes en imágenes multinivel	15
2.3.3 Segmentación	17
2.3.4 Adelgazamiento	21
2.3.5 Extractores de bordes en imágenes binarias	28
2.3.6 Vectorización	32
2.3.7 Suavizado	36
2.4 Recuperación de la imagen raster	37
2.5 El algoritmo propuesto de vectorización	40
2.5.1 Definiciones previas	40
2.5.2 El núcleo del algoritmo	41
2.5.3 Situaciones particulares	43
2.5.4 Formatos de almacenamiento	48
2.5.5 Codificación final de las estructuras	49
2.6 Medida de distancias entre líneas de una imagen en entornos no conexos	50
2.6.1 Definiciones previas	51
2.6.2 Búsqueda del vector más cercano	51
2.6.3 Poligonalización	54
2.6.4 Proceso posterior	57
2.6.5 Presentación	60
3.0 Tratamiento de estructuras 3-D	67
3.1 Sistema propuesto de representación 3-D	67

3.1.1 Necesidad del sistema propuesto y descripción general	67
3.1.2 Definición de primitivas	68
3.1.3 Creación de una base de datos de líneas	69
3.1.4 Creación de una base de datos de superficies	70
3.1.5 Intersección de superficies	71
3.1.6 Operaciones globales de perspectiva	72
3.1.7 Procedimientos de presentación	75
3.1.8 Resultados	78
3.2 Algoritmo propuesto de líneas ocultas con superficies multifórmes	90
3.2.1 Descripción general	90
3.2.2 Entradas al sistema	92
3.2.3 Conversión de la base de datos superficial	92
3.2.4 El núcleo del algoritmo	92
3.2.5 Corte entre líneas y triángulos planos en el espacio 3-D	95
3.2.6 Corte entre un segmento y un triángulo plano en el espacio 3-D	95
3.2.7 Reducción de los trozos ocultos de un segmento	102
3.2.8 Optimización para el dibujo	102
4.0 Ecuaciones vectoriales propuestas 2-D y 3-D	103
4.1 Proyección de un punto sobre un triángulo plano según la dirección de un vector en el espacio 3-D	103
4.2 Corte de una recta contra un plano en el espacio 3-D	104
4.3 Corte de un segmento contra un triángulo plano en el espacio 3-D	105
4.4 Proyección de un vector en el espacio 2-D sobre dos direcciones no ortogonales	105
4.5 Corte de líneas en el espacio 2-D	107
4.6 Corte de segmentos en el espacio 2-D	108
4.7 Corte de planos en el espacio 3-D	109
4.8 Corte de triángulos planos en el espacio 3-D	111
5.0 Conclusiones	117
6.0 Aplicaciones de los algoritmos generados	119
6.1 Transmisión de firmas y mapas por línea telefónica	120
6.1.1 Necesidad de la transmisión de firmas	120
6.1.2 Información en firmas y mapas	120
6.1.3 Comparación con el FAX. Factor de compresión	121

6.1.4 Adquisición de la firma: imagen y gráfica	122
6.1.5 Tratamiento	122
6.1.6 Resultados	126
6.2 Medida de distancias entre dominios magnéticos	130
6.2.1 Necesidad de la medida de distancias	130
6.2.2 Necesidad de la automatización del proceso	132
6.2.3 Posible adquisición gráfica de los dominios	132
6.2.4 Tratamiento de las imágenes raster	132
6.2.5 Resultados	134
6.3 Tratamiento de imágenes arqueológicas	135
6.3.1 Necesidad del tratamiento de imágenes arqueológicas	135
6.3.2 Adquisición de las imágenes	137
6.3.3 Elementos empleados	138
6.3.4 Resultados	149
6.4 Otras aplicaciones	149
7.0 Líneas futuras de investigación	153
Apéndice A. Sistemas de adquisición de imágenes y gráficos	155
A.1 Adquisición de imágenes con cámara	155
A.1.1 La cámara	155
A.1.2 El procesador de imagen	155
A.1.3 El controlador	156
A.2 Adquisición de imágenes por microdensitómetro	157
A.3 Adquisición de gráficos por tablilla digitalizadora	160
Apéndice B. Sistemas de presentación y tratamiento digital de imágenes	161
Apéndice C. Presentación de algunos sistemas prácticos propuestos	163
C.1 Sistema propuesto de operación raster-vector	164
C.2 Sistema propuesto de mejora de imagen	165
C.3 Sistema propuesto de tratamiento de firmas adquiridas como gráficos	169
C.4 Sistema propuesto de búsqueda de líneas cercanas y formación de mapas de distancias	170
C.5 Sistema propuesto de representación y manejo de estructuras 3-D	175
Apéndice D. Diccionario de términos técnicos	181

Referencias	185
--------------------------	------------

Figuras

1. Dominios magnéticos originales	8
2. Subsector de los dominios magnéticos originales	8
3. Histograma de los dominios magnéticos originales	9
4. Dominios magnéticos con corrección de máxima entropía global	9
5. Histograma de la corrección de máxima entropía global, dominios magnéticos	10
6. Dominios magnéticos con corrección de máxima entropía local ventana de 3x3	10
7. Dominios magnéticos con corrección de máxima entropía local ventana de 5x5	11
8. Dominios magnéticos con corrección de máxima entropía local ventana de 7x7	11
9. Histograma de los dominios con corrección de máxima entropía local, ventana de 3x3	12
10. Histograma de los dominios con corrección de máxima entropía local, ventana de 5x5	12
11. Histograma de los dominios con corrección de máxima entropía local, ventana de 7x7	13
12. Letras originales	14
13. Letras con corrección de máxima entropía global	14
14. Perro original	16
15. Perro con filtro de Sobel	16
16. Dominios magnéticos con segmentación de umbral fijo	18
17. Dominios magnéticos con segmentación de umbral dinámico, ventana pequeña (3x3)	18
18. Dominios magnéticos con segmentación de umbral dinámico, ventana mediana (5x5)	19
19. Dominios magnéticos con segmentación de umbral dinámico, ventana correcta (7x7)	19
20. Letras con segmentación de umbral fijo	20
21. Onda original	20
22. Onda con segmentación de umbral fijo	21
23. Adelgazamiento de líneas con anchura igual a 2 píxeles	22
24. Adelgazamiento de nudos con número par de píxeles	22
25. Adelgazamiento de escalones con número par de píxeles	23
26. Esquema octal de una máscara entorno de un punto	23
27. Conjunto de máscaras empleado para el adelgazamiento	24
28. Primer conjunto simplificado de máscaras	25
29. Segundo conjunto simplificado de máscaras	26

30.	Dominios magnéticos adelgazados	27
31.	Onda adelgazada	27
32.	Letras adelgazadas	28
33.	Conjunto de máscaras empleado para la extracción de bordes en imágenes binarias	29
34.	Subsector del perro original	29
35.	Perro con segmentación de umbral fijo	30
36.	Perro con extractor de bordes para imagen binaria	30
37.	Onda con extractor de bordes para imagen binaria	31
38.	Dominios magnéticos con extractor de bordes para imagen binaria	31
39.	Letras con extractor de bordes para imagen binaria	32
40.	Dominios magnéticos vectorizados	33
41.	Onda vectorizada	33
42.	Perro con bordes vectorizados	34
43.	Letras vectorizadas	34
44.	Superposición de extractor de bordes y vectorización del esqueleto	35
45.	Vectores empleados en un suavizado	37
46.	Base de datos de vectorización de dominios magnéticos	38
47.	Gráfico inicial para hacer imagen raster	39
48.	Imagen raster a partir de gráfico	39
49.	Esquema de actualización de máscaras	42
50.	Caso particular de estructuras arborescentes	44
51.	Caso particular de píxel envuelto en nudo	44
52.	Caso particular de nudo con número par de píxeles	45
53.	Primer conjunto de máscaras para casos particulares	45
54.	Segundo conjunto de máscaras para casos particulares	46
55.	Tercer conjunto de máscaras para casos particulares	46
56.	Cuarto conjunto de máscaras para casos particulares	47
57.	Suavizado erróneo sin final de línea artificial	47
58.	Orientación de segmentos de línea	52
59.	Esquema general de segmentos maestro y esclavos	53
60.	Búsqueda ortogonal y en abanico de líneas cercanas	55
61.	Búsqueda errónea en los vértices	56
62.	Polígono típico de rellenado	57
63.	Caso de cambio de esclavo al recorrer el maestro	58
64.	Casos posibles de cambio de un esclavo sobre el mismo maestro	59
65.	Poligonalización en el caso particular de corte de segmentos	60

66.	Posibles cortes al rellenar por verticales un polígono de cuatro lados	61
67.	Intersección de polígonos debida a la búsqueda en abanico	62
68.	Intersección de polígonos debida a la búsqueda desde distintos maestros	62
69.	Mapa de distancias entre líneas	63
70.	Imagen vectorizada para medir distancias entre líneas	63
71.	Mapa de distancias sobre imagen vectorizada	64
72.	Esquema general de la medida de distancias entre las líneas de una imagen	65
73.	Perspectiva cónica	73
74.	Hexágono correspondiente a un cubo proyectado	77
75.	Cuadrado correspondiente a un cubo visto desde arriba o abajo	77
76.	Cilindro con ejes en perspectiva ortogonal	79
77.	Cilindro con ejes en perspectiva cónica	80
78.	Envoltorio del campo eléctrico de una onda electromagnética	81
79.	Esfera en perspectiva ortogonal	82
80.	Prisma autopenetrado y abierto en cónica	83
81.	Composición de la misma primitiva tres veces con giros y penetraciones	84
82.	Composición de la misma primitiva siete veces con giros y penetraciones	85
83.	Composición de líneas y superficies con penetración	86
84.	Composición de primitivas penetrantes	87
85.	Composición de dos primitivas	88
86.	Curvas de nivel de función sinc bidimensional	89
87.	Imagen original para curvas de nivel	90
88.	Curvas de nivel de la imagen original anterior	91
89.	Casos posibles de proyección de un polígono alabeado de 4 lados	93
90.	Posición relativa de las proyecciones en 2-D de una recta y un triángulo	96
91.	Signos en los productos cruzados en el caso de una recta contra un triángulo proyectados en 2-D	96
92.	Posición relativa de las proyecciones en 2-D de un segmento y un triángulo	98
93.	Signos en los productos cruzados en el caso de un segmento contra un triángulo proyectados en 2-D	98
94.	Proyección de un punto sobre un triángulo según la dirección de un vector en el espacio 3-D .	104
95.	Corte de un segmento contra un triángulo plano en el espacio 3-D	106
96.	Casos posibles de los tres parámetros cortando un segmento contra un triángulo plano en el espacio 3-D	106
97.	Proyección de un vector en el espacio 2-D sobre dos direcciones no ortogonales	107
98.	Corte de líneas en el espacio 2-D	108
99.	Corte de segmentos en el espacio 2-D	109

100. Corte de planos en el espacio 3-D	110
101. Corte de triángulos planos en el espacio 3-D	112
102. Segmentos a examinar en corte de triángulos planos en el espacio 3-D	113
103. Tabla final de corte de triángulos planos en el espacio 3-D	114
104. Firma adquirida por tablilla gráfica digitalizadora	123
105. Firma gráfica con suavizado superpuesto	123
106. Firma raster a partir de firma gráfica	124
107. Firma adquirida por digitalización de imagen raster	124
108. Firma raster con segmentación	125
109. Firma raster con adelgazamiento	125
110. Firma raster vectorizada	126
111. Descripción de la base de datos de la firma vectorizada	127
112. Firma adquirida por digitalización de imagen raster con trazo grueso	128
113. Firma raster de trazo grueso con segmentación	128
114. Firma raster de trazo grueso con adelgazamiento	129
115. Firma raster de trazo grueso vectorizada	129
116. Eliminación de ramas sobre la firma vectorizada de trazo grueso	130
117. Especificación de la base de datos de la firma vectorizada de trazo grueso	131
118. Dominios vectorizados para mapa de distancias	133
119. Mapa de distancias o de energía de anisotropía de material magnético en función de sus dominios	133
120. Superposición del mapa de distancias a los dominios vectorizados	134
121. Elemento arqueológico original	136
122. Histograma del elemento arqueológico original	137
123. Elemento arqueológico con corrección de máxima entropía local y ventana muy pequeña ...	138
124. Elemento arqueológico con corrección de máxima entropía local, ventana de 27x27	139
125. Histograma de la corrección de máxima entropía local con ventana muy pequeña	140
126. Histograma de la corrección de máxima entropía local, ventana de 27x27	140
127. Tres valores de factor de realce: bajo, correcto y alto	141
128. Elemento arqueológico con factor de realce igual a 2,0	141
129. Realce de bordes con factor 2,3 sobre la corrección de máxima entropía local ventana de 27x27.	142
130. Corrección de máxima entropía global sobre realce hecho a la corrección local (27x27).	143
131. Realce de bordes sobre las correcciones anteriores	144
132. Corrección logarítmica del elemento arqueológico con realce de bordes	145
133. Realce de bordes con factor 3,2 sobre la corrección no lineal con $k = 1,5$	146
134. Realce de bordes con factor 3,2 sobre la corrección no lineal con $k = 2,3$	147

135. Realce de bordes con factor 3,2 sobre la corrección no lineal con $k=3,2$	148
136. Aplicación del sistema 3-D a un diseño de oscilador de microondas	150
137. Aplicación del sistema 3-D a un segundo diseño de oscilador de microondas	151
138. Adquisición de imágenes raster por cámara	158
139. Función típica de distribución de luminancia en una imagen	159
140. Camino óptico de un microdensitómetro	160
141. Operación de vectorización y su inversa	164
142. Opciones generales	165
143. Opciones de tratamiento global de histograma	166
144. Opciones de realce de bordes	167
145. Opciones de extracción y bordes y su presentación	168
146. Opciones de manejo de firmas como gráficos y su conversión a imágenes raster	169
147. Selección de imagen de entrada	170
148. Tratamiento previo	171
149. Parámetros de recorrido	172
150. Actualización de recorrido	173
151. Parámetros de resultado	174
152. Inicialización de terminal	175
153. Parámetros de entrada al sistema	176
154. Entrada por fichero o tablas	177
155. Dibujo tridimensional	178
156. Generación de curvas de nivel	179
157. Presentación en trazador de curvas	180

1.0 Introducción

Las técnicas de gráficos por ordenador han sufrido una gran revolución en los últimos años. La popularización de los sistemas informáticos y el desarrollo de los ordenadores personales han sido causas decisivas para ello. La simplificación de multitud de algoritmos, reduciendo la categoría de sus operaciones internas, ha hecho posible en muchos casos su realización en hardware.

El tratamiento digital de imágenes es algo aparentemente distinto. Sin embargo, los gráficos y las imágenes pertenecen a la misma familia. Un gráfico es una imagen que produce el ordenador cuando tiene como entrada una descripción. Una imagen es un resultado que produce un ordenador ante otra imagen de entrada. Este paralelismo ha hecho que se pensase en impulsar el realismo de los gráficos. Los resultados del estudio de iluminaciones y sombras crean gráficos que producen la sensación de imágenes reales. El avance ha sido particularmente espectacular en estos últimos años (8).

De este parentesco entre imágenes y gráficos puede extraerse algunas consecuencias. La aplicación conjunta de técnicas clásicas de tratamiento digital de imagen (10,11,13,14), y las de gráficos por ordenador (1,2,4,5,6), produce resultados excepcionales. Usualmente se combina el tratamiento digital de imágenes con técnicas también clásicas de proceso de señal (15,16), y herramientas matemáticas potentes de uso específico (17,18,19,20,21,72).

Los gráficos tienen una especial particularidad respecto al tratamiento de imágenes. En un tratamiento de imagen, un algoritmo diseñado con objeto de realizar alguna operación puede llevarse a cabo sin ningún tipo de problema en la mayoría de los casos. En los gráficos, la aplicación de un algoritmo suele presentar con alta probabilidad algunos casos particulares que hay que tratar por separado. El tratamiento de estos casos particulares normalmente implica hacer más lento el algoritmo, y, por ello, todos los algoritmos gráficos llevan un compromiso entre la magnitud del problema a resolver y el tiempo empleado en hacerlo. Muchas veces se hace un estudio previo de la probabilidad de que aparezca un suceso y la intensidad con que aparece. Si ambas pueden despreciarse frente a otros parámetros del gráfico, y la solución al problema total es costosa, se ignora la posible situación. Esto hace que muchas veces los algoritmos gráficos presenten lagunas. Dos pueden ser las causas que dan lugar a situaciones particulares en el tratamiento de gráficos: la discretización de la imagen analógica correspondiente y la necesaria aritmética en

punto flotante con sus truncamientos.

Como consecuencia de todo lo expuesto, queda una puerta abierta a la investigación en el sentido de solucionar los problemas de casos particulares con algoritmos cada vez más sencillos. Estas soluciones pasan normalmente por la modificación completa del algoritmo general. Un algoritmo puede mejorarse siempre en un porcentaje dado, pero se llega a una barrera asintótica en la optimización. En estos casos es mejor adoptar un algoritmo completamente nuevo.

Otra fase que irrumpe hoy día con fuerza es la sustitución de los algoritmos clásicos por algoritmos vectoriales. La puerta a la investigación también está abierta en este sentido. Los algoritmos vectoriales evitan el uso de determinantes y funciones trigonométricas, que tienen serios inconvenientes en un ordenador. Los primeros crean habitualmente problemas mal condicionados, además de su lentitud de cálculo. Las segundas, además también de su lentitud, pueden dar problemas de infinitos, con lo cual las diferencias cortas entre puntos próximos carecen de sentido al evaluarse en la máquina. Por otra parte, los algoritmos vectoriales son idóneos para el uso de coordenadas paramétricas. Este tipo de coordenadas es el más rápido y el que mejor funciona dentro del ordenador.

Muchos son los intentos de ir resolviendo eficazmente los casos particulares de los algoritmos clásicos en gráficos (1,2,4,5,6), con métodos cada vez más eficientes. Por ejemplo, en el caso de segmentación y tratamiento de histogramas se pueden encontrar experimentos (30,36,23,24,37,38,41,43,44,45), donde hay intentos de introducir tratamiento y parámetros de texturas. El tratamiento de histogramas es una operación que no aparece normalmente en la bibliografía como previa para adecuar una imagen a su vectorización, aunque implícitamente va combinada con la segmentación, que es una operación previa obligada a la vectorización que se propone, y eso sí se puede leer en algunos antecedentes. En cuestiones de adelgazamiento hay muchos ensayos (25,26,27,29,31,46,48). Particular interés tiene el expuesto en (46), aunque máscaras basadas en las ideas de (31) sean más eficaces, aplicadas al propio algoritmo de (46). Del tema de vectorización hay muy pocas cosas publicadas (28,32,33,34,35). Las de mayor interés son las ideas de (28,32). El suavizado de estructuras vectorizadas empieza a tener algoritmos pseudovectoriales en (40).

En otro orden de cosas, los algoritmos de zonas vistas y ocultas no salen a la luz fundamentalmente por razones comerciales. Los más clásicos (82,83,84,85,86) han sido superados por experimentos publicados en (87,88,89,90,91,92,93). Por otra parte, la mayoría de ellos resuelven sólo el caso de estructuras uniformes, y se complican mucho cuando aparecen estructuras con distintos grados de libertad en el espacio del gráfico. Asimismo, no resuelven la penetración de estructuras.

Existen diversas formas de modelado de sólidos. El lector interesado puede ver algunas en (75,76,77,78,80). Aquí se propone un sistema intermedio entre el modelo de alambre y el modelado por parches. Algún antecedente en sistema de representación 3-D relevante puede verse en (60,61).

Como ya se ha explicado, todos los algoritmos mencionados no resuelven la totalidad de las situaciones particulares, lo que los hace inútiles para determinadas aplicaciones. Todos los algoritmos mencionados han sido estudiados y se ha conseguido ver cuáles son los casos particulares que no resuelven. A la luz de esta información, ha sido posible la realización de nuevos algoritmos que resuelven más casos particulares, y, alguna vez, la totalidad de los mismos. Esta expresión de "la totalidad de los mismos", debe entenderse como siempre en el sentido empleado en los gráficos. Este sentido consiste en que hasta la fecha no se ha encontrado una situación que no resuelva satisfactoriamente sin necesidad de esquivarla.

El objetivo general aquí propuesto es, pues, la consecución de un sistema de generación y tratamiento de gráficos 2-D y 3-D, en combinación o a partir de técnicas de tratamiento digital de imagen, persiguiendo aplicaciones concretas que ya han surgido, y que se describen con posterioridad. El sistema debe ser tal que no cometa errores en ninguna situación particular, y, por supuesto, si lo hiciere, la probabilidad y la intensidad del error han de ser menores que en el último algoritmo publicado. Los temas en los que se centrará el interés serán la conversión de imágenes raster en gráficos vectoriales y viceversa, con un algoritmo eficiente de vectorización, y la creación de un sistema de representación 3-D, con un rápido algoritmo de líneas ocultas capaz de resolver cualquier situación sobre estructuras multiformes, todo ello apoyado en un nuevo conjunto de algoritmos vectoriales de cortes entre estructuras.

2.0 Tratamiento de estructuras 2-D

2.1 El problema de la vectorización

Se entiende por vectorizar una imagen la operación de transformación de dicha imagen en un conjunto de vectores. Es pues, la conversión de una imagen raster en un gráfico vectorial. Esta técnica encaja perfectamente entre las de extracción y codificación de esqueletos o bordes. Por ello, la vectorización es una operación que debe aplicarse sobre imágenes que presenten estructuras con direcciones predominantes.

Las estructuras representadas por los vectores producto de la vectorización pueden ser los esqueletos de aquéllas con direcciones predominantes, o sus bordes, que previamente deben ser extraídos para convertir dichas estructuras precisamente en sus bordes, que sí son algo con direcciones claramente dominantes.

Algunos ejemplos de imágenes susceptibles de ser tratadas con un proceso de vectorización son las firmas, cualquier tipo de documento escrito, dominios magnéticos de estructuras amorfas sometidas a compresión ("stripe domains"), fotografías de ríos, mapas, etc... En todas ellas existen rasgos con direcciones claramente dominantes, y la información no se pierde si son sustituidas las estructuras de la imagen por sus esqueletos o sus bordes adelgazados, que luego son codificados como vectores, según el caso. Esto hace que la transmisión de este tipo de imágenes se pueda realizar reduciendo la cantidad de información a transmitir, siendo una forma de codificación con compresión.

2.2 Descripción de un proceso de vectorización

Por todo lo expuesto, la vectorización debe pasar por un proceso de esqueletización, o por una extracción de bordes seguida de una esqueletización de éstos. Algunos algoritmos propuestos con anterioridad incluyen en el término vectorización la esqueletización, pero se ha demostrado más eficiente el adelgazamiento de estructuras sobre imágenes raster, para luego hacer la conversión a vectores de estructuras previamente adelgazadas. A su vez, los algoritmos de adelgazamiento trabajan de forma mucho más efi-

ciente si son aplicados sobre imágenes binarias, lo cual sugiere la necesidad de realizar una segmentación previa. De la misma forma, la segmentación se simplifica si previamente se aplica alguna técnica de mejora de imagen. En lo que se refiere a la vectorización de bordes, se puede realizar extrayendo los bordes sobre una imagen multinivel, o sobre la imagen después de la segmentación, viéndose esta opción como la más eficiente en la mayoría de las aplicaciones. Los algoritmos empleados para la extracción de bordes en ambos casos son claramente distintos. Por último, la vectorización se termina aplicando técnicas de suavizado de las líneas de vectores que crea, y éste es el último paso en su realización.

De esta forma, la vectorización queda realizada a través de un conjunto de operaciones ordenadas, como sigue: mejora de imagen (opcional), extracción de bordes en la imagen multinivel (opcional en este lugar), segmentación, extracción de bordes en la imagen binaria (opcional en este lugar), adelgazamiento, vectorización propiamente dicha, y suavizado.

La vectorización produce un patrón en el dominio de la propia imagen. El resultado es almacenado en dos bases de datos, donde la primera es el patrón y la segunda es su distribución sobre la imagen. Las dos bases de datos son:

1. Universo
2. Punteros

El Universo es el conjunto de vértices de los vectores formados en el proceso de vectorización. La base de Punteros consiste en sucesiones ordenadas de punteros, que señalan el orden en que aparecerán los vértices del Universo formando líneas con los segmentos rectos delimitados por dichos vértices. Son, pues, punteros sobre el Universo. Los punteros pueden repetirse, y no indicaría más que un punto de cruce entre las líneas producto de la vectorización.

2.3 El proceso propuesto de vectorización

La vectorización propiamente dicha puede llevarse a cabo sobre cualquier imagen multinivel, que presente las características adecuadas. Sin embargo, la aplicación de herramientas propias del tratamiento digital de imágenes antes de hacer la vectorización, simplifica enormemente la realización de la misma. Se describirá a continuación el conjunto de herramientas empleado, cuya filosofía puede verse en (35,57,58), aunque aquí se presenta con nuevos y más eficaces algoritmos. Algún precedente con ideas bien distintas puede encontrarse en (48), y con ideas más similares en (47).

En la parte de mejora de imagen se hace una aplicación de los métodos de corrección de histograma hacia la máxima entropía global, y se propone un algoritmo nuevo de corrección máxima entropía local capaz de superar los problemas de la corrección global. En la extracción de bordes en imágenes multinivel, se pasa revista a los diferentes métodos existentes y se escoge uno de ellos como el más eficiente para esta aplicación. En segmentación, también se analizan las técnicas existentes y se expone un nuevo algoritmo para segmentación de umbral dinámico. En adelgazamiento, se propone un algoritmo antiguo como base del que se presenta aquí como óptimo. En vectorización propiamente dicha, se propone un algoritmo completamente nuevo que suprime cualquier defecto de algoritmos anteriores y presenta una serie de cualidades que lo hacen maximalmente eficiente en todos los aspectos. En suavizado, se presenta un método nuevo basado en matemáticas vectoriales para resolver este problema.

2.3.1 Mejora de imagen

La presencia de estructuras que hagan posible la vectorización de una imagen, no implica necesariamente que la imagen esté preparada para este proceso. La fuente de imagen, normalmente fotografía, puede estar tomada con apertura de diafragma y velocidad mal escogidas, o, simplemente, con valores de compromiso para obtener información en la luz y en la sombra. Se hace imposible, pues, la aplicación de umbrales fijos generalizados en cualquier proceso posterior, e incluso los resultados de umbrales dinámicos son muy deficientes en las sombras, y costosos de cálculo.

El tratamiento para mejora de imagen introducido en este punto debe ser capaz de eliminar los gradientes de luminancia innecesarios, sin tener efectos de filtro paso bajo ni la introducción de ruido no deseado. Se trata de desordenar la información de luminancia de la imagen, para ordenar la información de bordes con posterioridad. Las técnicas de corrección de máxima entropía son idóneas para este fin. Una amplia teoría de estabilidad por métodos estadísticos sobre imágenes consideradas como sistemas no lineales realimentados en interacción con el medio exterior sustenta todo este proceso. Sin embargo, en este punto se centrará la atención sobre la realización práctica y los resultados obtenidos.

Un ejemplo puede verse sobre la imagen original de la Figura 1 en la página 8, de la que se extrae un subsector de interés que se muestra en la Figura 2 en la página 8. Se trata de los conocidos "stripe domains" (115) de una muestra de material magnético amorfo $Ni_{40} - Fe_{40} - P_{16} - B_6$. La fotografía muestra gradientes de iluminación. Su histograma se puede ver en la Figura 3 en la página 9. Las características de esta imagen hacen que directamente sea bueno el resultado de la aplicación de una corrección de tipo global de máxima entropía, y el resultado se muestra en la Figura 4 en la página 9, con su histograma en la Figura 5 en la página 10.

Figura 1. Dominios magnéticos originales

Una variante de este proceso se desarrolla aquí, y consiste en la corrección dinámica de histograma. La

Figura 2. Subsector de los dominios magnéticos originales

Figura 3. Histograma de los dominios magnéticos originales

imagen se divide en parcelas o ventanas, sobre las que se aplica una corrección de máxima entropía, y los parámetros de transformación son diferentes en cada una. Así, la corrección dinámica de histograma hace

Figura 4. Dominios magnéticos con corrección de máxima entropía global

Figura 5. Histograma de la corrección de máxima entropía global, dominios magnéticos

una transformación uniforme del mismo sobre cada ventana. Las ventanas no son disjuntas, de tal forma

Figura 6. Dominios magnéticos con corrección de máxima entropía local ventana de 3x3

Figura 7. Dominios magnéticos con corrección de máxima entropía local ventana de 5x5

que es necesaria una imagen auxiliar para la realización del método. El tamaño de la parcela es clave para una verdadera mejora. El efecto de una ventana pequeña es similar a un filtro paso alto, con la consi-

Figura 8. Dominios magnéticos con corrección de máxima entropía local ventana de 7x7

Figura 9. Histograma de los dominios con corrección de máxima entropía local, ventana de 3x3

guiente aparición de ruido, mientras que una ventana grande no elimina el gradiente de luminancia, tal

Figura 10. Histograma de los dominios con corrección de máxima entropía local, ventana de 5x5

Figura 11. Histograma de los dominios con corrección de máxima entropía local, ventana de 7x7

como se desea. Una ventana excesivamente pequeña puede proporcionar resultados que se asemejen a un dibujo, al ser un filtro demasiado paso alto. Una secuencia de corrección dinámica se puede ver en la Figura 6 en la página 10 con ventana muy pequeña (3x3), en la Figura 7 en la página 11 con ventana algo mayor (5x5) y en la Figura 8 en la página 11 con ventana casi correcta (7x7). El último resultado ya se parece a la corrección global bastante y varía poco si se sigue aumentando el tamaño de la ventana. Los respectivos histogramas se pueden ver en la Figura 9 en la página 12, en la Figura 10 en la página 12 y en la Figura 11. La primera ventana es excesivamente pequeña, lo que introduce demasiado ruido, siendo cada vez más adecuada a medida que aumenta su tamaño. Es un ejemplo donde la corrección global surte buen efecto, pero no es el caso general. En capítulos posteriores y sobre aplicaciones se podrá observar algún caso donde la corrección local de histograma es esencial.

Otro ejemplo se puede ver en la Figura 12 en la página 14. Una corrección global de histograma facilita mucho el paso posterior de la segmentación, al acentuar los contrastes. El resultado se ve en la Figura 13 en la página 14, donde a pesar de que el fondo se oscurece, las letras aparecen más negras y el contraste general aumenta.

Realmente, las técnicas de corrección global de histograma, incluidas también las de máxima entropía, lo que hacen es disminuir dicha entropía. Esto es debido a la aplicación sobre imágenes discretas (digitalizadas) y sobre un histograma de niveles discretos, de técnicas pensadas para histogramas conti-

Figura 12. Letras originales

nuos. La corrección dinámica resuelve este problema aumentando la entropía cuando las ventanas son las adecuadas, y de esta forma se la puede maximizar. Para ver este efecto basta observar los histogramas mostrados hasta este punto. No obstante, la correlación entre parcelas vecinas es ligeramente eliminada, a pesar de que las ventanas no sean disjuntas, y una corrección global de histograma sobre la corrección dinámica hace más agradable la imagen final, aún a costa de perder algo de entropía.

Figura 13. Letras con corrección de máxima entropía global

2.3.2 Extractores de bordes en imágenes multinivel

Puede tener interés la vectorización de los bordes de una imagen, ya que su estructura puede hacer que la única información verdaderamente útil resida en ellos. Ejemplos de este tipo de imágenes son los mapas.

Una algorítmica muy diferente se emplea para la extracción de bordes en el caso de imágenes multinivel respecto al caso de imágenes binarias. En este apartado se tratará el caso multinivel y más adelante, cuando se haya introducido la conversión a imagen binaria de otra multinivel, se expondrá la problemática de la extracción de bordes en imágenes binarias.

La extracción de bordes en imágenes multinivel es un tema muy estudiado por diversos autores (13). Los filtros no lineales suelen dar mejores resultados que los filtros lineales en esta operación. El filtro de Sobel es una solución de compromiso entre la simplicidad y las prestaciones. Otros tipos de filtros más complicados son capaces de separar en clases de equivalencia las distintas estructuras de la imagen original, estableciendo las correspondientes relaciones de equivalencia. Las diferentes imágenes de gradiente en las respectivas clases de equivalencia muestran realzadas las características de bordes deseadas.

Las clases de equivalencia formadas pueden ser de dos tipos, según se encuentren en el dominio de la imagen o en el de la transformada. En el dominio de la imagen las estructuras que forman cada clase de equivalencia son estructuras gráficas, tales como puntos aislados, estructuras alargadas, etc... (10,12). En el dominio de la transformada cada clase de equivalencia está asociada a un filtro paso banda, obteniéndose las estructuras sintonizadas con esa banda. Este análisis es particularmente útil en el estudio de texturas.

A continuación se puede ver una presentación de la extracción de bordes en el dominio de la imagen. En la Figura 14 en la página 16 puede verse una imagen original a la que se aplica el conocido filtrado no lineal de Sobel, que da excelentes resultados en la gran mayoría de los casos. La imagen filtrada es mostrada en la Figura 15 en la página 16.

Figura 14. Perro original

Figura 15. Perro con filtro de Sobel

2.3.3 Segmentación

Un paso que facilita mucho la vectorización es la segmentación. Consiste en convertir la imagen multinivel entrante en otra saliente que sea binaria. Se define el umbral de segmentación como el valor de luminancia por encima del cual la imagen de salida será blanca, y por debajo del cual será negra. El umbral de segmentación es un tema muy ensayado en la literatura (6,30,36,37,38,42,43,44,45), con una revisión de tipo general en (41), pero deja muchas puertas abiertas.

Las técnicas de segmentación se dividen en dos grandes grupos de acuerdo con su umbral:

1. Umbral fijo
2. Umbral dinámico

Algunas técnicas de umbral fijo pueden verse en (6,35). Son aplicables sobre imágenes que presenten profundos contrastes entre la luz y la sombra, homogéneamente repartidos y con similar profundidad. Las técnicas de corrección de histograma descritas en el apartado anterior son idóneas para convertir imágenes que no posean estas características en imágenes aptas para ser segmentadas con umbral fijo. Realmente, las técnicas de segmentación propuestas en la literatura se basan en la contemplación del histograma de la imagen, y mejorar éste es precisamente el objetivo de la mejora de imagen, tal como se desprende de lo expuesto con anterioridad. No obstante, el tamaño de la ventana en el caso de corrección de histograma con umbral dinámico, puede dejar lagunas si existen en la imagen estructuras básicas de tamaños bien diferenciados, que necesitan un tamaño de ventana de compromiso. Si éste es el caso, la segmentación con umbral fijo deja bastante que desear, haciéndose necesario el uso de umbral dinámico también en la segmentación.

Como ya se ha visto, existen algunos ensayos en segmentación con umbral dinámico en la bibliografía. Se propone aquí un umbral dinámico igual al nivel de gris correspondiente a una probabilidad fija y dada para toda la imagen, calculando el histograma sobre ventanas de la misma. Nuevamente, al igual que en la mejora de imagen, la elección del tamaño de la ventana es un factor determinante de la calidad de la imagen segmentada. El uso de ventanas excesivamente grandes hace un filtrado paso bajo no lineal a la imagen no deseado, mientras que una ventana excesivamente reducida conduce a la introducción de ruido, ya que equivale a un filtro paso alto, no lineal también.

Sobre la imagen original de la Figura 2 en la página 8, y tras los tratamientos previos descritos, se realiza una segmentación de umbral fijo que se muestra en la Figura 16 en la página 18. Una segmen-

Figura 16. Dominios magnéticos con segmentación de umbral fijo

tación de umbral dinámico con ventana estrecha se muestra en la Figura 17 en la página 18, con ventana

Figura 17. Dominios magnéticos con segmentación de umbral dinámico, ventana pequeña (3x3)

Figura 18. Dominios magnéticos con segmentación de umbral dinámico, ventana mediana (5x5)

más correcta en la Figura 18 en la página 19, y con ventana prácticamente correcta en la Figura 19 en la página 19. La probabilidad escogida es de 0.65, ya que se revela este valor como el que mejores resultados

Figura 19. Dominios magnéticos con segmentación de umbral dinámico, ventana correcta (7x7)

Figura 20. Letras con segmentación de umbral fijo

provoca.

Sobre la imagen de la Figura 12 en la página 14 o sobre la Figura 13 en la página 14 se muestra otro ensayo de segmentación. En este caso la segmentación de umbral fijo da excelentes resultados que se muestran en la Figura 20. Otro experimento se hace sobre la Figura 21. Esta figura procede de trabajos realizados en torno al estudio de modelos visuales y síntesis de texturas que se ve en (120), y presenta

Figura 21. Onda original

Figura 22. Onda con segmentación de umbral fijo

unas características idóneas para su tratamiento como imagen test. Una segmentación de umbral fijo se muestra en la Figura 22.

2.3.4 Adelgazamiento

El adelgazamiento de una imagen tiene por objeto convertir una imagen con estructuras claramente definidas con una dimensión predominante, en otra imagen donde aparezcan sólo los esqueletos de estas estructuras. De ahí que se llame también esqueletización a esta operación. Se realiza esta operación sobre imágenes previamente segmentadas, y existe gran cantidad de ensayos en la literatura (25,26,27,29,31,46), no siendo definitiva ninguna de las soluciones.

El adelgazamiento es un proceso iterativo, que se realiza moviendo un conjunto de máscaras sobre la imagen, convirtiendo a negros los píxeles cuyo entorno coincida con alguna de las máscaras. Cuando se ha pasado el conjunto de máscaras por toda la imagen, todas las estructuras han sido rebajadas por los bordes. El proceso se repite iterativamente, hasta que no se pueda rebajar en ningún sitio y ningún píxel sea convertido a negro. Cada máscara es, pues, un detector de bordes muy particular, capaz de determinar un borde en cualquier dirección. Sin embargo, este conjunto de detectores de bordes debe ser muy estricto, en el sentido de que además de rapidez en la operación, dada la cantidad de veces que se usan

Figura 23. Adelgazamiento de líneas con anchura igual a 2 píxeles

(una por píxel en cada iteración), no debe fallar nunca, debe mantener continuidad en la estructura adelgazada, si así era en la imagen original, y debe detectar todas las clases de equivalencia descritas en los detectores de bordes de imágenes multinivel de una sola vez, y sin operaciones de proyección sobre subespacios. Las máscaras más habituales son de 3x3, y, siendo tan simples, dan sin embargo excelentes resultados.

Figura 24. Adelgazamiento de nudos con número par de píxeles

Figura 25. Adelgazamiento de escalones con número par de píxeles

El adelgazamiento tiene muchos problemas en estructuras con un número par de píxeles blancos de anchura. Cuando se llega en una iteración a tener dos píxeles de anchura, el problema consiste en decidir que sólo hay que retirar uno, y cuál hay que retirar, puesto que no se puede escoger al azar. Si esto se hiciera así, y la estructura terminase en un nudo grueso previamente adelgazado podría presentarse una discontinuidad. Similares problemas aparecen en nudos con número par de píxeles de lado, donde el problema anterior se magnifica al tener las dos dimensiones en juego. También las estructuras escalonadas

Figura 26. Esquema octal de una máscara entorno de un punto

Figura 27. Conjunto de máscaras empleado para el adelgazamiento

muestran semejantes problemas. Una descripción del problema y la evolución de la estructura con número par de píxeles de anchura puede verse en la Figura 23 en la página 22. La descripción del problema del nudo de número par de píxeles de lado se puede ver en la Figura 24 en la página 22, y la problemática de un tipo de escalón en la Figura 25 en la página 23. En caso de utilizar un algoritmo que, por ser más

Figura 28. Primer conjunto simplificado de máscaras

sencillo, no garantice la continuidad, es conveniente aplicar después del adelgazamiento una operación de unión de estructuras sobre la imagen (39).

Se propone aquí un algoritmo basado en las ideas de (46), con un conjunto de máscaras extraído de las ideas expuestas en (31), ya que el conjunto de máscaras propuesto por el autor de (46) provoca adelgazamientos que favorecen determinadas direcciones en algunas imágenes. El conjunto de máscaras aquí propuesto elimina este efecto. Todas las operaciones son del tipo lógico, con lo cual las comparaciones con las máscaras son muy rápidas en el ordenador, y una posible realización hardware se facilita muchísimo. Precisamente ésta es la idea expuesta en (46).

Siguiendo el esquema de identificación de máscara por números en octal expuesto en (6), y mostrado en la Figura 26 en la página 23, el conjunto de máscaras propuesto y utilizado se muestra en la Figura 27 en la página 24. Aunque no es éste el caso, es de destacar que en algunos conjuntos de máscaras sólo existe máscara detectora de un sentido en algunas direcciones, dado que el otro sentido queda eliminado por otras máscaras no pertenecientes a la misma dirección. Así, el conjunto de máscaras está

Figura 29. Segundo conjunto simplificado de máscaras

optimizado en cuanto a su cantidad. Por otro lado, si esas máscaras que no están existiesen, se verificaría en cada iteración un rebajado dos veces sobre cada dirección en el sentido eliminado por las otras máscaras del conjunto. Esto provocaría que determinadas direcciones se favoreciesen en el adelgazamiento, y es algo que debe evitarse a toda costa.

El algoritmo consiste en crear una imagen de entornos donde cada elemento contiene el octal equivalente a su entorno, dispuesto para ser comparado con las máscaras, y actualizada únicamente al final de cada iteración. Al mismo tiempo, cada vez que se decide eliminar un píxel en una iteración, dado que se concluye que pertenece a un borde, se actualiza su entorno, creándose una nueva imagen entornos corregidos en el transcurso de la iteración dada con los octales correspondientes. Cuando se va a retirar un píxel, el test con las máscaras debe ser verificado simultáneamente con las dos imágenes de entornos. Si esto no es así, el píxel correspondiente no se elimina. Esto soluciona las situaciones de líneas con número par de píxeles de anchura, los escalones y los cruces de número par de píxeles. El proceso se hace para todos los píxeles de la imagen en cada iteración. Se itera hasta que ningún píxel necesite ser eliminado en alguna iteración. La convergencia del proceso es muy rápida y garantiza continuidad en los esqueletos obtenidos. El algoritmo está optimizado en el sentido de que no actúa sobre píxeles que van siendo negros durante el proceso.

El conjunto de máscaras propuesto en (46) puede verse en dicha referencia y es más complicado que el que aquí se propone. Otros conjuntos más simples dan buenos resultados en determinadas situaciones donde las exigencias o la aplicación no sean muy estrictas. Dos de ellos se presentan en la Figura 28 en la

Figura 30. Dominios magnéticos adelgazados

página 25 y en la Figura 29 respectivamente.

Figura 31. Onda adelgazada

Figura 32. Letras adelgazadas

Algunos resultados de adelgazamiento pueden verse sobre las imágenes segmentadas de la Figura 16 en la página 18 de la Figura 22 en la página 21 y de la Figura 20 en la página 20 respectivamente en la Figura 30 en la página 27, en la Figura 31 en la página 27 y en la Figura 32.

2.3.5 Extractores de bordes en imágenes binarias

Se han expuesto con anterioridad algunas consideraciones sobre extracción de bordes en imágenes multinivel. Los mismos algoritmos pueden emplearse en el caso de imágenes binarias. Sin embargo, una mayor efectividad se consigue diseñando un algoritmo específico para este tipo de imágenes, que muestran particularidades muy acusadas.

El algoritmo que aquí se propone consiste en la aplicación de las ideas de adelgazamiento, pero justamente al revés. En vez de buscar el esqueleto de una imagen, se buscan los bordes. En adelgazamiento, cada iteración va encaminada a eliminar píxeles si se detecta que residen en un borde. En el caso de querer detectar el borde, sólo hay que quedarse con los píxeles que residen en el borde de la primera iteración del adelgazamiento, con un conjunto reducido de máscaras, que se muestran en la Figura 33 en la página 29, y eliminar el resto de los píxeles de la imagen.

El algoritmo seguido con las máscaras de la Figura 33 en la página 29 consiste en hacer, sobre una imagen segmentada, un "o" inclusivo entre cada una de las máscaras que se muestran y la máscara correspondiente al punto según sus vecinos y de acuerdo con los mismos criterios empleados en el adelgazamiento. Si el resultado es igual a alguna máscara de la Figura 33 en la página 29, significa que el píxel

Figura 33. Conjunto de máscaras empleado para la extracción de bordes en imágenes binarias

tiene 0's (píxeles de luminancia nula) en al menos los vecinos que están nulos en esa misma máscara. En ese caso, el píxel correspondiente se elimina por pertenecer al interior de alguna estructura.

Este procedimiento tiene una gran ventaja sobre la aplicación de técnicas de extracción de bordes en imágenes multinivel. Todas las operaciones implicadas son lógicas, como ya se describió en el adelgazamiento, lo que aumenta la velocidad de proceso y permite una muy fácil realización hardware, que aún lo haría más rápido.

Normalmente, el conjunto de máscaras debería detectar el borde directamente. No obstante, el algoritmo es propiamente de adelgazamiento, y esto es lo que optimiza. El borde puede contener algunas zonas

Figura 34. Subsector del perro original

Figura 35. Perro con segmentación de umbral fijo

con anchura superior a un elemento de imagen (píxel). Esto queda solucionado inmediatamente si sobre

Figura 36. Perro con extractor de bordes para imagen binaria

Figura 37. Onda con extractor de bordes para imagen binaria

el borde detectado se aplica un adelgazamiento. El proceso es muy rápido, pues el número de píxeles que quedan después de detectar el borde y eliminar el resto es muy pequeño, de tal forma que el adelga-

Figura 38. Dominios magnéticos con extractor de bordes para imagen binaria

Figura 39. Letras con extractor de bordes para imagen binaria

zamamiento raramente necesita más de una o dos rápidas iteraciones, ya que evita actuar sobre píxeles negros.

Algunos resultados pueden verse sobre el original de la Figura 14 en la página 16, al que se le aplica previamente una segmentación de umbral fijo, según la Figura 35 en la página 30 sobre el subsector de la Figura 34 en la página 29. Su extracción de bordes puede verse en la Figura 36 en la página 30. Otros ejemplos se muestran en la Figura 37 en la página 31 sobre la imagen segmentada de la Figura 22 en la página 21, en la Figura 38 en la página 31 sobre la imagen segmentada de la Figura 16 en la página 18 y en la Figura 39 sobre la imagen segmentada de la Figura 20 en la página 20.

2.3.6 Vectorización

El proceso previamente descrito es la forma que se revela como más eficiente para dejar una imagen en condiciones de aplicar el proceso de vectorización propiamente dicha. En este proceso, es donde la imagen raster se convierte en un conjunto de vectores gráficos. Algunos algoritmos se han propuesto con anterioridad, de los que las ideas de almacenamiento de los datos (28,32,33) (paquete GEL) se aplican en el algoritmo aquí propuesto, con alguna variante que se explicará más adelante, lo mismo que dicho algoritmo, del que ahora se expone un breve resumen.

El algoritmo propuesto está basado en dos definiciones: el rango y la máscara. El rango de un píxel en la imagen raster se define como el número de píxeles vecinos no nulos en un entorno de conectividad 8. La máscara es un indicador codificado de los vecinos no nulos, que permite llevar a cabo el proceso con

Figura 40. Dominios magnéticos vectorizados

operaciones lógicas en su totalidad, y marca los trayectos previamente utilizados desde cada píxel. Hay situaciones particulares donde el rango ha de ser modificado antes de comenzar la vectorización, en orden

Figura 41. Onda vectorizada

Figura 42. Perro con bordes vectorizados

a evitar estructuras arborescentes y otros problemas que se verán con posterioridad.

El máximo valor posible de rango es 8. Aunque algunos rangos son modificados artificialmente antes de comenzar la vectorización, este sigue siendo el valor máximo. Dado que el rango de cada píxel decrece a medida que avanza el proceso, los primeros píxeles a analizar son los de rango 8. Cada uno de ellos se toma como píxel inicial de una línea, la cual se sigue mientras los píxeles que se encuentren tengan rango

Figura 43. Letras vectorizadas

Figura 44. Superposición de extractor de bordes y vectorización del esqueleto

igual a 2. Si aparece en el recorrido un píxel con otro valor de rango, la línea se da por terminada, porque se trata de un final real de línea (rango = 1), o de una bifurcación (rango > 2). Cada vez que se toca un píxel se rebaja su rango en tantas unidades como entradas y salidas se utilicen de él. Cuando se ha terminado de analizar las líneas comenzadas por píxeles de rango igual a 8, el proceso se repite con los de rangos inferiores, evitando empezar línea por los de rango igual a 2, hasta rango 1. Si el proceso se detiene aquí, aún pueden existir píxeles de rango 2 que corresponden a polígonos cerrados. Estos son analizados aparte. El proceso se detiene cuando todos los píxeles de la imagen tienen rango nulo.

Se aplica durante y después de la vectorización un algoritmo de suavizado basado totalmente en operaciones vectoriales. Se describe este suavizado en capítulos posteriores.

Algunos resultados de la vectorización se pueden ver en los dominios magnéticos vectorizados de la Figura 40 en la página 33, en la onda vectorizada de la Figura 41 en la página 33, en la vectorización de los bordes en la Figura 42 en la página 34, y en la vectorización de las letras previamente adelgazadas en la Figura 43 en la página 34. Una superposición de extracción de bordes en imágenes binarias a la vectorización del esqueleto puede verse en la Figura 44.

2.3.7 Suavizado

La operación de suavizado se realiza en la práctica en el interior de la vectorización. No obstante, a la vista de los resultados, puede ser interesante hacer un nuevo suavizado sin hacer otra vez la vectorización.

Existen algunos precedentes de algoritmos de suavizado, más o menos evolucionados (40). El algoritmo que aquí se propone opera sobre estructuras no uniformes, y está desarrollado con operaciones vectoriales, para así evitar todos los problemas mal condicionados que provoca en los ordenadores el uso de determinantes.

En la explicación subsiguiente, se considerará línea al conjunto ordenado de segmentos rectos, uno a continuación de otro, donde cada uno tiene su vértice final común con el inicial del siguiente.

El procedimiento está basado en avanzar sobre la línea que se va a suavizar, haciendo productos escalares y cruzados. Se definen dos umbrales para suavizado, a saber:

1. Ángulo que permita aproximar dos segmentos por uno recto, o seno del ángulo, o valor de producto cruzado, que es en realidad dicho seno del ángulo.
2. Relación de longitudes entre dos segmentos para poder aproximar por uno solo en la dirección del más largo.

Se forman tres vectores desde cada punto, normalmente origen de un segmento que se desea suavizar con el siguiente, de acuerdo a la Figura 45 en la página 37:

1. Desde el punto dado hasta el siguiente de la línea
2. Desde este último punto hasta el punto donde se desea la aproximación
3. Desde el punto dado hasta el punto donde se desea la nueva aproximación

Se hace un producto escalar entre los vectores 1 y 2. Si este producto es negativo o nulo, el suavizado no sería posible de acuerdo al umbral de producto cruzado, porque tiene lugar una "regresión" de la línea en su segmento siguiente. Sólo se aproximará en caso de que el segundo umbral sea mayor que la relación de módulos de los vectores 2 y 1. Si el producto escalar es positivo, entonces se efectúa un producto cruzado entre los vectores 1 y 3. El suavizado se hace de acuerdo al umbral de producto cruzado, que

Figura 45. Vectores empleados en un suavizado

debe ser mayor que el ángulo que forman los vectores 1 y 3 (o su seno), o de acuerdo al segundo umbral con un criterio de "o" inclusivo.

En el caso de ser la primera operación de suavizado, como ya se ha comentado, se incluye en la vectorización. En este caso, se intenta el suavizado antes de decidir si un segmento existe o puede asociarse con el anterior obtenido en el seguimiento de la línea.

La caracterización de la base de datos formada en el caso de los dominios magnéticos vectorizados y suavizados puede verse en la Figura 46 en la página 38.

2.4 Recuperación de la imagen raster

La conversión de un gráfico vectorial a una imagen raster es un paso bastante más sencillo. Se basa en la aplicación de algoritmos muy similares a los empleados sobre los terminales gráficos para el dibujo de líneas a partir de los vértices. El que se ha llevado a la práctica ha sido el de Bresenham (52), que es capaz de realizar todo a base de sumas y una multiplicación por dos que es resuelta como otra suma. Los grados de conectividad de estas conversiones pueden ser 4 y 8. Se analizan ambas en (51). Para la obtención de la imagen raster, se opera sobre las bases de datos de vectorización, creando una matriz que será posteriormente la imagen raster. Los píxeles de esta imagen se ponen a blanco si son vértices contenidos en dichas bases de datos, y se rellena entre cada pareja de vértices con píxeles blancos siguiendo el algoritmo de Bresenham.

Se puede ver un ejemplo de este proceso en la Figura 48 en la página 39, a partir del gráfico vectorial de la Figura 47 en la página 39, donde se ha hecho el negativo de la imagen raster.

Numero de segmentos.....=	582
Numero de lineas.....=	283
Numero medio de segmentos por linea =	2.05654
Desviacion.....=	0.23096
Numero de troncos.....=	155
Tanto por ciento de troncos.....=	26.63230 %
Numero de ramas.....=	118
Tanto por ciento de ramas.....=	20.27492 %
Longitud media de las lineas.....=	5.85866
Desviacion.....=	5.39020

Figura 46. Base de datos de vectorización de dominios magnéticos

Figura 47. Gráfico inicial para hacer imagen raster

Figura 48. Imagen raster a partir de gráfico

2.5 El algoritmo propuesto de vectorización

2.5.1 Definiciones previas

El algoritmo de vectorización que aquí se propone se basa en la definición de dos parámetros para cada elemento de imagen (píxel), a saber:

1. Rango
2. Entorno

Se define el rango de un elemento de imagen como el número de píxeles de luminancia no nula que tiene a su alrededor, cuyo camino no se haya empleado previamente al pasar vectorizando por dicho elemento de imagen, con conectividad 8. Es decir, que el rango mínimo es 0 y el máximo, 8. El rango de cada píxel es la primera operación a calcular. Para ello se define una imagen de rangos, donde cada elemento tiene el rango del píxel correspondiente en la original. El rango debe decrementarse en una unidad cada vez que se entra o sale del píxel en cuestión durante el proceso de vectorización. Por ello, esta imagen de rangos se llama de rangos actualizados. El rango actualizado es el parámetro más importante, puesto que define la parada del proceso en el momento que todos los píxeles de la imagen tengan rango nulo.

Se define el entorno de un elemento de imagen como una máscara de valor octal que marque la posición de los píxeles de su alrededor, cuyo camino se haya empleado previamente al pasar vectorizando por dicho elemento de imagen, con conectividad 8, de acuerdo con el mismo esquema mostrado en la Figura 26 en la página 23 y que se explica con más detalle en (6). El entorno es la segunda operación a calcular sobre la imagen, creando otra imagen de entornos, donde cada elemento tiene un valor decimal entero igual al valor octal de la máscara de su respectivo píxel en la imagen original.

Para el correcto funcionamiento del proceso, es necesario definir aún dos imágenes auxiliares más. La primera contendrá los rangos iniciales de cada elemento al principio de la vectorización, y se llamará de rangos iniciales. La segunda contendrá los rangos iniciales incluyendo las modificaciones propias de los casos particulares que se especificarán. Se llamará esta imagen de rangos modificados. Así, la imagen de rangos iniciales calculada al principio permanece inalterable en el proceso, salvo algunos cambios comunes con la de rangos actualizados y que se realizarán antes de vectorizar, correspondientes a casos particulares. Estos cambios artificiales serán distintos a los que ante las mismas circunstancias se realizarán antes de

vectorizar sobre la imagen de rangos modificados. La imagen de entornos que contiene las máscaras se irá actualizando a medida que el proceso de vectorización avance.

2.5.2 El núcleo del algoritmo

La vectorización comienza uniendo con líneas quebradas los píxeles de luminancia no nula, tomando como píxel inicial de estas líneas los que sean de rango máximo, igual a 8. A pesar de que el rango ha podido ser aumentado artificialmente ante las situaciones particulares que se describirán con posterioridad, nunca se da la situación en que sea mayor que 8. Sobre cada píxel de la línea iniciada con el de rango 8 y mientras se efectúa su seguimiento, se inicia una búsqueda en abanico, con conectividad 8, hasta encontrar una salida. Todos los píxeles de la línea deben tener rango actual, inicial y modificado, es decir, los valores de las tres imágenes de rangos correspondientes al píxel dado, iguales a 2. Si esto no es así, se interpreta un final de línea. Esto se debe a que si se quiere un proceso de vectorización que sea fiel a la imagen adelgazada, debe apoyarse en los nudos de la misma. Si el rango inicial de un píxel no es 2, pertenece a un nudo (rango > 2) o a un final de línea (en caso de ser igual a 1). Si no lo es el actualizado, los caminos alrededor de ese píxel ya han sido utilizados, con lo cual se debe crear un final de línea artificial, o también se trata de un nudo. Si no lo es el rango modificado, se puede estar ante una situación particular que exija hacer un final de línea sin existir nudo ni final verdadero. Esto último se describirá en los casos particulares.

Al acabar esta operación con los píxeles de rango 8, ya no existe ningún píxel con este rango actualizado. Esto se debe a que cada vez que se toca un píxel en la vectorización, su rango actualizado se hace disminuir en tantas unidades como entradas y salidas del píxel se utilicen. El rango actualizado máximo posible es, pues, 7.

La actualización del rango y la máscara tiene lugar simultáneamente sobre los dos píxeles que se unen en cada momento. En el caso del rango basta restar una unidad. En el caso de la máscara se definen dos máscaras nuevas correspondientes a la unión, tomando como píxel central cada uno de los dos en cuestión, siendo estas máscaras nulas excepto en el cuadrado correspondiente al píxel vecino unido. La actualización se realiza haciendo un "o" inclusivo entre la máscara vieja y dicha máscara para cada uno de los dos píxeles. Un ejemplo se ve en la Figura 49 en la página 42.

En este momento se toman todos los píxeles de rango actualizado igual a 7 como comienzo de línea, siguiendo un procedimiento exactamente igual al de antes con los de rango 8. Se repite la operación tomando como píxeles iniciales de línea los de valores entre 6 y 1, en orden decreciente y con la excepción del 2 para el rango actualizado.

Figura 49. Esquema de actualización de máscaras

Se debe tomar una precaución especial cuando se trata con líneas que comiencen por píxeles de rango actualizado igual a 2. En este caso, se comprueba que el rango inicial no es 2, en cuyo caso se toma dicho píxel como comienzo de línea. Se trata entonces de algún nudo de rango superior que se ha ido reduciendo durante la vectorización, pudiendo arrancar perfectamente una línea desde él. Si la precaución no se toma, cualquier píxel en el interior de una línea tiene todos sus rangos iguales a 2 y podría ser candidato a iniciar una línea. Esta línea quedaría así partida sin necesidad en al menos dos trozos, lo que imposibilitaría un suavizado y haría mayor inútilmente la base de datos de vectorización, ya que cada línea nueva lleva un registro de protocolo tal y como se describió. En el peor de los casos, una línea puede quedar partida en muchos trozos. Veamos un ejemplo. Supóngase que se toma un píxel central de una línea como comienzo. Si el píxel siguiente es un nudo o un final, la línea se debe dar por terminada de acuerdo con lo expuesto. Un poco más adelante, el píxel anterior al que se dio de rango 2 en la misma línea puede tomarse como principio de línea. Su vecino de rango 2 ya no tiene tal rango actualizado, puesto que se utilizó su salida para unirse al que era nudo o final. Así, su rango actualizado es 1, y debe tomarse como fin de línea, creándose otra línea muy corta de un sólo segmento, que en realidad debiera estar unido al anterior. El proceso se multiplica en cadena hasta que se termine la línea.

Al final, aún quedan píxeles con rango actualizado no nulo en la imagen. La precaución seguida con los píxeles de rango inicial igual a 2 hace que estructuras poligonales cerradas y aisladas donde todos los

píxeles tienen dicho rango de ese valor hayan sido ignoradas. Así, se debe iniciar de nuevo un proceso que tome píxeles de rango actualizado igual a 2 como píxeles iniciales de línea, sin tener ahora la precaución de antes. En este momento dicha precaución ya no hace falta, y las poligonales cerradas salen como líneas enteras y de una sola vez. Esto se debe a la ausencia total de píxeles en la imagen con rango distinto de 2, por lo que ninguno se puede tomar como fin de línea. El fin de la poligonal cerrada y aislada tiene lugar cuando se regresa a su primer píxel, que en este momento tiene rango actualizado igual a 1 porque se utilizó su salida para unirse con los siguientes, que, al estar en cadena, utilizan su entrada y su salida únicas, quedando con rango cero, dos unidades menor que el inicial.

Antes de proceder a la unión de dos píxeles mediante un segmento, una vez determinado que hay que unirlos, se intenta un suavizado con los segmentos anteriores de la línea que se está formando. Esta operación se hace de acuerdo a dos umbrales y se detalla en otro capítulo dedicado al suavizado. Si es posible suavizar, se sustituye el último segmento de la línea y el nuevo que habría que trazar por uno único desde el punto inicial del anterior hasta el punto final del nuevo. Esta operación es de capital importancia, pues es la única forma de no introducir segmentos cortos y totalmente alineados como componentes de una línea que en realidad no es quebrada, sino recta, quedando de esta forma con el mínimo número de segmentos necesarios.

El proceso de vectorización se detiene cuando todos los píxeles de la imagen tienen un rango actualizado igual a cero.

2.5.3 Situaciones particulares

Si la vectorización se realiza tal y como se ha expuesto, aparecen problemas en situaciones particulares. En la Figura 50 en la página 44 se pueden ver diversas estructuras arborescentes que no deben aparecer así, junto a su solución correcta. En la Figura 51 en la página 44 se puede ver otro ejemplo con error junto a su solución correcta. Lo mismo en la Figura 52 en la página 45. Estos dos últimos casos se pueden resolver de diversas formas, pero se ha optado por utilizar una manera que resuelva los dos al mismo tiempo.

El tratamiento de los casos particulares tiene lugar como previo a la vectorización, una vez calculados los tres rangos que en principio son iguales. Los cambios consisten en engañar al proceso, haciéndole ver que ante una determinada situación real tiene otra muy distinta. Para ello se opera como sigue:

1. Se crea un nuevo entorno de máscaras que se pone a cero inicialmente, y para todos los píxeles de la imagen se opera como se describe a continuación. La máscara utilizada para comenzar la

Figura 50. Caso particular de estructuras arborescentes

vectorización no será nula, tal como sería teóricamente, sino que será la máscara nueva surgida de este tratamiento previo.

2. Se define el conjunto de máscaras de la Figura 53 en la página 45.
3. Se define el conjunto de máscaras de la Figura 54 en la página 46.
4. Se define el conjunto de máscaras de la Figura 55 en la página 46.
5. Se define el conjunto de máscaras de la Figura 56 en la página 47.
6. Para cada píxel, se hace un "y" lógico entre su máscara entorno en la imagen adelgazada y las máscaras de la Figura 53 en la página 45.

Figura 51. Caso particular de píxel envuelto en nudo

Figura 52. Caso particular de nudo con número par de píxeles

7. Si el resultado es igual a la máscara test, significa que la máscara del píxel tiene 1's al menos en las posiciones de la máscara test. Los rangos inicial y actualizado se aumentan artificialmente en el número de máscaras de la Figura 53 que dan dicho resultado. El rango modificado permanece inalterable. Esto resolverá posteriormente las situaciones de la Figura 51 en la página 44 y la Figura 52. Si el rango no se aumentase, aparecerían errores posteriores, debido a que los test que se hacen después rebajan artificialmente el rango. Los píxeles que se encuentren en dicha situación, verifican varios test posteriores, y se quedarían sin rango.
8. Para cada píxel, se hace un "y" lógico entre su máscara entorno en la imagen adelgazada y las máscaras

Figura 53. Primer conjunto de máscaras para casos particulares

Figura 54. Segundo conjunto de máscaras para casos particulares

ras de la Figura 55 en la página 46.

9. Cada vez que el resultado es igual a la máscara test, puede aparecer alguna estructura arborescente elemental. Se hace otro "y" lógico entre la misma máscara entorno del píxel en la imagen adelgazada y la máscara de la Figura 54 correspondiente (mismo lugar, pues los conjuntos de máscaras están ordenados) a la máscara de la Figura 55 que daba el mencionado resultado.
 10. Si el resultado de este último "y" lógico no es igual a la máscara test, se tiene la estructura arborescente similar a la de la Figura 50 en la página 44. Entonces se reducen artificialmente los tres rangos en una unidad, y el entorno de vectorización sufre un "o" lógico con la máscara de la
-

Figura 55. Tercer conjunto de máscaras para casos particulares

Figura 56. Cuarto conjunto de máscaras para casos particulares

Figura 56 en la página 47 correspondiente a la máscara de la Figura 55 que daba antes el mencionado resultado. Con esto, se incorpora a la máscara nueva el "1" en la posición diagonal. Este resultado dirá al proceso que la salida del píxel que provocaba la estructura arborescente ya ha sido utilizada, cuando en realidad no se dibujará la línea.

11. Si el resultado del último "y" lógico es la máscara test, los tres rangos se reducen artificialmente una unidad también, y el entorno sufre también un "o" lógico, pero con la máscara correspondiente del mismo orden de la Figura 54 en la página 46. En este caso se trata de una estructura como la de la Figura 51 en la página 44 o la de la Figura 52 en la página 45. La máscara incorpora 1's en la horizontal o la vertical, con lo que el proceso entenderá con posterioridad que esas salidas ya han sido

Figura 57. Suavizado erróneo sin final de línea artificial

utilizadas, cuando realmente no se dibujará ningún segmento por ellas. Los nudos se resuelven, pues, por las diagonales, tal como indican las mismas Figura 51 en la página 44 y Figura 52 en la página 45. Aquí se ve la necesidad del aumento de rango artificial anterior. Para ver un ejemplo, se parte de una situación como la de la Figura 51 en la página 44. Esta situación parte de un rango igual a 5 y verifica 4 test de estos últimos, retirando dos de ellos el mismo camino por el píxel de la derecha. Quedaría, pues, un rango igual a 1 y con dos posibles salidas por las diagonales, ya que las horizontales y verticales se prohíben. Dado que el entorno es incompatible con el rango, se hace necesario aumentar artificialmente el rango en este caso y similares, tal como se realizó con anterioridad. Es posible que píxeles en la situación de la Figura 51 en la página 44 y de la Figura 52 en la página 45 queden con algún rango igual a 2. De ahí la necesidad de mantener rangos iniciales que no se modifiquen en estas situaciones, ya que permanecieron inalterables cuando hubo aumentos. Son los rangos modificados. Al mantener algún rango distinto de 2, se hace un final de línea artificial. Así, este vértice sirve de punto de apoyo en la operación de suavizado. Si no existiera fin de línea artificial, un suavizado podría dar resultados erróneos, tales como el de la Figura 57 en la página 47.

La contemplación de los casos particulares a base de falsear el rango y el entorno de cada píxel se muestra como la solución óptima que evita el tratamiento particular a posteriori, siendo sustituido por un tratamiento global a priori, exento de errores, y que proporciona una rapidez de evaluación de las mencionadas situaciones particulares que no es común en la mayoría de los algoritmos de vectorización existentes.

2.5.4 Formatos de almacenamiento

Como ya se ha explicado, el objeto de la vectorización es codificar los esqueletos o los bordes de las estructuras de la imagen con vectores almacenados en dos bases de datos: El Universo y los Punteros. La primera contiene todos los vértices de los vectores, y, la segunda, un conjunto de punteros que señalan los elementos del Universo formando líneas sobre la imagen con los vectores definidos por los vértices apuntados.

El Universo es una base de datos compacta, en el sentido de que lo único importante es el orden de sus elementos para que el puntero de la segunda base de datos sea capaz de señalarlos. Su formato no merece especial atención.

En la base de Punteros existe una lista de éstos para cada línea de la imagen, con un formato adecuado y versátil. Éste consiste en la utilización de dos registros para cada estructura. El primero contiene información sobre el número de punteros pertenecientes a la lista de la estructura de que se trate, y algunas

informaciones adicionales tales como si la línea es una rama, tronco, etc..., su longitud, sus características gráficas de posterior representación, etc..., todas ellas codificadas. El segundo registro contiene la lista de punteros de la estructura.

El formato de almacenamiento descrito puede encontrarse con más detalle en los trabajos de Navalón y Jiménez (28,32,33,35) sobre el uso del paquete GEL (Lenguaje Geométrico), que es el nombre de este formato.

En la mayoría de las aplicaciones se puede emplear un formato alternativo. Consiste en la creación de una sola base de datos, llamada de Coordenadas, que sustituye al Universo y los Punteros. Esta nueva base, de estructura igual a la de Punteros, contendrá en el lugar de éstos las coordenadas de los puntos directamente. Así, el Universo deja de ser necesario. La información puede aparecer duplicada en algunos casos con este nuevo formato, dado que los puntos de corte de las líneas vectorizadas en el gráfico se repetirán. No obstante, la probabilidad de existencia de un punto de cruce es muy baja frente al número total de puntos, con lo cual dicha duplicidad es absolutamente mínima.

En una transmisión, con el primer formato, habría que enviar primero el Universo y luego los Punteros. Con el segundo, sólo la base de Coordenadas. La cantidad de información transmitida es ligeramente superior en este segundo caso, pero el incremento es tan pequeño, que queda compensado con creces por la paginación necesaria para moverse sobre el Universo con los Punteros en el momento que el número de puntos sea suficientemente grande. Para compensar en parte esta paginación habría que ordenar previamente el Universo, con unos criterios no demasiado claros, lo que hace algo menos eficiente su utilización.

2.5.5 Codificación final de las estructuras

La base de datos final de vectores contiene en el registro de protocolo de cada línea informaciones acerca de la misma, tal como ya se describió (paquete GEL). Además del número de segmentos rectos que la componen, y el tipo de línea deseado para su representación, color, etc..., se incorpora aquí otro tipo de informaciones, tales como si se trata de un tronco, una rama, línea aislada, su longitud, etc... Una línea es una rama si el rango inicial del píxel situado en uno de sus extremos es igual a 1, mientras que el rango del píxel del otro extremo es mayor que 2. Una línea es un tronco si el rango inicial de cualquiera de los píxeles de sus extremos es mayor que 2. Una línea es aislada si el rango inicial de cualquiera de sus extremos es igual a 1. Consideraciones especiales deben hacerse con las líneas cerradas, donde los dos extremos tienen rango inicial igual a 2, siendo también aisladas.

2.6 Medida de distancias entre líneas de una imagen en entornos no conexos

Este tema es una variante que se puede aplicar sobre imágenes vectorizadas y suavizadas. En el capítulo de aplicaciones se describe alguna de ellas (57,58). El proceso se describe a continuación. Algunas ideas anteriores, aunque para aplicaciones muy diferentes, y, por tanto, con soluciones distintas, pueden verse en (53,55). Estas ideas anteriores no son aplicables porque las líneas de las imágenes a tratar en el caso que aquí se contempla están en entornos fuertemente no conexos. No obstante, son de interés las exposiciones sobre el cálculo de distancias hechas en (54).

Se ha desarrollado un algoritmo de búsqueda de las parejas de vectores más próximas en una imagen, para construir un mapa de distancias basado en las distancias calculadas entre los vectores. El método se ha desarrollado en varios pasos, a partir de una imagen vectorizada y suavizada. La base de punteros del paquete GEL se reordena para simplificar el proceso. Se ha creado un algoritmo de búsqueda de las parejas de vectores más próximos que resuelve todos los problemas que se presentan en el tratamiento de los vértices de los vectores por medio de técnicas de gráficos por ordenador. A continuación se realiza una operación de poligonalización, que consiste en la formación de polígonos entre las parejas de vectores localizados como más próximos.

Se define una imagen auxiliar sobre la imagen inicial vectorizada. Los polígonos se superponen a dicha imagen auxiliar, y se rellena sobre ella el área encerrada por los polígonos. El valor para rellenar polígonos se elige de acuerdo al criterio de mostrar información de distancias entre los diferentes trozos de todos los segmentos más próximos emparejados que circunden las áreas encerradas. De esta forma, la imagen auxiliar rellena es el mapa de distancias deseado. Los problemas que aparecen normalmente en operaciones de gráficos por ordenador en los vértices de los vectores cuando se forman y rellenan los polígonos son objeto de tratamientos especiales.

Los pasos previamente descritos son el centro del algoritmo. Al final se hace una operación de filtrado sobre el mapa de distancias para suavizar los resultados. Es necesario evitar las discontinuidades creadas por la construcción discreta de los polígonos rellenos.

La información final del mapa de distancias se puede presentar en muy diversas formas: mapas de grises, mapas gráficos 3-D, curvas de nivel, mapas de colores y funciones estadísticas.

2.6.1 Definiciones previas

El punto de partida lo constituye una imagen vectorizada. Cada línea de la base de datos se divide en sus segmentos rectos. Antes de iniciar ningún proceso de búsqueda, conviene orientar los segmentos que resultan de partir las líneas quebradas que los contienen. Se adopta el criterio de avance positivo de un segmento aquél que va de izquierda a derecha, eligiendo para los segmentos verticales el sentido de abajo a arriba. El lado positivo de un segmento se define como su lado izquierdo en el sentido de avance positivo. Estas definiciones se pueden ver de forma gráfica en la Figura 58 en la página 52.

Una vez que la base de datos está ordenada, la búsqueda puede comenzar. Se analiza cada uno de los segmentos, basando este análisis en un puntero que comienza señalando a su punto inicial y salta posteriormente a lo largo de él. El segmento analizado en un instante dado se llamará "segmento maestro", y la distancia de salto del puntero sobre él, "umbral de camino". Los puntos del segmento maestro señalados por el puntero se llamarán "puntos base".

2.6.2 Búsqueda del vector más cercano

Se realiza una búsqueda a izquierda y derecha desde el segmento maestro en cada uno de sus puntos base. Se traza por cada uno de dichos puntos una línea virtual, ortogonal al maestro en un primer paso, y en forma de abanico a continuación. Esta línea virtual se llamará "línea de búsqueda". Una ilustración sobre esto puede verse en la Figura 59 en la página 53.

2.6.2.1 Búsqueda ortogonal

Se detectan los segmentos que cortan a las líneas de búsqueda trazadas ortogonalmente al segmento maestro, y se calculan los puntos de corte. Estos puntos de corte son llamados "puntos imagen" del correspondiente punto base. Se calculan también las distancias entre cada punto base y su respectivo punto imagen. Después de analizar todos los segmentos de la base de datos, los dos más cercanos a cada lado del maestro se llamarán "segmentos esclavos" correspondientes al maestro en el punto base de que se trate. El procedimiento de corte se ha realizado en coordenadas paramétricas, de acuerdo con las expresiones que se encuentran en capítulos posteriores dedicados a ecuaciones vectoriales.

Si la imagen se procesa solamente con esta búsqueda ortogonal, los resultados del mapa de distancias en los vértices de los vectores se separarán bastante del resultado deseado. El esclavo de un maestro en las proximidades de un vértice es, normalmente, el segmento siguiente perteneciente a la misma línea, y no puede ser encontrado con búsqueda ortogonal. Se impone una búsqueda en abanico que incluya la orto-

Figura 58. Orientación de segmentos de línea

gonal.

2.6.2.2 Búsqueda en abanico

Se realiza una búsqueda en abanico en vez de la búsqueda ortogonal. Se hace desde cada punto base trazando líneas de búsqueda entre 0 y 360 grados. Por supuesto, sólo se trazan en realidad algunas de estas líneas. La separación entre dos líneas consecutivas de búsqueda en abanico se llama "umbral de abanico".

Los vectores de dirección de las líneas de búsqueda desde el maestro se hallan como sigue. El vector director del maestro se puede escribir en la forma:

$$\vec{v}_m = x_m \hat{x} + y_m \hat{y} \quad (2.1)$$

La búsqueda en abanico se hace sobre un ángulo dado θ respecto al sentido de avance positivo del maestro y sobre su cara positiva. Las componentes del vector de dirección de las líneas de búsqueda en abanico se pueden hallar, siguiendo la ecuación (2.1), de acuerdo a las siguientes expresiones:

Figura 59. Esquema general de segmentos maestro y esclavos

$$\begin{aligned} v_{bx} &= x_m \cos(\theta) - y_m \sin(\theta) \\ v_{by} &= y_m \cos(\theta) + x_m \sin(\theta) \end{aligned} \quad (2.2)$$

Estos vectores sólo se calculan para el primer cuadrante, pues para los restantes la búsqueda se hace con las prolongaciones de las líneas de búsqueda que definen (tercer cuadrante), o con el suplemento del ángulo y su respectiva prolongación (cuadrantes segundo y cuarto).

Se muestra un esquema de la búsqueda en abanico junto con la ortogonal en la Figura 60 en la pági-

2.6.2.3 Situaciones especiales de búsqueda

Otra clase de problemas tiene lugar al analizar los puntos extremos de una línea maestra. La respuesta del sistema en estos casos dice que los esclavos son los segmentos que tienen dichos puntos comunes con el maestro, es decir, son los pertenecientes a la misma quebrada. Esto sólo es cierto si las líneas de búsqueda son casi paralelas al maestro. En otros casos, la solución consiste en detectar que la línea de búsqueda tiene longitud nula, y continuar buscando los esclavos correctos. Un esquema de esta situación se puede ver en la Figura 61 en la página 56.

2.6.3 Poligonalización

2.6.3.1 Generación de polígonos

Se puede formar polígonos de cuatro lados entre las líneas perpendiculares de búsqueda trazadas por dos puntos base consecutivos del maestro y sus respectivos puntos imagen. Los dos lados que faltan del polígono son los tramos de maestro y esclavo que quedan entre las dos líneas de búsqueda. La longitud de estas líneas de búsqueda entre el segmento maestro y el segmento esclavo estima la distancia entre ellos. Si el polígono es rellenado con una intensidad de gris proporcional al promedio de longitudes de las dos líneas de búsqueda que delimitan el polígono, el área encerrada lleva la información de la distancia entre las líneas más próximas de la zona. De esta forma se genera el mapa de distancias que se desea. Un polígono típico puede verse en la Figura 62 en la página 57.

Hay casos en los que, si no se toman las debidas precauciones, se podrían formar polígonos entre tres líneas, a saber, un maestro y dos esclavos. Esto es incorrecto, pues, dado que el rellenado de los polígonos debe mostrar información de acuerdo a la distancia entre dos segmentos, es absurdo introducir tres, caso en el que el rellenado no tendría sentido. Esta situación se presenta cuando al avanzar un "umbral de camino" sobre el maestro, el esclavo cambia. La solución es detectar este cambio de esclavo, formando y rellenando el polígono normalmente si no hay tal cambio, y formando dos polígonos distintos, tras una búsqueda adicional, uno con cada esclavo, en caso contrario. Esta situación se muestra en la Figura 63 en la página 58, donde P_1 es el polígono incorrecto y P_2 el correcto.

Esta parte del proceso es la que lleva mayor tiempo de cálculo. El modelo elaborado para la formación de los dos polígonos está basado en la solución de diez casos posibles de posiciones relativas del maestro y sus dos esclavos, en combinación con el "umbral de camino". No obstante, se buscó un algoritmo que resuelve todos los casos al tiempo, con lo cual se asegura que otras posibles situaciones similares quedarán

Figura 60. Búsqueda ortogonal y en abanico de líneas cercanas

resueltas. Estos casos se pueden ver en la Figura 64 en la página 59. El caso de segmentos que se corten se resuelve de forma particular y la situación se muestra en la Figura 65 en la página 60, y resulta un polígono degenerado en tres lados.

El algoritmo elaborado para estas situaciones se describe como sigue. Se trazan nuevas líneas de búsqueda paralelas a aquéllas que se estén utilizando en ese momento, pero desde los vértices de los esclavos en litigio, calculando sus puntos de corte con el maestro. De estos últimos, los que se salgan del polígono antiguo incorrecto se eliminan directamente, y, de los que quedan, se toma aquél que implique menor avance sobre el maestro, y se utiliza como nuevo punto base. Se busca su punto imagen, y se puede así

Figura 61. Búsqueda errónea en los vértices

formar el nuevo polígono correcto. El algoritmo se ve de una forma gráfica en la Figura 63 en la página 58.

Por todo lo expuesto, el "umbral de camino" no es necesario que sea exageradamente pequeño, pues cada vez que se cambia de esclavo al avanzar sobre el maestro, se realiza una búsqueda para formar los dos polígonos como ya se ha explicado.

2.6.3.2 Rellenado de polígonos

Varios métodos generales de relleno de polígonos se describen en la bibliografía (56). Sin embargo, las condiciones particulares del proceso descrito admiten la posibilidad de un algoritmo más simple, al hacer que todos los polígonos a rellenar sean convexos. El relleno se realiza por medio de un rayado vertical con líneas adyacentes. El nivel de gris con que se rellena el interior del polígono es uniforme y proporcional a la distancia calculada entre el maestro y el esclavo. El blanco corresponde a la mínima distancia calculada sobre la imagen, y el negro a la máxima. La asignación de grises se hace linealmente mediante una tabla de consulta de vídeo (VLT, Video Look-up Table). Dado que los polígonos son convexos, los cortes con las rayas verticales de relleno pueden ser simples o dobles, pero nunca de orden mayor, lo cual facilita y simplifica el algoritmo. Los cortes dobles se detectan y se almacenan como simples. Un esquema de los cortes al rellenar por verticales se puede ver en la Figura 66 en la página 61.

Figura 62. Polígono típico de relleno

2.6.3.3 Intersección de polígonos

El proceso de búsqueda en abanico seguido de la poligonalización y el relleno, provoca situaciones de intersección entre polígonos, que se concretan en dos casos. El primero se debe a la búsqueda en abanico, y se muestra en la Figura 67 en la página 62. El segundo se debe al solapado de búsquedas desde dos maestros diferentes y se muestra en la Figura 68 en la página 62. Las áreas de intersección deben representar la mínima distancia entre los segmentos que hay en las proximidades, y, por tanto, son rellenos con el máximo nivel de gris entre los polígonos que intersectan.

2.6.4 Proceso posterior

La imagen resultante es el mapa de distancias. Este mapa aparece como un mosaico con ruido en las esquinas de los polígonos, producido por el sistema de relleno discreto. Este problema se soluciona con un filtrado que elimina los ruidos y suaviza los saltos entre los polígonos. Se usa un filtro de media, dada su simplicidad.

Figura 63. Caso de cambio de esclavo al recorrer el maestro

Figura 64. Casos posibles de cambio de un esclavo sobre el mismo maestro

Figura 65. Poligonalización en el caso particular de corte de segmentos

2.6.5 Presentación

La forma más común de presentación del mapa de distancias es en forma de mapa de grises, correspondiendo los tonos claros a distancias cortas y los oscuros a distancias más largas.

Un mapa de distancias en grises se puede ver en la Figura 69 en la página 63, sobre un original vectorizado en la Figura 70 en la página 63. En la Figura 71 en la página 64 se pueden ver ambas superpuestas.

Los mapas codificados en colores pueden ser interesantes en alguna circunstancia, como presentación alternativa al de grises. El color rojo se usa para las distancias más cortas y el azul para las más largas. Las distancias intermedias pasan por los tintes cian, verde y amarillo, siguiendo el Spectrum Locus. La reproducción práctica de estos colores tiene lugar según el triángulo de Maxwell.

Otro tipo de presentación puede realizarse mediante gráficos 3-D y sus curvas de nivel asociadas, donde los ejes x-y son las dimensiones de la imagen y el eje z corresponde al nivel de luminancia del mapa de distancias. Utilizando estos niveles de luminancia también se puede presentar un histograma de barras, que da información estadística. Se explican algoritmos desarrollados aquí para los gráficos 3-D y sus cur-

Figura 66. Posibles cortes al rellenar por verticales un polígono de cuatro lados

vas de nivel. Un esquema general de este sistema puede verse en la Figura 72 en la página 65.

Figura 67. Intersección de polígonos debida a la búsqueda en abanico

Figura 68. Intersección de polígonos debida a la búsqueda desde distintos maestros

Figura 69. Mapa de distancias entre líneas

Figura 70. Imagen vectorizada para medir distancias entre líneas

Figura 71. Mapa de distancias sobre imagen vectorizada

Figura 72. Esquema general de la medida de distancias entre las líneas de una imagen

3.0 Tratamiento de estructuras 3-D

3.1 Sistema propuesto de representación 3-D

3.1.1 Necesidad del sistema propuesto y descripción general

Se presenta un sistema completo basado en técnicas de gráficos por ordenador para la representación 3-D de curvas y superficies no uniformes, destacando el hecho del uso exclusivo de coordenadas paramétricas. Dada la gran ventaja del uso de este tipo de coordenadas en los ordenadores, como luego se verá, existen algoritmos capaces de encontrar óptimamente las coordenadas paramétricas de estructuras formuladas implícitamente (22). Se discutirán más adelante algunas ideas acerca del tratamiento y almacenamiento de los resultados, presentándose algunos de estos últimos.

Todos los problemas relacionados con penetraciones de curvas y superficies multiformes son resueltos con pleno éxito. Se emplea un único tratamiento para cada elemento del dibujo 3-D, sea curva o superficie, sin apenas distinción entre ellos. Se han diseñado algoritmos especiales para el tratamiento de líneas ocultas que se incluye en este sistema de representación.

La mayoría de los algoritmos actuales son capaces de resolver con mayor o menor eficiencia el problema de dibujar curvas y superficies uniformes, es decir, cuando sólo existe una coordenada z para cada pareja de coordenadas x, y , y en muchas circunstancias presentan muchos errores cuando las estructuras tienen vértices agudos. Sin embargo, ésta no es la situación habitual cuando las curvas y superficies se formulan en coordenadas paramétricas ya que en este caso no es posible representar una superficie por una matriz bidimensional $z = z(x, y)$. En este caso muchos algoritmos actuales aproximan las superficies por modelos de líneas cuyo aspecto deja bastante que desear, cometiendo muchos errores, y no resolviendo ningún tipo de penetraciones o autopenetraciones que son intrínsecas a las superficies multiformes.

Una idea podría ser hacer tres matrices independientes $z = z(u, v)$, $y = y(u, v)$ y $x = x(u, v)$, para cada superficie implicada en el dibujo, siendo u y v dos parámetros, dado que ésta es la forma de definir una superficie en coordenadas paramétricas, y utilizar un algoritmo convencional de líneas ocultas para cada

matriz por separado. Si esto se lleva a la práctica, la ocupación de memoria sería excesiva cuando el número de superficies a dibujar sea grande. Además, el cálculo de las partes ocultas en las superficies es más difícil utilizando métodos convencionales, porque habría tres matrices-superficie para cada superficie real. Por otra parte, un tratamiento diferente se haría necesario para dibujar curvas en el mismo gráfico que las superficies, detectando penetraciones y líneas ocultas. En el algoritmo propuesto se crean dos bases de datos a partir de las líneas y superficies. La primera base de datos contiene todas las curvas del dibujo, incluyendo las nuevas líneas definidas por las intersecciones de superficies. Para ello se ha diseñado otro algoritmo capaz de resolver las penetraciones entre superficies. Las superficies se dibujan como rejillas de curvas, siendo almacenadas también estas curvas en esta base de datos. La segunda base de datos contiene las coordenadas de cada parche de cuatro lados de la rejilla.

A continuación se realiza una operación de perspectiva sobre un universo o conjunto de puntos que contiene las coordenadas de todos los puntos que definen las curvas en la primera base de datos. El algoritmo de líneas ocultas diseñado especialmente para este proceso se aplica en este punto sobre todas las líneas de la primera base de datos, al cruzarla contra la segunda base de datos, que contiene los parches. Este algoritmo es capaz de detectar las penetraciones de las líneas contra los parches, resolviendo todos los problemas que se presenten (60).

3.1.2 Definición de primitivas

3.1.2.1 Definición de las estructuras básicas

Se define un conjunto de funciones primitivas de acuerdo con los requerimientos del dibujo deseado. Estas funciones pueden ser superficies y curvas, formuladas en cualesquiera de sus posibles expresiones en coordenadas paramétricas.

Una curva es una estructura con un único grado de libertad. Depende, pues de un sólo parámetro. Una superficie es una estructura con dos grados de libertad. Depende, pues, de dos parámetros. Las expresiones generalizadas de curvas y superficies se muestran en la ecuación (3.1) y en la expresión (3.2).

$$\begin{aligned}x &= x(\lambda, \beta) \\y &= y(\lambda, \beta) \\z &= z(\lambda, \beta)\end{aligned}\tag{3.1}$$

$$\begin{aligned}x &= x(t) \\y &= y(t) \\z &= z(t)\end{aligned}\tag{3.2}$$

3.1.2.2 Entradas al proceso

Una vez definida la necesidad de tener primitivas, la pregunta que puede surgir es cuál debe ser su formato. Es lógico pensar que cualquier función matemática expresada en paramétricas es válida. No obstante, existen funciones y superficies que tienen necesariamente que ser dadas por tablas numéricas. De esta forma, el sistema ha de prever la posibilidad de entrada por tablas almacenadas en ficheros. Este es, por ejemplo, el caso del levantamiento 3-D de la transformada de Fourier de una imagen, o de la imagen misma. En el caso de superficies, la entrada debe ser de uno, dos o tres ficheros matriciales bidimensionales, correspondiendo cada dimensión a cada uno de los dos parámetros propios de una superficie.

Hay algunas veces que las estructuras vienen en función de un número reducido de puntos, y es necesario interpolarlas en orden a suavizar su aspecto, y representar con menor error. Existen diversos métodos de interpolación de curvas y superficies, y se pueden ver en (70,71,72,73,74).

3.1.3 Creación de una base de datos de líneas

3.1.3.1 Almacenamiento de curvas

Las curvas son evaluadas en el rango deseado de valores de los parámetros. Los resultados son almacenados en una base de datos con un formato muy similar al propuesto por Jiménez y Navalón en (28,32,33), ocupando cada línea dos registros en el fichero. El primero contiene información acerca del número de puntos de la curva, el estilo de línea (punteada, continua, etc), el color, etc.. El segundo registro contiene una lista ordenada de las coordenadas 3-D de los puntos de la curva. Este formato es muy flexible en el sentido de que no existe necesidad alguna de que las líneas tengan el mismo número de puntos.

Antes de su almacenamiento, cada curva primitiva se corrige en rotación, escalado 3-D independiente en cada coordenada, y translación, según se desee para ir formando el dibujo. Estas operaciones se realizan simultáneamente al cálculo de los puntos.

3.1.3.2 Almacenamiento de superficies

Las superficies se almacenan como dos conjuntos de curvas. Cada curva de un conjunto dado se calcula dando valores diferentes a uno de los parámetros, mientras el otro permanece constante. El aspecto final de las superficies es el de una rejilla.

El formato de almacenamiento es el mismo utilizado para las curvas, que ya ha sido descrito. Antes del almacenamiento, se han de realizar transformaciones de rotación escalado 3-D independiente en cada dimensión, y translación, similarmente a lo descrito para las líneas, al objeto de ir componiendo el dibujo.

3.1.4 Creación de una base de datos de superficies

Hasta ahora no se han hecho diferencias sensibles entre el almacenamiento de curvas y superficies. Si se pretende la utilización de un algoritmo de líneas ocultas con posterioridad, es necesario tener esta base de datos que determina qué estructuras son opacas y pueden ocultar a otras.

La representación de superficies como rejillas de curvas sugiere la representación de las estructuras opacas como conjuntos de parches opacos de cuatro lados. Así, se crea una segunda base de datos para almacenar estos parches, donde cada registro contiene una lista ordenada de las coordenadas 3-D de los vértices de cada parche. Esta base de datos puede crearse en el momento de cálculo de la superficie como mejor opción. En caso de no ser posible por alguna circunstancia, existen técnicas de creación de parches sobre superficies previamente definidas (63).

Antes del almacenamiento, debe aplicarse a los vértices de los parches las mismas transformaciones de rotación, escalado 3-D independiente en cada dimensión y translación, que se aplican a las curvas de la superficie de la que proceden, ya que las dos formas de almacenamiento representan las mismas estructuras. En la práctica, las dos bases de datos se calculan simultáneamente, de forma que no es necesario que el sistema recuerde los parámetros empleados en las transformaciones utilizadas al crear la base de datos de líneas.

3.1.5 Intersección de superficies

La situación más normal para una superficie dada en coordenadas paramétricas es que sea multiforme. En este caso, algunas líneas nuevas no calculadas como líneas de la rejilla deben ser dibujadas e incluidas como producto de las autopenetraciones de la superficie. El problema es idéntico cuando aparecen varias superficies en el dibujo. Las penetraciones entre ellas y las autopenetraciones de cada una deben ser calculadas.

Se han diseñado en la bibliografía muchos algoritmos de corte de superficies (64,65,66,67,68,69). Sin embargo, aquí se propone un algoritmo basado en las mismas ideas de cortes entre superficies que permiten hacer las líneas ocultas en este sistema de representación 3-D, en orden a menor ocupación de memoria del conjunto, y ganar eficiencia, que se describe a continuación.

La intersección se lleva a cabo formando otra base de datos con la base de superficies. En ella, cada parche de cuatro lados se divide en dos triangulares, de forma que cualquier parche alabeado se aproxima por dos planos. A continuación, se cortan entre sí todos los triángulos de la nueva base de datos, exceptuando las parejas procedentes del mismo parche primitivo de cuatro lados, generando aristas. El corte de triángulos se hace con otro algoritmo que se presenta aquí, y que se expone en otro capítulo más adelante. Todo se realiza con operaciones vectoriales, evitando hacer cortes si no hay posibilidad tras hacer operaciones previas con ventanas. El evitar el uso de determinantes en todos los casos elimina la posibilidad de problemas mal condicionados. La mayoría de los tests establecidos se hacen alrededor de ciertos valores. En punto flotante es difícil obtener valores exactos, y por ello los valores críticos se deben interpretar en cada caso particular.

Se debe guardar una precaución respecto al resultado que devuelve el algoritmo de corte de triángulos empleado. Si la traza de corte entre dos triángulos coincide con un trozo de algún lado de los polígonos alabeados de que proceden, debe operarse como si no hubiera tal intersección, puesto que esa arista de corte ya se tiene en cuenta en el dibujo al estar en la base de datos de líneas.

El sistema incluye la posibilidad de incorporar nuevas superficies al sistema sin necesidad de volver a calcular las intersecciones de las antiguas. Para ello, el corte de cada triángulo se hace siempre con los que quedan debajo de él en la lista de la base de datos, recordando cuál fue el último de las superficies antiguas. Cuando entra la nueva superficie, se buscan sólo las aristas generadas por sus cortes con las antiguas (penetraciones), y con la nueva ella misma (autopenetraciones).

Las nuevas líneas son almacenadas en la base de datos de líneas anteriormente descrita, en el mismo

formato que también se ha explicado.

3.1.6 Operaciones globales de perspectiva

Se forma previamente a las operaciones de perspectiva un conjunto de puntos, con todos los vértices de las curvas y superficies almacenadas en la base de líneas, que se llamará Universo. Los puntos de los parches de la base de datos superficial no son incluidos en este Universo porque son coincidentes con los puntos de las rejillas de curvas almacenadas en la base de líneas. En dicha base, además, existen los vértices de las curvas que no pertenecen a superficies y que no aparecen por tanto en la base de datos de superficies.

En orden a evitar problemas de cálculo en el ordenador, el rango de las tres coordenadas para cada punto del Universo se normaliza a un cubo que circunscriba al dibujo, a su vez de dimensiones normalizadas.

Una vez en este punto, es el momento de realizar las operaciones de perspectiva. Se incluyen en este sistema dos tipos de perspectiva: ortogonal y cónica. La visión del objeto 3-D se realiza definiendo tres ángulos: rotación, azimuth y elevación, y, en el caso de la perspectiva cónica, hace falta también definir el punto de vista.

El objetivo de las transformaciones es colocar un eje (eje x) paralelo a la línea de visión y hacia el observador, de forma que el dibujo se pueda realizar en el plano y - z , dibujando tan sólo estas dos coordenadas. La coordenada x queda tan sólo con la información de profundidad, al objeto de calcular las líneas ocultas con posterioridad.

Después de realizar estas operaciones se crea una nueva base de datos de líneas con los vértices girados y proyectados y las mismas curvas de la base de datos inicial. Si consideramos el conjunto de transformaciones como una transformación global lineal, se pueden definir sus parámetros sobre el estado final y el inicial de cada vértice. Ahora se debe aplicar a los vértices de los parches de la base de datos de superficies la misma transformación global con los parámetros evaluados, creando una nueva base de datos superficial con los mismos parches girados. Esto ha de ser así porque los parches representan la misma estructura que las rejillas de curvas, y deben girarse y proyectarse en la misma forma que ellas.

Las operaciones de perspectiva ortogonal se basan en giros sobre los ejes. Dado un azimuth, se gira sobre el eje z . Dada una elevación, se gira sobre el eje y . Dada una rotación, se gira sobre el eje x . Los giros se hacen en este orden y sobre los ejes tal como quedan del giro anterior. La expresión de giro de un

entre el punto de vista y el centro de la proyección, y el plano de proyección y el centro de la proyección, de acuerdo con la Figura 73 en la página 73:

$$\begin{aligned}\frac{y' - y_{op}}{y - y_{op}} &= \frac{D + R}{R - (x - x_{op})} \\ \frac{z' - z_{op}}{z - z_{op}} &= \frac{D + R}{R - (x - x_{op})}\end{aligned}\tag{3.5}$$

donde y_{op}, z_{op} son las coordenadas proyectadas en 2-D tras la operación de perspectiva ortogonal, y', z' son las coordenadas proyectadas en 2-D tras la aplicación de la operación de cónica. Si el cubo circunscrito está normalizado a 100 de lado, $y_{op} = z_{op} = 50$. R es la distancia del centro de la proyección al plano de proyección. En orden a evitar problemas de cálculo, es necesario aplicar las mismas operaciones también la coordenada x , siendo $x_{op} = 50$ la correspondiente al centro de la proyección. Despejando en la expresión (3.5) e introduciendo la coordenada x , se tiene:

$$\begin{aligned}y' &= \frac{R + D}{R - (x - x_{op})} (y - y_{op}) + y_{op} \\ z' &= \frac{R + D}{R - (x - x_{op})} (z - z_{op}) + z_{op} \\ x' &= \frac{R + D}{R - (x - x_{op})} (x - x_{op}) + x_{op}\end{aligned}\tag{3.6}$$

Valores razonables para D son entre 50 y 200. Con $D = 50$ y $R > 200$ se tiene la perspectiva ortogonal directamente. Ideas similares a las expuestas pueden verse en (6,7).

3.1.7 Procedimientos de presentación

3.1.7.1 Dibujo 3-D de las estructuras

Suele ser normal disponer de un procedimiento de presentación de líneas y superficies que no incluya el cómputo de las partes ocultas. Esto es muy útil, pues elimina el tiempo de cálculo de líneas ocultas, y permite escoger la posición y vista definitiva del dibujo.

El dibujo definitivo no debe mostrar las zonas ocultas tras estructuras opacas. Se ha desarrollado un algoritmo de líneas ocultas especialmente para este sistema. Su descripción detallada es objeto de capítulos posteriores. No obstante se adelantan algunas de sus particularidades en este punto:

- El algoritmo de líneas ocultas que se presenta trabaja con estructuras 3-D multiformes en coordenadas paramétricas.
- La disposición de las dos bases de datos de entrada, de líneas y de superficies, hace posible hacer el mismo proceso con líneas y superficies, sin hacer distinciones entre las dos estructuras.
- Se ahorra gran cantidad de CPU por el hecho de que cada operación lleva antes algún test previo que determina si es posible la situación que ha de calcular dicha operación.
- Todas las operaciones se realizan con vectores, evitando los convencionales determinantes, ganando en rapidez, exactitud y evitando en lo posible errores de redondeo.
- Los movimientos del trazador de curvas que ha de dibujar al final se optimizan para hacer más rápida esta operación.
- El algoritmo trata los vértices agudos en forma correcta, sin ningún tipo de error.
- Todas las penetraciones y autopenetraciones de las estructuras 3-D del dibujo se resuelven.

3.1.7.2 Presentación de ejes 3-D

Los ejes son aristas del cubo normalizado y circunscrito que ya se ha mencionado, y deben escogerse de tal forma que no se superpongan al dibujo y tengan un vértice común. El sistema que se describe a continuación ha sido optimizado para perspectivas ortogonales.

Se comprueba que existen 48 formas diferentes de escoger las aristas del cubo como ejes. Pueden agruparse en 6 formas para diferentes rangos de ángulos de rotación, 4 para sendos rangos de variación del azimut, y 2 para los correspondientes rangos de variación de la elevación. Como son independientes todas las posibilidades, el número de formas diferentes de escoger los ejes es el producto $6 \times 4 \times 2 = 48$.

En la Figura 74 en la página 77 se puede ver el hexágono correspondiente al cubo circunscrito normalizado. Para escoger tríadas de aristas como ejes, se toman las dos inferiores y la de la izquierda. Al ser un hexágono, hay que cambiar de tríada cada 60 grados, lo que hace 6 posibilidades ($360/6$).

En la Figura 75 en la página 77 se puede ver el cuadrado correspondiente al mismo cubo. Para escoger la tríada de ejes con los mismos criterios, hay que cambiarla cada 90 grados, lo que hace 4 posibilidades ($360/4$).

El rango de elevación es más simple. Basta coger la tríada superior o la inferior. Son dos posibilidades. Para escoger la tríada entre las distintas posibilidades, primero se mira el azimut. Para cada uno de los cuatro rangos, se mira la elevación, y para cada elevación, se mira la rotación. De este encadenamiento sale la tríada óptima de aristas que pueden conformar los ejes del cubo.

3.1.7.3 Dibujo 3-D de curvas de nivel

Es conveniente disponer de un sistema que represente curvas de nivel. En este caso se ha diseñado un algoritmo para la obtención de dichas curvas, dadas las características multiformes de las superficies en el dibujo. Las curvas de nivel con esta clase de superficies pueden presentar intersecciones, no siendo útiles algoritmos convencionales.

El algoritmo propuesto aprovecha las ideas de intersección de superficies necesarias y desarrolladas para el algoritmo de líneas ocultas propuesto, en orden a aumentar la eficiencia y disminuir la ocupación global de memoria del sistema de representación. No obstante, algunos otros algoritmos se proponen en la bibliografía (59,62).

Figura 74. Hexágono correspondiente a un cubo proyectado

El sistema permite dibujar las curvas de nivel deseadas en proyección sobre un plano cualquiera o en 3-D, con y sin líneas ocultas. El escalado de niveles se realiza mediante una opción que asigna colores a dichos niveles. El nivel máximo corresponde al rojo saturado, y el mínimo al púrpura saturado, siguiendo el borde del Spectrum Locus para los niveles intermedios, en forma lineal mediante valores asignados a una tabla de consulta de vídeo (Video Look-up Table, VLT). Como siempre, la representación gráfica práctica de estos colores tiene lugar según el triángulo de Maxwell, en el interior del Spectrum Locus.

Figura 75. Cuadrado correspondiente a un cubo visto desde arriba o abajo

El algoritmo aquí propuesto está basado en consideraciones similares a la intersección entre superficies multiformes. El punto de partida es la base de datos de superficies descrita con anterioridad al formar una superficie 3-D. Para cada parche de cuatro lados de la base de datos se define un parche de cuatro lados plano y rectangular, donde la coordenada correspondiente al nivel deseado es igual en los cuatro vértices a dicho nivel. Las otras coordenadas de los vértices se hacen de forma que la proyección del parche de la base de datos sobre el plano que contiene el parche rectangular plano esté inscrita en este último parche, con un margen de seguridad, para evitar problemas de coincidencia de vértices o aristas en intersecciones posteriores.

Dado que se ha diseñado aquí un algoritmo nuevo de corte entre triángulos en el espacio 3-D, que se explica en otro capítulo, parece lo más sencillo utilizarlo aquí. Para ello, el parche plano se divide en 2 parches triangulares y el parche de cuatro lados de la base de datos de superficies se divide en 2 parches triangulares planos que lo aproximan. Las intersecciones de los triángulos procedentes del parche alabeado de cuatro lados de la base de datos de superficies contra los triángulos del parche plano definido sobre aquél, dan segmentos correspondientes a las curvas de nivel.

Las curvas de nivel obtenidas se almacenan en una base de datos de líneas, del modo expuesto para la base de líneas que describe la superficie como una rejilla. Los diferentes niveles se distinguen por un código en el dibujo de la línea, o por el color, de la forma que ya se ha mencionado en capítulos anteriores que introducían la necesidad de representar curvas de nivel 3-D de superficies multiformes.

3.1.8 Resultados

Se puede ver en la Figura 76 en la página 79 un cilindro con ejes en perspectiva ortogonal. En la Figura 77 en la página 80 se ve el mismo cilindro en perspectiva cónica, donde los ejes no están optimizados, ya que, como se ha explicado, sólo se ha hecho esta operación para la perspectiva ortogonal. En la Figura 78 en la página 81 se puede ver la envolvente del campo eléctrico de una onda electromagnética modulada en AM, que se propaga con atenuación y polarización circular. En la Figura 79 en la página 82 se puede ver un ejemplo de esfera en perspectiva ortogonal. En la Figura 80 en la página 83 se ve un caso de autopenetración. Una composición sobre una primitiva única se puede ver en la Figura 81 en la página 84 y en la Figura 82 en la página 85. En la Figura 83 en la página 86 se puede ver una línea que penetra un cilindro abierto, localizando perfectamente los puntos de penetración para hacer las zonas ocultas. Por último, en la Figura 84 en la página 87 se ve una composición de figuras con todas sus penetraciones sobre esferas abiertas, y en la Figura 85 en la página 88 otra composición con dos primitivas.

Figura 76. Cilindro con ejes en perspectiva ortogonal

En lo que se refiere a curvas de nivel, se pueden ver las de de una función sinc bidimensional en la Figura 86 en la página 89, y las de la imagen original de la Figura 87 en la página 90 en la Figura 88 en la página 91.

Figura 77. Cilindro con ejes en perspectiva cónica

Figura 78. Envolvente del campo eléctrico de una onda electromagnética

Figura 79. Esfera en perspectiva ortogonal

Figura 80. Prisma autopenetrado y abierto en cónica

Figura 81. Composición de la misma primitiva tres veces con giros y penetraciones

Figura 82. Composición de la misma primitiva siete veces con giros y penetraciones

Figura 83. Composición de líneas y superficies con penetración

Figura 84. Composición de primitivas penetrantes

Figura 85. Composición de dos primitivas

Figura 86. Curvas de nivel de función sinc bidimensional

Figura 87. Imagen original para curvas de nivel

3.2 Algoritmo propuesto de líneas ocultas con superficies multiformes

3.2.1 Descripción general

Una vez efectuadas las operaciones globales de perspectiva por el procedimiento previamente descrito, la primera operación es transformar la base de datos de superficies, que contiene las estructuras opacas, en otra cuyos parches son triangulares. Es la llamada base de triángulos útiles, pues sólo se almacenan los que podrían ocultar estructuras después de su proyección sobre un plano ortogonal a la línea de visión (eje x tal como se escogió), eliminándose todos los que degeneran en líneas o puntos. Todos los parches de cuatro lados suelen convertirse en dos triángulos. Sin embargo, algunos parches de cuatro lados pueden degenerar en uno sólo, o en ninguno. El número total de situaciones posibles es de 81 (92).

Después, la base de datos de líneas es comparada contra la de triángulos útiles, detectando intersecciones, penetraciones y líneas ocultas. El algoritmo está optimizado en el sentido de no operar sobre áreas donde la ocultación sea imposible. Para ello se realizan algunas operaciones con ventanas 2-D y 3-D, y algunas operaciones vectoriales de productos mixtos, escalares y vectoriales antes de buscar la ocultación.

Todas las operaciones se llevan a cabo con vectores, eliminando los determinantes, utilizados en proce-

Figura 88. Curvas de nivel de la imagen original anterior

dimientos convencionales (6). Así se evitan todos los problemas mal condicionados. Por otra parte, la obtención de resultados exactos es imposible operando en punto flotante, y por eso los resultados cercanos al cero o a valores test fijos deben ser interpretados en cada caso particular.

La salida del algoritmo la constituye otra base de datos que contiene las líneas a pintar por ser vistas en el dibujo. Esta base de datos está optimizada para su dibujo, minimizando los movimientos de un trazador de curvas.

El algoritmo propuesto realiza un correcto tratamiento de los vértices agudos. Algunos algoritmos convencionales (88,89) no son capaces de resolver estas situaciones correctamente, provocando fuertes errores. El sistema propuesto resuelve con éxito todas las penetraciones entre las estructuras 3-D del dibujo, lo cual es otra ventaja sobre los algoritmos antiguos. Es importante destacar que una superficie multiforme

puede autopenetrarse, y esta situación también se resuelve. Por otra parte, existen muy pocos algoritmos de líneas ocultas con las estructuras en coordenadas paramétricas, que de esta forma son multiformes la mayoría de las veces. De esta forma, el método propuesto es muy útil en gran número de aplicaciones para diseño de objetos 3-D.

3.2.2 Entradas al sistema

Dos bases de datos constituyen la entrada al sistema: la de líneas y la de superficies. Ambas ya han sido explicadas en la descripción del sistema de representación 3-D. Las dos bases de datos ya contienen las modificaciones de perspectiva y punto de vista deseados.

3.2.3 Conversión de la base de datos superficial

El estado de la base de datos de superficies no es útil porque los parches de cuatro lados no son planos generalmente. Se necesita una conversión a parches triangulares. Un parche de cuatro lados se descompone como máximo en dos triángulos, dependiendo del aspecto de su proyección 3-D y el punto de vista. Por otra parte, se trata de eliminar los parches que al ser proyectados dejan de ocultar estructuras, y así simplificar el proceso.

La partición de los parches de cuatro lados se hace de acuerdo al criterio descrito por Ohno (92). Consiste en el cálculo de las cuatro áreas de los cuatro posibles triángulos planos formados por los cuatro vértices del parche. Una simple inspección de los signos de estas áreas es suficiente para concluir la forma de la partición. El parche puede desaparecer en algunos casos, porque su proyección degenera en un punto o se reduce a una línea, o simplemente degenera en un solo triángulo. Solamente son posibles 81 situaciones de proyección de parches de cuatro lados (92). La Figura 89 en la página 93 muestra estas posibilidades. Los cuadros sombreados o son imposibles combinaciones o son degeneraciones que hacen que el parche no pueda ocultar estructuras.

3.2.4 El núcleo del algoritmo

Las bases de datos de entrada están listas para inicializar el proceso. El algoritmo de líneas ocultas está basado en comparar la base de datos de líneas contra la base de superficies convertida, que ocultará algunos trozos de líneas. Las partes visibles de la base de datos de líneas son las que se dibujan después de la aplicación del algoritmo.

El proceso hace para cada línea de la base de datos de líneas una comparación de cada uno de sus

A134		+			0			-		
A123	A234	+	0	-	+	0	-	+	0	-
	A124									
+	+									
	0									
	-									
0	+									
	0									
	-									
-	+									
	0									
	-									

A134		+			0			-		
A123	A234	+	0	-	+	0	-	+	0	-
	A124									
+	+	123 134 1	124 2	123 134 3	123 10	123 11	123 12	124 234 19	124 20	125 354 21
	0	234 4	5	6	123 13	14	15	234 22	NADA 23	243 24
	-	123 134 7	8	9	123 16	17	18	145 235 25	142 26	142 243 27
0	+	134 28	134 29	134 30	37	38	NADA 39	46	47	143 48
	0	134 31	32	33	40	NADA 41	42	49	50	143 51
	-	134 34	35	36	NADA 43	44	45	143 52	143 53	143 54
-	+	124 234 55	124 56	154 253 57	64	65	132 66	73	74	132 143 75
	0	234 58	NADA 59	234 60	67	68	132 69	76	77	243 78
	-	152 345 61	142 62	142 243 63	132 70	132 71	132 72	132 143 79	142 80	132 143 81

Figura 89. Casos posibles de proyección de un polígono alabeado de 4 lados

segmentos contra todos los parches triangulares en la forma que se describe a continuación.

1. El primer paso consiste en una operación de ventana 3-D entre una línea y todos los parches triangulares. Si no hay volúmenes comunes entre los respectivos paralelepípedos rectangulares o la línea está delante de un triángulo dado, se marca este triángulo. Así, no se hará ningún test posterior con ninguno de los segmentos rectos de la línea contra ese triángulo.

2. Se hace a continuación otra operación de ventanas 3-D entre cada segmento de la línea y todos los triángulos no marcados. De ahora en adelante, las operaciones que se describirán serán relativas a un solo segmento y un solo triángulo, siendo éste último del conjunto de los que superen esta segunda operación de ventana 3-D.
3. El siguiente paso es una operación de corte 2-D entre las proyecciones del triángulo y la recta que contiene al segmento, ambos proyectados ortogonalmente a la línea de visión. Si existe intersección, se hace una nueva operación de corte entre las mismas proyecciones, pero del segmento y el triángulo. Si de nuevo existe intersección, el trozo de segmento proyectado que queda en el interior del triángulo proyectado, se denomina candidato a ser oculto.
4. Se realiza ahora un test 3-D de penetración entre el candidato y el triángulo, determinando el trozo correcto de segmento oculto.
5. Por último, se hace una reordenación de todos los trozos contiguos de cada segmento, borrando, sustituyendo y absorbiendo dichos trozos según sea necesario. El objetivo es optimizar los movimientos del trazador de curvas. Todas las operaciones son repetidas para cada línea de la base de datos de líneas.

En la descripción que sigue, el concepto de línea será aplicado a un conjunto de segmentos rectos consecutivos, cada uno con un vértice común con el siguiente. El concepto de parche se aplicará a los parches triangulares útiles.

3.2.5 Corte entre líneas y triángulos planos en el espacio 3-D

3.2.5.1 Una línea contra un parche

Se define un paralelepípedo rectangular circunscrito sobre cada línea del dibujo, y lo mismo sobre cada parche. Si no existen volúmenes comunes entre los dos paralelepípedos, el parche se marca y ningún segmento de la línea se intenta cortar contra el triángulo en los pasos siguientes. El mismo criterio se aplica cuando el paralelepípedo del parche está detrás del de la línea, mirando desde el punto de vista.

3.2.5.2 Un segmento contra un parche

Se define otro paralelepípedo 3-D circunscrito a cada segmento de línea, y lo mismo para cada parche triangular. Si no existen volúmenes comunes entre los dos paralelepípedos, el segmento no se intenta cortar contra el parche. El mismo criterio se aplica cuando el paralelepípedo del parche está detrás del de la línea, mirando desde el punto de vista.

3.2.6 Corte entre un segmento y un triángulo plano en el espacio 3-D

Una vez detectado que puede existir ocultación por parte del triángulo se realiza esta operación, que a su vez se puede dividir en tres pasos, tal como se describe a continuación, calculando las posiciones relativas entre las estructuras que se detallan.

3.2.6.1 Posición relativa de las proyecciones ortogonales a la línea de visión de una recta y un triángulo en el espacio 3-D

El plano de proyección es el mismo que el utilizado en el cálculo de triángulos útiles, es decir, ortogonal al eje de visión. Esta operación se hace calculando tres productos cruzados, con los vectores mostrados en la Figura 90 en la página 96.

El segmento está entre los puntos 1 y 2, y el triángulo tiene sus vértices en los puntos A,B y C. Los productos cruzados son los que se especifican en la ecuación siguiente:

$$\begin{aligned} p_1 &= \vec{r} \wedge \vec{u} \\ p_2 &= \vec{r} \wedge \vec{v} \\ p_3 &= \vec{r} \wedge \vec{w} \end{aligned} \tag{3.7}$$

Figura 90. Posición relativa de las proyecciones en 2-D de una recta y un triángulo

Si las tres cantidades P_1 , P_2 y P_3 tienen el mismo signo, no puede haber cortes entre el triángulo y la dirección de la línea. Por el contrario, los cambios de signo marcan los lados del triángulo que intersectan con la misma dirección, de acuerdo con la Figura 91, siempre según la Figura 90 y la ecuación (3.7). Se puede llegar a las mismas conclusiones calculando tres determinantes (6), pero este tipo de operaciones tiene gran cantidad de problemas en los ordenadores.

$P_1 = \vec{r} \wedge \vec{u}$	+	+	+	+	-	-	-	-
$P_2 = \vec{r} \wedge \vec{v}$	+	+	-	-	+	+	-	-
$P_3 = \vec{r} \wedge \vec{w}$	+	-	+	-	+	-	+	-
Lados del triángulo que intersectan la dirección de la línea	No hay	AC BC	AB BC	AB AC	AB AC	AB BC	AC BC	No hay

Figura 91. Signos en los productos cruzados en el caso de una recta contra un triángulo proyectados en 2-D

Una situación particular aparece cuando uno o más productos cruzados son nulos. Esto implica que no hay intersecciones, pero existe o un punto de contacto o un lado de contacto entre la recta y el triángulo. El cálculo de ceros siempre es un problema en punto flotante. Así, se define un entorno del cero en el que todos los productos que estén en su interior se consideran nulos. Los lados del triángulo que intersectan con la recta se muestran en la Figura 91 en la página 96. Como puede observarse en dicha figura, no existen casos de productos nulos. Esto se debe al tratamiento que se hace en estas situaciones como sigue:

1. Un producto es nulo y los otros dos tienen signos diferentes. Consecuentemente, hay dos intersecciones entre la recta y el triángulo, aunque uno de ellos es doble, porque tiene lugar en un vértice. En este caso el producto nulo se cambia, asignándole un valor positivo muy pequeño, menor que un umbral dado como valor de un entorno del cero donde todo se considera nulo. De esta forma, la única situación de producto nulo que daría intersecciones se transforma y no aparecen situaciones de producto cero en la Figura 91 en la página 96.
2. Un producto es nulo y los otros dos tienen el mismo signo, en cuyo caso no hay intersección, y no puede haber ocultación.
3. Dos productos son nulos, en cuyo caso tampoco hay intersección, sino lado común, y, por tanto, tampoco hay posibilidad de ocultación.
4. Los tres productos nulos no es situación posible cuando el triángulo no tiene posibilidad de estar degenerado, como es aquí el caso.

3.2.6.2 Posición relativa de las proyecciones ortogonales a la línea de visión de un segmento y un triángulo en el espacio 3-D

El plano de proyección es el mismo especificado anteriormente. Después de determinar los dos lados del triángulo que intersectan con la recta soporte del segmento, (por ejemplo AB y AC en la Figura 92 en la página 98), esta operación se realiza calculando cuatro productos cruzados, con los vectores que se muestran en la Figura 92 en la página 98.

El segmento está delimitado por los puntos 1 y 2, y el triángulo tiene sus vértices en los puntos A,B y C. Los productos cruzados son, siguiendo la misma Figura 92 en la página 98:

Figura 92. Posición relativa de las proyecciones en 2-D de un segmento y un triángulo

P_1	P_2	P_3	P_4	Posibles trozos del segmento a ser ocultos
-	-	-	-	12
+	+	+	+	12
+	-	-	-	E2
-	+	+	+	E2
-	+	-	-	1F
+	-	+	+	1F
-	-	-	+	1F
+	+	+	-	1F
-	-	+	-	E2
+	+	-	+	E2
+	-	-	+	EF
-	+	+	-	EF

Figura 93. Signos en los productos cruzados en el caso de un segmento contra un triángulo proyectados en 2-D

$$\begin{aligned}
p_1 &= \vec{u} \wedge \vec{r} \\
p_2 &= \vec{u} \wedge \vec{s} \\
p_3 &= \vec{w} \wedge \vec{r} \\
p_4 &= \vec{w} \wedge \vec{s}
\end{aligned} \tag{3.8}$$

El trozo de segmento que puede ser ocultado por el triángulo se determina de acuerdo con la Figura 93 en la página 98, siempre según la Figura 92 en la página 98 y la ecuación (3.8). En dicha figura, se denomina E al punto de corte del segmento y el lado AB del triángulo, y F al correspondiente punto de corte con el lado AC del mismo segmento.

Si p_1 y p_2 tienen el mismo signo, p_3 y p_4 tienen el mismo signo también, y el signo de p_1 y p_2 es el contrario del de p_3 y p_4 , entonces no existe intersección entre el segmento y el triángulo. Las situaciones de producto nulo de p_1 y p_2 , o p_3 y p_4 simultáneamente, no son posibles. Se entiende por producto nulo cualquier producto cuyo valor esté en un entorno definido del cero, como ya se ha descrito anteriormente.

Algunos casos se tratan de forma especial. Si p_1 es cero y p_2 no, entonces se asigna a p_1 un valor con el mismo signo de p_2 , pero muy pequeño, menor que el umbral para considerarlo nulo. Si p_2 es cero y p_1 no, entonces se asigna a p_2 un valor con el mismo criterio. El proceso se repite para p_3 y p_4 . Después de estos cambios, no puede aparecer ningún caso de producto nulo. De esta forma, el análisis de los casos posibles de p_1 , p_2 , p_3 , y p_4 se simplifica, de acuerdo con la Figura 93 en la página 98.

Algunas ideas similares las expone Pavlidis en (6). Sin embargo, las operaciones se realizan en forma diferente. En (6) se hace uso de determinantes, con todos los problemas mal condicionados que acompañan su uso en ordenadores. El algoritmo aquí propuesto realiza todas las operaciones con productos entre vectores, evitando todos los problemas que dan los determinantes.

Todos los procesos de corte se llevan a cabo en coordenadas paramétricas. Es bien sabido que un segmento se corta con otra estructura geométrica si el valor del parámetro en la dirección del segmento pertenece al intervalo cerrado $[0,1]$. Todos los segmentos están ordenados en la base de datos, y de esta forma los valores 0 y 1 del parámetro se asignan unívocamente a sus vértices. Se debe tener otra precaución en este punto con los problemas derivados de operar en punto flotante. Así, se definen sendos entornos de los puntos 0 y 1, de tal forma que cualquier valor de parámetro que salga de un cálculo y pertenezca a ellos estando fuera del mismo intervalo cerrado, se cambia por 0 y 1 respectivamente, provocando la existencia de intersección.

Los trozos de segmentos candidatos a ser ocultos no tienen sus puntos extremos ordenados de acuerdo a su valor de parámetro generalmente. Así, si el valor del parámetro para el segundo punto extremo es menor que para el primero, los dos puntos se intercambian. Es otra ventaja de las coordenadas paramétricas, ya que sólo ha sido necesario comparar el valor del parámetro para la ordenación, que además es un número entre 0 y 1.

3.2.6.3 Posición relativa de un segmento y un triángulo plano en el espacio 3-D

El trozo de segmento que podría quedar oculto por un triángulo ya se ha determinado con la operación anterior. Ahora, el camino es detectar las posibles penetraciones del candidato a ser oculto por el triángulo que tiene posibilidad de hacerlo.

Se muestra un ejemplo en la Figura 95 en la página 106 que aparece en el capítulo siguiente. El cálculo de dos productos mixtos es suficiente para saber si existe penetración. Estos productos son los siguientes:

$$\begin{aligned} p_1 &= \vec{r} \cdot (\vec{u} \wedge \vec{v}) \\ p_2 &= \vec{a} \cdot (\vec{u} \wedge \vec{v}) \end{aligned} \tag{3.9}$$

A partir de estas expresiones, se puede hacer la siguiente discusión:

1. Si p_1 y p_2 tienen signos diferentes, la penetración aparece.
2. Se debe tener precaución de nuevo con los valores que no son nulos y deben serlo, por estar en un entorno del cero. Con este criterio sobre producto nulo, si p_1 es cero y p_2 no, entonces se asigna a p_1 un valor muy pequeño con el mismo signo de p_2 .
3. Si p_2 es cero y p_1 no, se asigna a p_2 un valor con el mismo criterio.
4. La situación de p_1 y p_2 cero simultáneamente no es posible.

Si no existen penetraciones, se prueba un punto extremo del candidato contra un vértice del triángulo, en la dirección del punto de vista, que, al estar giradas las bases de datos, debe coincidir con el eje x. Si la coordenada x del candidato está más próxima al punto de vista que la correspondiente del vértice del triángulo, el candidato es visible. Si no, es oculto.

Si existe penetración, el punto donde tiene lugar en el triángulo debe ser determinado. Este problema se resuelve calculando cuatro productos mixtos, siendo uno de ellos el denominador de normalización de los otros tres. Estos productos son los siguientes:

$$\begin{aligned}\lambda &= \frac{\vec{r} \cdot (\vec{v} \wedge \vec{a})}{\vec{r} \cdot (\vec{u} \wedge \vec{v})} \\ \beta &= \frac{\vec{r} \cdot (\vec{a} \wedge \vec{u})}{\vec{r} \cdot (\vec{u} \wedge \vec{v})} \\ t &= \frac{\vec{a} \cdot (\vec{u} \wedge \vec{v})}{\vec{r} \cdot (\vec{u} \wedge \vec{v})}\end{aligned}\tag{3.10}$$

Las cantidades λ y β son los dos parámetros del triángulo, a lo largo de los lados AB y AC, cuyos vectores son $\vec{u}$ y $\vec{v}$ respectivamente. La cantidad t es el parámetro a lo largo de la dirección del segmento. Los tres parámetros deben estar en el intervalo cerrado $[0,1]$, porque ésta es la condición de existencia de penetración.

Un nuevo test se realiza para determinar el trozo de candidato oculto por el triángulo, ya que al haber penetración, un trozo de candidato será visible. Se comparan las coordenadas x (dirección del punto de vista) de un extremo del segmento y del punto de penetración. Si el extremo está más cerca del punto de vista que el punto de penetración, entonces el candidato es visible entre aquel extremo y el punto de penetración, siendo el resto oculto. Si ocurre lo contrario, el candidato es visible entre el otro extremo y el punto de penetración, siendo el resto oculto.

A continuación es necesaria una reordenación de los dos vértices del definitivo trozo de segmento que queda oculto, de acuerdo a los valores de parámetro. Precauciones similares a las ya descritas deben tenerse en cuenta en las cercanías de los valores 0 y 1.

En el caso de que un segmento tenga ocultos uno o más trozos, se marca al objeto de simplificar algunas etapas posteriores.

3.2.7 Reducción de los trozos ocultos de un segmento

Cada vez que aparece un trozo oculto en un segmento, se compara con los trozos ocultos previamente encontrados sobre el mismo segmento. Si son contiguos o tienen zona común, el último es absorbido, y el antiguo correspondiente es alargado de forma que incluya el nuevo. Si el último trozo oculto está incluido en alguno antiguo oculto también, directamente se borra el nuevo.

Si aparece una operación de absorción, borrado o sustitución de trozos ocultos sobre un segmento, debe hacerse una comparación con el resto de los trozos ocultos del mismo segmento no implicados en la operación. Esto ha de ser así, pues al crear trozos más grandes, pueden aparecer nuevas zonas comunes o contigüidades entre los trozos ocultos ya existentes de un segmento. La operación se repite hasta que sea imposible hacer sustituciones, borrados o absorciones entre los trozos. Al final, todos los trozos ocultos de un segmento deben ser disjuntos.

Después se hace una reordenación de los trozos ocultos de cada segmento, de acuerdo con el valor de parámetro de sus vértices. Al ser disjuntos, esta reordenación no tiene ningún problema con el solapamiento de trozos.

3.2.8 Optimización para el dibujo

La optimización para el dibujo está basada en la forma de almacenamiento de los trozos visibles de los segmentos en una base de datos final. El formato de almacenamiento es el mismo que el de la base de datos de líneas inicial. Los trozos visibles de los segmentos contiguos que pertenecen a la misma línea se detectan, y se ordenan unos a continuación de otros, siguiendo el orden de la línea inicial, antes de la aplicación del algoritmo de líneas ocultas. De esta forma, un trazador de curvas puede dibujar las líneas vistas de las estructuras 3-D de forma ordenada, evitando movimientos inútiles.

El formato final de almacenamiento consiste en tomar las líneas de la base de datos inicial y cortar dichas líneas cada vez que aparece un trozo oculto en alguno de sus segmentos, abriendo una línea nueva en la base de datos, hasta el trozo oculto siguiente del mismo u otro segmento.

4.0 Ecuaciones vectoriales propuestas 2-D y 3-D

Se va a tratar en este capítulo de un conjunto de ecuaciones vectoriales propuestas para el cálculo de intersecciones en el espacio 3-D de rectas (o líneas, como también se las llamará) y segmentos contra triángulos, y de triángulos contra triángulos. Asimismo, se expone otro conjunto de ecuaciones vectoriales que permitirá el cálculo de la proyección de un vector en el espacio 2-D sobre dos direcciones oblicuas. Al objeto de que la descripción de ecuaciones vectoriales quede completa, se incluye el corte entre dos rectas y dos segmentos en el espacio 2-D, aunque esto último ya se puede encontrar en (7).

La ventaja de la utilización de ecuaciones y algoritmos vectoriales reside en el hecho de que evita el uso de determinantes y permite el uso de coordenadas paramétricas. Ambas ventajas eliminan por completo gran cantidad de problemas al realizar los algoritmos en los ordenadores.

4.1 Proyección de un punto sobre un triángulo plano según la dirección de un vector en el espacio 3-D

Sea un triángulo plano en el espacio 3-D formado por dos vectores $\vec{u}$ y $\vec{v}$ y un punto A, que resulta ser el vértice común de los vectores $\vec{u}$ y $\vec{v}$, tal y como puede verse en Figura 94 en la página 104. Se desea obtener el punto Q, proyección del P según la dirección del vector $\vec{r}$. Se define un vector $\vec{a}$ entre el punto P y el punto A.

Las coordenadas del punto Q se hallan según las ecuaciones siguientes:

Figura 94. Proyección de un punto sobre un triángulo según la dirección de un vector en el espacio 3-D

$$\begin{aligned}
 \lambda &= \frac{\vec{r} \cdot (\vec{v} \wedge \vec{a})}{\vec{r} \cdot (\vec{u} \wedge \vec{v})} \\
 \beta &= \frac{\vec{r} \cdot (\vec{a} \wedge \vec{u})}{\vec{r} \cdot (\vec{u} \wedge \vec{v})} \\
 t &= \frac{\vec{a} \cdot (\vec{u} \wedge \vec{v})}{\vec{r} \cdot (\vec{u} \wedge \vec{v})}
 \end{aligned}
 \tag{4.1}$$

donde λ y β son los parámetros de las direcciones de los vectores $\vec{u}$ y $\vec{v}$ respectivamente, y t es el parámetro en la dirección de $\vec{r}$.

4.2 Corte de una recta contra un plano en el espacio 3-D

La recta se define por un punto y un vector, y el plano por un punto y dos vectores. El esquema es el mismo que el de la Figura 94, donde la línea se define por el punto P y el vector $\vec{r}$. El punto de corte es el punto Q, dado por la ecuación (4.1).

4.3 Corte de un segmento contra un triángulo plano en el espacio 3-D

El segmento se define por dos puntos, P y R, los cuales definen a su vez un vector $\vec{r}$. El triángulo se define por tres puntos, A, B, y C, que a su vez definen dos vectores, $\vec{u}$ y $\vec{v}$. El esquema se ve en la Figura 95 en la página 106. Se define un vector $\vec{a}$ entre los puntos P y A.

Todas las operaciones encaminadas a buscar el punto Q ya se describieron al cortar una línea contra un plano. Si la operación es entre un segmento perteneciente a una recta y un triángulo perteneciente a un plano, se debe realizar un test con los valores de los parámetros λ , β , y t de la ecuación (4.1).

1. Si λ y β pertenecen al intervalo cerrado $[0,1]$, pero t no, entonces la línea que contiene al segmento intersecta al triángulo.
2. Si t pertenece a dicho intervalo, pero λ y β no, el segmento intersecta al plano que contiene al triángulo.
3. Si los tres parámetros, λ , β y t , están dentro del intervalo, el segmento penetra el triángulo.
4. Por último, si las tres cantidades están fuera del intervalo, la línea que contiene al segmento intersecta al plano que contiene al triángulo.

Un resumen se puede ver en la Figura 96 en la página 106.

4.4 Proyección de un vector en el espacio 2-D sobre dos direcciones no ortogonales

Se definen las dos direcciones sobre las que se quiere proyectar por medio de dos vectores $\vec{v}_1$ y $\vec{v}_2$, tal como se indica en la Figura 97 en la página 107. Se define un vector $\vec{w}$ entre dos puntos, O y P. Las proyecciones $\vec{u}$ y $\vec{v}$ del vector $\vec{w}$ sobre las direcciones $\vec{v}_1$ y $\vec{v}_2$ son dadas por las siguientes expresiones:

Figura 95. Corte de un segmento contra un triángulo plano en el espacio 3-D

$$\begin{aligned}\vec{u} &= \vec{w} + \vec{v}_2 \frac{\vec{v}_1 \wedge \vec{w}}{\vec{v}_2 \wedge \vec{v}_1} \\ \vec{v} &= \vec{w} + \vec{v}_1 \frac{\vec{v}_2 \wedge \vec{w}}{\vec{v}_1 \wedge \vec{v}_2}\end{aligned}\tag{4.2}$$

	$t \in [0,1]$	$t \notin [0,1]$
λ y $\beta \in [0,1]$	Segmento-Triángulo	Línea-Triángulo
λ y $\beta \notin [0,1]$	Segmento-Plano	Línea-Plano

Figura 96. Casos posibles de los tres parámetros cortando un segmento contra un triángulo plano en el espacio 3-D

Figura 97. Proyección de un vector en el espacio 2-D sobre dos direcciones no ortogonales

4.5 Corte de líneas en el espacio 2-D

La situación se puede ver en la Figura 98 en la página 108. Se definen dos líneas por dos vectores $\vec{r}$ y $\vec{s}$. El punto de corte Q se encuentra aplicando las expresiones siguientes:

$$\begin{aligned} t_1 &= \frac{\vec{a} \wedge \vec{s}}{\vec{r} \wedge \vec{s}} \\ t_2 &= \frac{\vec{b} \wedge \vec{r}}{\vec{s} \wedge \vec{r}} \end{aligned} \quad (4.3)$$

Las cantidades t_1 y t_2 son las coordenadas paramétricas del punto Q según las dos líneas, recordando que los vectores que definen las rectas son $\vec{r}$ y $\vec{s}$, respectivamente.

Figura 98. Corte de líneas en el espacio 2-D

4.6 Corte de segmentos en el espacio 2-D

El esquema se puede ver en la Figura 99 en la página 109. Se definen dos segmentos por sus puntos extremos, AB y CD. Estas parejas de puntos definen dos vectores $\vec{r}$ y $\vec{s}$ respectivamente. Se definen dos vectores adicionales $\vec{a}$ y $\vec{b}$. Las mismas operaciones realizadas anteriormente para determinar el corte entre dos líneas en el espacio 2-D son de aplicación en este apartado, incorporando un test posterior. Las ecuaciones necesarias se ven en las expresiones (4.3).

El test a realizar consiste en lo siguiente:

1. Si las cantidades t_1 y t_2 pertenecen al intervalo cerrado $[0,1]$, entonces existe intersección entre los dos

Figura 99. Corte de segmentos en el espacio 2-D

segmentos.

2. Si t_1 pertenece pero t_2 no, el segmento cuya dirección está dada por $\vec{r}$ intersecta la línea que contiene a $\vec{s}$.
3. Si t_1 está fuera del intervalo y t_2 está dentro, el segmento cuya dirección está dada por $\vec{s}$ intersecta la línea que contiene a $\vec{r}$.
4. Si ambos valores, t_1 y t_2 , están fuera del mismo intervalo, las dos direcciones intersectan, pero no los segmentos.

4.7 Corte de planos en el espacio 3-D

Se definen dos planos por medio de tres puntos cada uno. El primero por medio de A, B y C, y el segundo con D, E y F, tal como se indica en la Figura 100 en la página 110. Se definen dos vectores $\vec{r}$ y $\vec{s}$ para Π_1 y otros dos $\vec{u}$ y $\vec{v}$ para Π_2 . Se definen también dos vectores adicionales $\vec{a}$ y $\vec{b}$.

La intersección entre los dos planos es una traza o línea, que puede darse por dos puntos, cuyas coor-

Figura 100. Corte de planos en el espacio 3-D

denadas paramétricas se calculan sobre las expresiones:

$$\begin{aligned}\lambda_1 &= \frac{\vec{a} \cdot (\vec{u} \wedge \vec{v})}{\vec{r} \cdot (\vec{u} \wedge \vec{v})} \\ \beta_1 &= \frac{\vec{a} \cdot (\vec{u} \wedge \vec{v})}{\vec{s} \cdot (\vec{u} \wedge \vec{v})}\end{aligned}\tag{4.4}$$

donde λ_1 y β_1 son las coordenadas paramétricas de dos puntos que pertenecen a la traza en las direcciones de los vectores $\vec{r}$ y $\vec{s}$ respectivamente.

La misma traza se puede hallar con las expresiones:

$$\lambda_2 = \frac{\vec{b} \cdot (\vec{r} \wedge \vec{s})}{\vec{u} \cdot (\vec{r} \wedge \vec{s})}$$

$$\beta_2 = \frac{\vec{b} \cdot (\vec{r} \wedge \vec{s})}{\vec{v} \cdot (\vec{r} \wedge \vec{s})}$$
(4.5)

donde λ_2 y β_2 son las coordenadas paramétricas de otros dos puntos que pertenecen a la misma traza en las direcciones de los vectores $\vec{u}$ y $\vec{v}$ respectivamente.

4.8 Corte de triángulos planos en el espacio 3-D

Se definen dos triángulos TR1 y TR2 por sus vértices, tal como se indica en la Figura 101 en la página 112. El proceso se realiza en los siguientes pasos:

1. Se realiza una operación con ventanas en el espacio 3-D, y si existe intersección entre los volúmenes de los dos paralelepípedos rectangulares que circunscriben los dos triángulos en dicho espacio 3-D, el proceso continúa.
2. El paso siguiente es hacer un test de coincidencia con los vértices. Este test debe realizarse cuidadosamente en los valores nulos. La forma más normal de evaluar este paso consiste en calcular la diferencia entre las coordenadas 3-D de los vértices de los triángulos. El resultado nulo no puede aparecer normalmente en operaciones en punto flotante. En estos casos, los valores cuyo módulo sea más pequeño que un umbral dado se interpretan como nulos. Podemos distinguir las siguientes situaciones:
 - Si únicamente un vértice es coincidente, directamente no hay traza de intersección, pues sólo existe dicho punto de contacto, o bien los dos triángulos son coplanarios, en cuyo caso tampoco aparece traza de corte en el espacio 3-D.
 - Si dos o más de los vértices de uno de los triángulos son coincidentes con otros vértices del segundo triángulo, entonces existe al menos una arista común. El segmento de la traza de corte definitivo es la propia arista común.
3. Seis productos mixtos determinan si los dos triángulos se cortan, o si la intersección sólo tiene lugar entre uno de ellos y el plano que contiene al otro. Estos productos son los siguientes, según la Figura 100 en la página 110:

Figura 101. Corte de triángulos planos en el espacio 3-D

$$\begin{aligned}
 p_1 &= \vec{a} \cdot (\vec{r} \wedge \vec{s}) \\
 p_2 &= \vec{a}_1 \cdot (\vec{r} \wedge \vec{s}) \\
 p_3 &= \vec{a}_2 \cdot (\vec{r} \wedge \vec{s}) \\
 p_4 &= \vec{b} \cdot (\vec{u} \wedge \vec{v}) \\
 p_5 &= \vec{b}_1 \cdot (\vec{u} \wedge \vec{v}) \\
 p_6 &= \vec{b}_2 \cdot (\vec{u} \wedge \vec{v})
 \end{aligned} \tag{4.6}$$

4. El producto nulo en la ecuación (4.6) no se presenta normalmente en punto flotante. De la misma forma que antes con las diferencias, se consideran cero todos los productos cuyo resultado esté en un entorno pequeño de dicho valor. Si con este criterio aparecen nulos, se opera como sigue:

- Si entre los tres primeros productos hay uno nulo, y los otros dos tienen el mismo signo, se asigna un valor muy pequeño con el signo de los otros dos al producto nulo.
- El mismo esquema se repite para los tres segundos productos.
- Si entre los tres primeros productos hay 2 nulos, un lado del triángulo definido por $\vec{u}$ y $\vec{v}$ es un segmento de la traza de corte. Concretamente, se trata del lado cuyos vértices provoquen estos

Figura 102. Segmentos a examinar en corte de triángulos planos en el espacio 3-D

nulos. Estos dos vértices se guardan como extremos de uno de los segmentos de traza para ser utilizados en un lugar más adelante del algoritmo donde se efectúan productos escalares, mientras se sigue el camino normal con los tres segundos productos mixtos.

- El mismo esquema se repite con los tres segundos productos, que, si se verifica la situación de tener 2 nulos, un lado del triángulo definido por $\vec{r}$ y $\vec{s}$ sería otro segmento de la misma traza de corte. Los vértices que provocaren los nulos serán guardados directamente para su utilización como extremos de segmento de traza en el paso del algoritmo que efectúa más adelante unos productos escalares, mientras se sigue el camino normal con los tres primeros productos mixtos.
- Si la situación de tener dos nulos se produce simultáneamente entre los tres primeros y los tres segundos productos mixtos, el algoritmo pasa directamente con los vértices que provocan los cuatro nulos como extremos de segmentos de traza al apartado donde efectúa los productos escalares.
- Tres nulos entre los tres primeros o entre los tres segundos productos mixtos implica que los dos triángulos son coplanarios, y no se genera arista de corte nueva en el espacio 3-D.

$\begin{matrix} P_1 \\ \backslash \\ P_2 \end{matrix}$	< 0	$\begin{matrix} \geq 0 \\ y \\ \leq 1 \end{matrix}$	> 1
< 0		AC	AB
$\begin{matrix} \geq 0 \\ y \\ \leq 1 \end{matrix}$	AD	CD	BD
> 1	AB	BC

A = punto de intersección de $\vec{r}$ contra el triángulo de $\vec{u}$ y $\vec{v}$
 B = punto de intersección de $\vec{s}$ contra el triángulo de $\vec{u}$ y $\vec{v}$
 C = punto de intersección de $\vec{u}$ contra el triángulo de $\vec{r}$ y $\vec{s}$
 D = punto de intersección de $\vec{v}$ contra el triángulo de $\vec{r}$ y $\vec{s}$

Figura 103. Tabla final de corte de triángulos planos en el espacio 3-D

5. Una vez eliminada la posibilidad de nullos entre los productos mixtos, se continúa como sigue:

- Si p_1 , p_2 , y p_3 no tienen el mismo signo, el triángulo determinado por los vectores $\vec{u}$ y $\vec{v}$ corta al plano de $\vec{r}$ y $\vec{s}$.
 - Si p_4 , p_5 , y p_6 no tienen el mismo signo, el triángulo determinado por $\vec{r}$ y $\vec{s}$ corta al plano de $\vec{u}$ y $\vec{v}$.
 - Si ambas condiciones se presentan, los dos triángulos se cortan.
6. Una vez que la existencia del corte es segura, se redefinen los vectores $\vec{r}$, $\vec{s}$, $\vec{u}$, y $\vec{v}$. El cambio de signo en los productos de la ecuación (4.6) decide qué lados de cada triángulo intersectan como líneas al otro. Si p_1 y p_2 tienen signo diferente, entonces DF de TR2 en la Figura 101 en la página 112 corta a TR1. Si hay cambio de signo entre p_2 y p_3 , entonces el lado EF de TR2 corta también a TR1. No pueden aparecer más de dos intersecciones. El mismo esquema se repite con p_3 , p_4 y p_5 para encontrar los cortes de TR1 contra TR2. Los nuevos vectores $\vec{r}$ y $\vec{s}$ son los dos lados de TR1 que cortan a

TR2, con un vértice común en sus orígenes. Los nuevos $\vec{u}$ y $\vec{v}$ son los dos lados de TR2 que cortan a TR1.

7. Ahora se definen cuatro puntos, calculados con los nuevos vectores $\vec{r}, \vec{s}, \vec{u}$ y $\vec{v}$, utilizando las expresiones de la ecuación (4.6). Los puntos calculados a partir de la ecuación (4.4) son las intersecciones de $\vec{r}$ y $\vec{s}$ contra TR2, y los puntos dados por la ecuación (4.5) son las intersecciones entre los vectores $\vec{u}$ y $\vec{v}$ contra TR1. Las coordenadas de los puntos son paramétricas y debe operarse con cuidado de los errores de truncamiento en punto flotante. Los parámetros no pueden tomar valores mayores que 1 y menores que 0, ya que ésta es la condición de existencia de intersección. alguna operación puede dar resultados cerca del cero, pero fuera de rango, o cerca de 1, pero fuera de rango también. Estos valores se corrigen y se les asigna cero y uno respectivamente.
8. Las dos parejas de puntos calculadas con la ecuación (4.4) y la expresión (4.5) representan la misma línea, pero diferentes segmentos. Si los dos triángulos se cortan, los dos segmentos tienen una parte común, que es el segmento de traza intersección deseado. Esta parte común se halla como sigue. En la Figura 102 en la página 113, el vector $\vec{p}$ es el segmento intersección dado por la ecuación (4.4) y el vector $\vec{q}$ es el segmento intersección dado por la ecuación (4.5). Se hacen dos productos escalares y se normalizan, tal como se indica en la ecuación (4.7).

$$\begin{aligned}
 p_1 &= \frac{\vec{p}_1 \cdot \vec{p}}{\vec{p} \cdot \vec{p}} \\
 p_2 &= \frac{\vec{p}_2 \cdot \vec{p}}{\vec{p} \cdot \vec{p}}
 \end{aligned}
 \tag{4.7}$$

9. Una vez realizados los productos escalares, se hace un test de acuerdo a la Figura 103 en la página 114. Las combinaciones sombreadas son imposibles, una vez decidida la existencia de intersección. Los otros cuadrados muestran el segmento de intersección de los dos triángulos, que es el resultado del problema.

5.0 Conclusiones

Se ha revisado la gran mayoría de las técnicas de tratamiento y representación de estructuras gráficas y se ha conseguido la creación de nuevos algoritmos matemáticos para el manejo e intercambio entre imágenes y gráficos en 2-D y 3-D. A la vista de los resultados, las conclusiones más importantes son las siguientes:

1. Se ha conseguido establecer y comprobar el parentesco cercano entre imágenes raster y gráficos vectoriales, como dos formas de representación intercambiables mediante la operación de vectorización y su inversa. Para ello se han realizado varios pasos que comprenden el diseño de:
 - Técnicas específicas de mejora de imagen adecuadas a la vectorización.
 - Un algoritmo de segmentación con umbral dinámico que complementa las técnicas propuestas sobre umbral fijo.
 - Un algoritmo de adelgazamiento que garantiza continuidad en las estructuras, a base de operaciones lógicas, muy rápido.
 - Un extractor de bordes en imágenes segmentadas basado en técnicas de adelgazamiento.
 - Un algoritmo de vectorización propiamente dicha basado en operaciones lógicas, con reducción de casos particulares al caso general, muy rápido y eficiente, fácilmente realizable en hardware, al igual que el conjunto.
 - Un algoritmo de suavizado basado en operaciones vectoriales, que evita los problemas mal condicionados propios de los determinantes en algoritmos convencionales.
2. Se ha desarrollado un sistema de representación de estructuras 3-D para superficies y líneas multiformes, que comprende el diseño de:

- Un algoritmo de líneas ocultas que resuelve penetraciones, autopenetraciones y todo tipo de estructuras no uniformes en coordenadas paramétricas.
 - Un algoritmo que genera las líneas de intersección de superficies multiformes autopenetrantes e interpenetrantes.
 - Un algoritmo de curvas de nivel 3-D para superficies no uniformes.
3. Se hace uso exclusivo de coordenadas paramétricas, y se ha desarrollado un conjunto de ecuaciones y procedimientos vectoriales para cortes entre estructuras, cuyas ventajas más importantes son:
- Eliminación de problemas mal condicionados propios del uso de determinantes.
 - El parámetro es un número comprendido entre 0 y 1.

En resumen, se han expuesto algunas de las técnicas más punteras en el tratamiento de imágenes y gráficos, a la vez que se han propuesto nuevos algoritmos para tratamiento y representación de todo tipo de estructuras gráficas.

6.0 Aplicaciones de los algoritmos generados

La combinación del tratamiento digital de imágenes con la generación de gráficos en el ordenador ha dado sus primeros frutos en los resultados expuestos en (57,58). Consiste esta aplicación en el levantamiento de mapas de energía de anisotropía en imágenes de dominios magnéticos. Estos mapas son previamente calculados con una técnica híbrida entre el tratamiento digital de imágenes y los gráficos por ordenador. Una imagen raster de dominios magnéticos vistos desde su parte superior ("stripe domains") se convierte en gráficos vectoriales 2-D, para crear después una nueva imagen raster de distancias que a su vez se levanta en forma gráfica 3-D. En este proceso están interesados el Laboratorio de Magnetismo de la Facultad de Ciencias Físicas de la Universidad Complutense de Madrid y el Laboratorio de Física de la E.T.S. de Ingenieros de Telecomunicación de la Universidad Politécnica de Madrid.

Los nuevos ensayos en segmentación, adelgazamiento, vectorización y suavizado de estructuras se están abriendo camino con aplicaciones nuevas en sistemas de transmisión de imágenes y gráficos en códigos vectoriales. Estos algoritmos han sido desarrollados dentro de un programa global encaminado a la transmisión de firmas por línea telefónica para el Departamento de Investigación y Desarrollo de Telefónica, y algunos resultados pueden verse en (49,50).

La Facultad de Geografía e Historia de la Universidad Complutense de Madrid y de la Facultad de Letras de la Universidad de Murcia, y más concretamente sus Departamentos de Arqueología y de Arqueología y Prehistoria respectivamente, están interesados en la utilización de estos sistemas de tratamiento de imágenes y generación de gráficos para el reconocimiento de estructuras en piezas que por su antigüedad y estado presentan problemas de reconocimiento a simple vista. Estas colaboraciones ya han tenido fruto en (121,122,124).

Los nuevos algoritmos vectoriales han sido ampliamente utilizados en la creación de un nuevo algoritmo de líneas ocultas al que actualmente no se le ha encontrado ningún fallo. Las ideas expuestas en (92) han sido de gran utilidad en un preproceso. Resuelve este algoritmo todo tipo de penetraciones, generando sus aristas, y trata por igual todas las estructuras del gráfico, independientemente de sus grados de libertad en el espacio en cuestión. Una presentación de él se puede ver en (60). Este algoritmo está siendo utilizado en representación de esquemas de percepción visual, sistemas biológicos microscópicos, y levanta-

tamiento de túmulos arqueológicos con visión de secciones. A continuación se describen algunas de estas aplicaciones con detalle.

6.1 Transmisión de firmas y mapas por línea telefónica

6.1.1 Necesidad de la transmisión de firmas

Un sistema de transmisión de firmas por línea telefónica sería capaz de evitar el gran vacío existente en las transacciones bancarias, donde la firma puede ser una información indispensable. Es, por ejemplo, ampliamente conocido el problema del riesgo en abonos en metálico de cheques y talones en oficinas bancarias distintas de aquélla en la que fue abierta la cuenta deudora. La firma sólo está operativa en un fichero convencional y manual, existente en la sucursal o agencia urbana donde fue abierta la cuenta por el cliente.

Una gran disminución de riesgos en muchas operaciones, lo mismo que la posibilidad de la realización de otras, tendría lugar si se pudiese transmitir una firma desde una gran base de datos en una central contable hasta cualquier terminal convencional situado en sucursal o agencia urbana de la entidad.

6.1.2 Información en firmas y mapas

Una firma no es realmente una imagen binaria, pero sí lo es la información que contiene. Por otra parte, se ha de contar con un procedimiento que aísle la información de la imagen de lo que realmente no lo es. Así, la firma almacenada en las bases de datos debe ser absolutamente independiente del tipo de instrumental empleado para firmar, como rotulador, bolígrafo, etc.... El grueso de los trazos no lleva ninguna información que se utilice en el reconocimiento habitual. Tan sólo el esqueleto de la firma es útil para el reconocimiento de la misma, y, realmente, sólo algunos puntos de este esqueleto son necesarios para su reconocimiento. Por otra parte, el hecho de utilizar sólo el esqueleto hace posible su visión en terminales de baja resolución, lo que de otra forma no sería posible ya que sólo aparecerían manchas amorfas.

En el caso de un mapa ocurre exactamente igual con el esqueleto de los bordes que con el esqueleto de una firma. En este caso se impone una extracción de bordes previa. Estos bordes son adelgazados y vectorizados posteriormente. Si se trata de una firma, basta con hacer estas dos últimas operaciones. Como ya se ha descrito, se necesita realizar una segmentación anterior al adelgazamiento. El extractor de bordes en mapas puede ser orientado a imágenes binarias o multinivel, según se efectúe la operación de

segmentación antes o después de la extracción.

6.1.3 Comparación con el FAX. Factor de compresión

Los factores de compresión que se llegan a alcanzar con el sistema de almacenamiento y transmisión de firmas dependen de la naturaleza de dichas firmas, puesto que son tratamientos fuertemente no lineales. No obstante, se puede decir que respecto a una imagen raster original de 256x256 píxeles, cuantificada a un bit/píxel pueden oscilar alrededor de 0,01.

Una primera forma de solucionar el problema sería con la utilización de un facsímil, pero los inconvenientes respecto al sistema propuesto aquí son enormes y numerosos. Entre ellos se pueden citar los siguientes:

1. Un sistema facsímil necesita un terminal FAX receptor en cada sucursal y agencia urbana de la entidad bancaria, lo cual sería muy caro, cuando en cada sucursal se dispone ya de un terminal de ordenador normal.
2. El terminal FAX es incompatible de una forma directa con un ordenador, debido a que el FAX conmuta su velocidad de transmisión con el tiempo. Esto llevaría a la duplicidad de sistemas, dado que sería tremendamente difícil conectar el ordenador de la central contable al FAX transmisor y el terminal receptor de la sucursal al FAX receptor, todo ello independientemente de la simulación de protocolos.
3. El FAX transmite toda la imagen sin eliminar la información que no es necesaria para el reconocimiento de la firma. Por tanto, transmite más que el esqueleto, con información redundante e inútil para dicho reconocimiento.
4. La firma patrón ante el FAX debe estar en el soporte adecuado. Este soporte puede ser papel o fichas. De esta forma, lo único que se conseguirá es llevar el fichero convencional a la central contable, y un operador debería introducir las fichas manualmente en el FAX ante una petición desde el terminal de la agencia. Por otra parte, este soporte se deteriora fácilmente con el uso y puede llegar a inutilizarse totalmente.
5. El sistema FAX es absolutamente rígido, en el sentido de que sólo admite documentos sobre papel impreso. No se podría pensar en algún otro tipo de entrada al sistema para reconocimiento de firmas grabadas en microfilm, de forma que funcionasen como códigos similares a las tarjetas magnéticas de

cajeros automáticos.

6. El FAX es un sistema poco versátil en el sentido de que cualquier futura aplicación que pudiera hacerse introduciendo algún ordenador personal en sucursales principales se haría difícil y costosa, y, en algunos casos, imposible.
7. El terminal FAX, en definitiva, hace absolutamente inútil la gran capacidad de proceso del ordenador central que usualmente tienen los Bancos en su Central Contable, infrautilizando todas las instalaciones y duplicando servicios.

6.1.4 Adquisición de la firma: imagen y gráfica

La adquisición de mapas suele tener lugar por medio de la digitalización de alguna imagen que los contenga. El caso de la firma puede hacerse en la misma forma o directamente como un gráfico.

Una adquisición de la firma como gráfico, a través de tablilla gráfica, por ejemplo, puede hacerse cuando la firma se adquiere por primera vez, al objeto de su inclusión en una base de datos. Cuando la firma está en un cheque que se quiere comparar con la de la base de datos, entonces procede la digitalización como si fuera una imagen raster.

6.1.5 Tratamiento

Es diferente el tratamiento de una firma dependiendo de su forma de adquisición. Si se adquiere como un gráfico, sólo es necesario suavizar la base de datos, que tiene la forma de una base de datos de vectorización. Si se adquiere como una imagen, debe ser segmentada, adelgazada y vectorizada, antes de aplicar el suavizado.

Un mapa, una vez adquirido, debe someterse a una extracción de bordes y luego a una segmentación. Estas dos operaciones pueden realizarse en cualquier orden, pero según el caso los algoritmos empleados deben ser distintos. En un primer caso de extracción de bordes seguido de segmentación, es necesario aplicar un extractor de bordes de imágenes multinivel y una segmentación con umbral fijo suele ser suficiente. En un segundo caso de segmentación seguida de extracción de bordes, este extractor de bordes debe ser propio de imágenes binarias, y el algoritmo de segmentación podría ser de umbral dinámico. Con el mapa así tratado, los pasos siguientes son adelgazamiento, vectorización y suavizado. El mapa queda así como una base de datos vectorizada.

Figura 104. Firma adquirida por tablilla gráfica digitalizadora

El sistema presentado incluye la posibilidad de volver a la imagen raster binaria correspondiente al gráfico, al objeto de probar técnicas de codificación apropiadas a imágenes binarias.

Figura 105. Firma gráfica con suavizado superpuesto

Figura 106. Firma raster a partir de firma gráfica

Figura 107. Firma adquirida por digitalización de imagen raster

Figura 108. Firma raster con segmentación

Figura 109. Firma raster con adelgazamiento

Figura 110. Firma raster vectorizada

6.1.6 Resultados

En la Figura 104 en la página 123 se puede ver una firma adquirida por tablilla digitalizadora. Un suavizado se muestra superpuesto a la original en la Figura 105 en la página 123, y su imagen raster correspondiente en la Figura 106 en la página 124.

En la Figura 107 en la página 124 se muestra una firma adquirida como imagen raster. La segmentación se ve en la Figura 108 en la página 125, el adelgazamiento en la Figura 109 en la página 125 y una vectorización en la Figura 110. La base de datos formada se describe en la Figura 111 en la página 127. Allí se puede ver los elementos que forman la imagen vectorizada, y su clasificación en ramas, troncos, etc...

Como ya se ha explicado, el sistema debe ser independiente del rotulador empleado para firmar. El grueso del trazo no es información relevante. Un experimento con otra firma, pero con trazo más grueso, se ve a continuación. En la Figura 112 en la página 128 se muestra el original raster. La segmentación se ve en la Figura 113 en la página 128, el adelgazamiento en la Figura 114 en la página 129 y una vectorización en la Figura 115 en la página 129. El trazo grueso provoca la aparición de ramas espúreas, que son eliminadas en Figura 116 en la página 130, merced al conocimiento que se tiene acerca de qué elementos son ramas y cuáles troncos en la base de datos de vectorización. Las ramas cuya longitud es menor que un umbral dado se eliminan. Dicho umbral es determinado por inspección de la longitud me-

Numero de segmentos..... =	716
Numero de lineas..... =	34
Numero medio de segmentos por linea =	21.05882
Desviacion..... =	21.32237
Numero de troncos..... =	26
Tanto por ciento de troncos..... =	3.63128 %
Numero de ramas..... =	8
Tanto por ciento de ramas..... =	1.11732 %
Longitud media de las lineas..... =	56.14706
Desviacion..... =	51.59521

Figura 111. Descripción de la base de datos de la firma vectorizada

dia de las ramas y su desviación, mostrada en el análisis de la base de datos que se ve en la Figura 117 en la página 131.

Figura 112. Firma adquirida por digitalización de imagen raster con trazo grueso

Figura 113. Firma raster de trazo grueso con segmentación

Figura 114. Firma raster de trazo grueso con adelgazamiento

Figura 115. Firma raster de trazo grueso vectorizada

Figura 116. Eliminación de ramas sobre la firma vectorizada de trazo grueso

6.2 Medida de distancias entre dominios magnéticos

6.2.1 Necesidad de la medida de distancias

La medida de distancias entre líneas de una imagen, así como la identificación de las parejas de segmentos más próximos puede ser realizada mediante la aplicación de técnicas de tratamiento digital de imágenes. Los dos segmentos vecinos de uno dado, uno a su derecha y otro a su izquierda, son localizados punto a punto, y se puede crear una nueva imagen superpuesta a la original, llamada mapa de distancias. En ella, los píxeles contienen información sobre la distancia entre los dos segmentos más próximos entre sí en las cercanías del píxel de la imagen original.

A lo largo del proceso aparecen muchos problemas propios sistemas gráficos, que son resueltos aplicando unas veces técnicas convencionales y otras, algoritmos desarrollados "ad hoc". El resultado final es un sistema automático de medida de distancias entre las líneas de la imagen original.

El sistema se diseñó y aplicó para la medida de distancias entre las paredes de dominios magnéticos. Las imágenes originales fueron fotografías obtenidas mediante un procedimiento descrito en (115), a partir de un material magnético amorfo con su eje fácil ortogonal a la superficie de la muestra. La apariencia de estos dominios en la imagen es la de rayas muy oscuras sobre fondo claro semejantes a un laberinto. El

Numero de segmentos.....=	548
Numero de lineas.....=	44
Numero medio de segmentos por linea =	12.45455
Desviacion.....=	10.97405
Numero de troncos.....=	32
Tanto por ciento de troncos.....=	5.83942 %
Numero de ramas.....=	12
Tanto por ciento de ramas.....=	2.18978 %
Longitud media de las lineas.....=	38.18182
Desviacion.....=	42.56774

Figura 117. Especificación de la base de datos de la firma vectorizada de trazo grueso

cálculo de distancias entre las paredes magnéticas da una idea de la distribución local de energía de anisotropía de la muestra.

6.2.2 Necesidad de la automatización del proceso

Anteriormente a la introducción de técnicas de tratamiento digital de imágenes en este campo, las distancias entre dominios eran estimadas manualmente con un micrómetro. Se medían unas pocas distancias y se extrapolaban los resultados a todas las zonas de apariencia similar. Posteriormente se asignaban valores de distancia entre líneas por zonas amplias, y no punto a punto, que es la principal ventaja del procedimiento automático propuesto. A ella hay que añadir el hecho de la disminución de la intervención del hombre en los cálculos, eliminando el riesgo de errores graves e introduciendo una precisión espacial imposible de obtener manualmente.

El sistema se aplicó sobre imágenes de una muestra de material amorfo $Fe_{40} - Ni_{40} - P_{14} - B_6$, sometida sucesivamente a siete grados diferentes de compresión. La secuencia de fotogramas escogida fue la mejor entre las que se tenían, desde el punto de vista de la minimización de artefactos en la película, o, al menos, que estos fueran lo más finos posible.

Otra aplicación del sistema es la creación de un mapa de gradiente de distancias sobre la imagen anteriormente conseguida, a base de aplicar el mismo proceso a la imagen raster correspondiente a las curvas de nivel del mapa de distancias.

6.2.3 Posible adquisición gráfica de los dominios

Los dominios son normalmente adquiridos por imagen raster digitalizada, pero una alternativa válida es introducirlos a partir de dibujos teóricos o prácticos a través de una tablilla digitalizadora de gráficos, ya que para la obtención del mapa de distancias se necesita dicha imagen vectorizada.

6.2.4 Tratamiento de las imágenes raster

El primer paso es convertir las imágenes raster de dominios en gráficos vectoriales. Este paso se realiza tal como se ha indicado, con segmentación, adelgazamiento, vectorización y suavizado. El proceso que se sigue es el que se ha detallado en el capítulo donde se describe la medida de distancias entre líneas en entornos no conexos.

Figura 118. Dominios vectorizados para mapa de distancias

Figura 119. Mapa de distancias o de energía de anisotropía de material magnético en función de sus dominios

Figura 120. Superposición del mapa de distancias a los dominios vectorizados

6.2.5 Resultados

Algunos resultados de la segmentación, adelgazamiento y vectorización de dominios magnéticos ya han sido vistos con los ejemplos que se han ido exponiendo junto con los algoritmos correspondientes. Se puede ver un mapa de distancias sobre la imagen vectorizada de la Figura 118 en la página 133 en la Figura 119 en la página 133, y la superposición del mapa a la vectorización en la Figura 120.

6.3 Tratamiento de imágenes arqueológicas

6.3.1 Necesidad del tratamiento de imágenes arqueológicas

Una de las aplicaciones de los algoritmos de mejora de imagen presentados es la Investigación Arqueológica. Se exponen a continuación las causas que hacen útil la aplicación de técnicas de tratamiento digital de imagen y gráficos por ordenador en Arqueología.

Un primer motivo es el modo de extracción de información por parte del arqueólogo, basado en obtención de fotografías en viajes y excavaciones. La mayoría de estos viajes y excavaciones son irrepetibles, siendo algunos objetos encontrados en éstas últimas irrecuperables. Entre todas las fotografías que se hagan, puede darse la mala fortuna de que alguna pieza clave haya sido tomada con exposición incorrecta, de forma que es imposible ver nada en ella, aún con proceso químico fotográfico convencional.

Un segundo motivo es el estado lamentable en que se encuentran muchas piezas arqueológicas, debido al paso del tiempo, a la labor humana de rapiña y destrozo en invasiones y guerras, y a los agentes naturales, como el viento, el agua y su erosión consiguiente.

El tratamiento digital de imágenes proporciona excelentes resultados en el caso de mejora de fotografías incorrectamente expuestas, y es capaz de realzar todo tipo de detalles en las imágenes de piezas deterioradas, de forma que un observador experto (Arqueólogo) es capaz de extraer importantes conclusiones. La aplicación de operadores sobre imágenes que llevan a otros dominios matemáticos, tales como las transformadas, la obtención de patrones, el agrupamiento, etc..., puede obtener clasificaciones automáticas de objetos arqueológicos sin demasiado esfuerzo por parte del experto. Por último, ante un objeto deteriorado a clasificar, sus patrones o sus espectros pueden hacer rápidamente una correcta clasificación, que de otra forma sería tediosa y con bastante probabilidad de error. De esta forma, se perfila el tratamiento digital de imágenes como herramientas muy útiles de análisis de las características básicas de los elementos arqueológicos, al ser capaz de aislar los patrones principales deseados en todas las estructuras a analizar y estudiar.

Otra aplicación que está ahora en estudio es la representación tridimensional de estructuras compuestas, tales como túmulos, que pueden generarse como conjuntos de primitivas. Un ejemplo de primitiva puede ser un determinado tipo de sarcófago, que se puede componer con otros elementos también definidos por primitivas para formar un dibujo completo. El sistema debe permitir el estudio de secciones sin necesidad de volver a dibujar el túmulo manualmente cada vez, tal y como se hace en la actualidad. Esta

Figura 121. Elemento arqueológico original

aplicación se está realizando a partir de los algoritmos de representación 3-D que aquí se han expuesto. De la misma forma, el algoritmo de curvas de nivel generado se está utilizando para el levantamiento 3-D de campos de excavación y para convertir en dibujos que puedan ser sacados por un trazador de curvas muchos objetos fotografiados en trabajos de campo.

Por otra parte, el profesional de la Arqueología necesita muchas veces la realización de dibujos a partir de fotografías y el estudio de secciones de estructuras compuestas. Las técnicas de gráficos por ordenador son capaces de ayudar al arqueólogo en estas tareas, a base de la extracción de curvas de nivel sobre fotografías y la representación 3-D de conjuntos de objetos que puede contemplar desde diversos ángulos y con diferentes cortes sobre las estructuras.

Figura 122. Histograma del elemento arqueológico original

6.3.2 Adquisición de las imágenes

Las imágenes fuente para el proceso son obtenidas a partir de fotografías tomadas de piezas arqueológicas. Estas imágenes son volcadas a una cinta magnética, tras un proceso de adquisición en un microdensitómetro de lecho plano, o una cámara con sistema de digitalización de imágenes en tiempo real.

En el caso de un microdensitómetro de lecho plano, es necesario elegir la ventana y el período de muestreo. Ventanas cuadradas del orden de 80 a 100 μm de lado son ideales acompañadas de un período de muestreo de 80 μm en cada dimensión. Un negativo o diapositiva de 24x36 mm es convertido en una imagen raster de 512x768 píxeles, con óptima definición, que se usa en el proceso definitivo, aunque para las pruebas basta una imagen de 256x384 píxeles, que hace los procesos significativamente más rápidos.

En el caso de una adquisición por cámara, el sistema empleado tiene la posibilidad de adquisición de imágenes cuadradas de 512x512 píxeles, haciéndose el proceso de prueba con un subsector o una reducción previa, en orden a establecer los caminos a seguir en el tratamiento de la imagen definitiva con la resolución máxima, con una mayor rapidez.

Figura 123. Elemento arqueológico con corrección de máxima entropía local y ventana muy pequeña

6.3.3 Elementos empleados

Las técnicas de tratamiento digital de imágenes que se muestran como óptimas para este proceso son las siguientes:

1. Realce de bordes
2. Extracción de bordes
3. Segmentación
4. Adelgazamiento
5. Vectorización

Figura 124. Elemento arqueológico con corrección de máxima entropía local, ventana de 27x27

6. Correcciones entrópicas
7. Correcciones logarítmicas
8. Correcciones exponenciales
9. Interpolación

De esta lista de procedimientos, la mayoría han sido ya explicados aquí con detalle. Las imágenes que se obtienen de su aplicación se muestran a continuación. Posteriormente se mostrarán el realce de bordes, las correcciones logarítmicas y las correcciones exponenciales, que aún no han sido aplicados.

Sea la imagen original Figura 121 en la página 136. La fotografía muestra gradientes de iluminación que hacen difícil cualquier proceso posterior. La Figura 122 en la página 137 muestra su histograma. La

Figura 125. Histograma de la corrección de máxima entropía local con ventana muy pequeña

corrección global de histograma buscando uniformidad del mismo (10,13), no da resultados en absoluto

Figura 126. Histograma de la corrección de máxima entropía local, ventana de 27x27

Figura 127. Tres valores de factor de realce: bajo, correcto y alto

satisfactorios, siendo éste un caso que requiere el tratamiento con corrección local. En la Figura 123 en la página 138 y en la Figura 124 en la página 139 pueden verse dos ejemplos de ventanas distintas. La primera es pequeña, y la segunda es más correcta. En la Figura 125 en la página 140 y en la Figura 126

Figura 128. Elemento arqueológico con factor de realce igual a 2,0

Figura 129. Realce de bordes con factor 2,3 sobre la corrección de máxima entropía local ventana de 27x27.

en la página 140 se ven los histogramas de las dos respectivas correcciones de umbral dinámico de la Figura 123 en la página 138, y de la Figura 124 en la página 139.

6.3.3.1 Realce de bordes

La información de una imagen se ve bastante deteriorada por el hecho de degradar sus bordes, ya que el ojo es sensible a ellos de forma muy acusada. Si se consiguen realzar dichos bordes, por el contrario, se logran ver aspectos y formas en la imagen que antes era imposible contemplar.

Existen en la bibliografía diversas técnicas de realce de bordes, pero la que mejores resultados muestra en la mayoría de las aplicaciones es la fórmula propuesta por Wallis (9,13):

$$G_{jk} = [F_{jk} - \hat{F}_{jk}] \left[\frac{A\sigma_d}{A\sigma_{jk} + \sigma_d} \right] + [\alpha m_d (1 - \alpha) \hat{F}_{jk}] \quad (6.1)$$

Figura 130. Corrección de máxima entropía global sobre realce hecho a la corrección local (27x27).

Esta fórmula, que transforma el valor de un elemento de la original F_{jk} en otro valor G_{jk} , tiene la ventaja de poseer muchos parámetros libres y seleccionables por el usuario, a saber:

1. Tamaño de la ventana para el filtro
2. Relación fondo/borde, α
3. Luminancia media, m_d
4. Desviación standard de la luminancia, σ_d
5. Factor de realce, A

La representación $\hat{F}_{jk}$ corresponde a un elemento de otra imagen cuyo valor es el promedio de las

Figura 131. Realce de bordes sobre las correcciones anteriores

luminancias de los píxeles de la ventana que tiene como píxel central en la imagen original al elemento F_{jk} . La representación σ_{jk} corresponde también a un elemento de otra imagen cuyo valor es la desviación standard de la luminancia en la misma ventana. Respecto al tamaño de la ventana, un cuadrado de 5×5 píxeles se perfila como la que mejores resultados da en este tipo de imágenes, aunque en general depende del tamaño del borde del detalle que se desee realzar. La mejor relación fondo/borde es 0,1. La óptima luminancia media depende de las características concretas de exposición de la fotografía. Si éstas son adecuadamente corregidas en un primer paso de correcciones entrópicas de histograma, es recomendable una luminancia media de nivel 128 en una imagen de digitalizada con 8 bits/píxel, con nivel de blanco igual a 255. La desviación standard depende de las características que se desee resaltar. Valores normales oscilan entre 30 y 80, siendo éste último el que responde a un realce de propósito general.

El parámetro más importante es el factor de realce. Un factor de realce bajo no cumple el propósito deseado, y su resultado se puede ver en la Figura 127 en la página 141, en un ejemplo de una sola variable en el plano espacio-luminancia. Un realce óptimo se ve en la misma Figura 127 en la página 141, y

Figura 132. Corrección logarítmica del elemento arqueológico con realce de bordes

el resultado de un realce excesivo provoca un fenómeno de rizado, también mostrado en la Figura 127 en la página 141. Dado que no todos los bordes son iguales, es necesario encontrar un valor de compromiso, dependiendo de qué se desee hacer. Para aumentar la calidad fotográfica de una imagen arqueológica, se recomiendan valores para el factor de realce entre 1,2 y 2,3. Para un realce que simplemente permita estudiar alguna característica relevante, sin importar excesivamente la calidad fotográfica, se han utilizado valores entre 2,3 y 6,0. Si lo que se pretende es estudiar un detalle muy determinado sin importar la introducción de ruido en toda la imagen, son valores recomendables entre 6,0 y 20,0.

Se pueden ver resultados que combinan los métodos descritos hasta el momento. En la Figura 128 en la página 141 se ve un realce con factor 2,0 sobre el original de la Figura 121 en la página 136. En la Figura 129 en la página 142 se muestra un realce de bordes con factor 2,3 sobre la corrección local de histograma con ventana de 27×27 , y en la Figura 130 en la página 143 se puede ver una corrección global de máxima entropía sobre el realce anterior. En la Figura 131 en la página 144 se muestra un nuevo realce de bordes sobre la última corrección, con factor 2,0.

Figura 133. Realce de bordes con factor 3,2 sobre la corrección no lineal con $k=1,5$

6.3.3.2 Correcciones logarítmicas

Hay veces que la imagen original tiene tal grado de deterioro que la única solución es formar la imagen con una corrección logarítmica, aun a costa de perder algo de información. El rango dinámico de la imagen se adecúa antes y después de la corrección al rango dinámico que muestra el terminal de imágenes. La operación de corrección logarítmica se puede ver en (10), donde cada elemento de imagen e_{ij} se transforma en otro elemento con valor nuevo de luminancia n_{ij} según la ecuación:

$$n_{ij} = \log_{10} [1 + e_{ij}] \quad (6.2)$$

Un realce de bordes con un factor pequeño sobre una corrección logarítmica aplicada a la imagen original de la Figura 121 en la página 136. mejora algo la calidad de los detalles suavizados por dicho tratamiento y se muestra en la Figura 132 en la página 145.

Figura 134. Realce de bordes con factor 3,2 sobre la corrección no lineal con $k=2,3$

6.3.3.3 Correcciones exponenciales

En algunos casos en que los originales están muy deteriorados, sea por sus condiciones reales o por las condiciones de exposición de la fotografía original, las correcciones logarítmicas son un último recurso que a veces no es necesario emplear. Las correcciones exponenciales son un paso con resultados intermedios muy interesantes, que son los mejores en el caso de la imagen arqueológica que aquí se trata.

Este tipo de correcciones consiste en adecuar el rango dinámico de la imagen al intervalo entre 0 y 1. En estas condiciones, y dado un valor real k , normalmente mayor que 1,0, se aplica a cada elemento de imagen e_{ij} una transformación para convertirlo a otro valor de luminancia n_{ij} según la ecuación:

$$n_{ij} = (e_{ij})^{\frac{1}{k}} \quad (6.3)$$

A continuación, se lleva el rango dinámico de la imagen de nuevo al intervalo entre 0 y 255 para su

Figura 135. Realce de bordes con factor 3,2 sobre la corrección no lineal con $k = 3,2$

presentación. En la Figura 133 en la página 146, en la Figura 134 y en la Figura 135 en la página 148 pueden verse los resultados de un realce de bordes con factor 3,2 sobre la aplicación de la expresión (6.3) con valores k respectivamente iguales a 1,5, 2,3 y 3,2.

6.3.3.4 Interpolación

A veces es de particular interés el estudio de un subsector de la imagen. El subsector tiene dimensiones menores que el original. Una interpolación por vecino para adecuarlo a las dimensiones del terminal provoca grandes efectos "ladrillo". Se han ensayado técnicas de interpolación por splines y por convolución cúbica. Los splines son más costosos en tiempo de cálculo, pero son los mejores resultados que se obtienen. No obstante, en muchas ocasiones, los resultados de la convolución cúbica son muy buenos y suficientes para la aplicación en cuestión. Algunas referencias se pueden encontrar en (14).

6.3.4 Resultados

Los resultados de las técnicas de tratamiento digital aplicadas se han ido mostrando en figuras según se explicaba el correspondiente algoritmo. Se remite por tanto al lector a las figuras mencionadas en esos apartados.

Las figuras mostradas son una representación iconográfica visigoda de las pocas que existen. El tratamiento aplicado permite contemplar algunas características que inclinan la balanza en favor de las tesis que defienden que se trata de un rey visigodo (113), en contra de las que señalan que se trata de un santo o personaje religioso (114). Si se confirma que efectivamente se trata de un rey, podría ser la única representación iconográfica de un monarca visigodo encontrada en España. La pieza es de gran importancia y se encuentra descrita con detalle en (117). Los resultados del tratamiento digital de este elemento arqueológico ya han dado sus frutos en (121,122).

6.4 Otras aplicaciones

Numerosas aplicaciones están surgiendo a la luz de estos sistemas de tratamiento y representación de estructuras gráficas.

Un campo que ha hecho uso del sistema de representación 3-D ha sido el de las microondas. En la Figura 136 en la página 150 y en la Figura 137 en la página 151 se pueden ver sendos gráficos que estudian la posibilidad y existencia de oscilación, así como su estabilidad y ruido de primer orden, en un sistema no lineal con FET realimentado con un resonador dieléctrico. Dicho oscilador se muestra en los esquemas que acompañan a las figuras. El diseño se ha hecho basado en un modelo de gran señal aplicado a un sistema de N-puertas y analíticamente se ha resuelto con técnicas descritas en (21,119).

El estudio de campos electromagnéticos se ve simplificado con la utilización de gráficos 3-D. En la Figura 78 en la página 81 ya se ha mostrado la envolvente del campo eléctrico de una onda electromagnética modulada en amplitud y propagándose en el vacío.

Otros campos tales como la biología molecular, están empezando a hacer representaciones 3-D con los sistemas que aquí se proponen. Algún antecedente de este tipo de representación puede verse en (118,123).

Otra aplicación muy interesante de los algoritmos para tratamiento de estructuras 2-D que aquí se

Figura 136. Aplicación del sistema 3-D a un diseño de oscilador de microondas

exponen es el reconocimiento de contadores de luz y teléfono. En el segundo caso, el procedimiento actual consiste en tomar fotografías del cuarto de contadores y luego leer manualmente dichas fotografías, apuntando los números. El adelgazamiento y la vectorización de los números pueden hacer posible su reconocimiento automático, que, de otra forma, dado su tamaño pequeño en la fotografía, sería difícil de realizar. La salida sería un listado por impresora con todas las cantidades de los contadores. Este camino ya se ha iniciado en la fecha actual.

Una aplicación muy recientemente comenzada consiste en la aplicación de la combinación de los algoritmos aquí expuestos 2-D y 3-D en el tratamiento de fotografías de satélites Meteosat en colaboración con varias empresas, al objeto de poder hacer predicciones sobre el estado de la atmósfera en áreas reducidas y con mayor fiabilidad.

Figura 137. Aplicación del sistema 3-D a un segundo diseño de oscilador de microondas

7.0 Líneas futuras de investigación

Muchos son los caminos que pueden seguirse arrancando desde aspectos muy concretos que aquí se apuntan. Se sugieren a continuación algunos de ellos.

1. Sobre los algoritmos y sistemas gráficos aquí diseñados se pueden establecer las siguientes líneas:
 - Los algoritmos de adelgazamiento, vectorización y suavizado, así como la obtención de imagen raster a partir de gráfico vectorial, admiten una rápida generalización a 3-D. En el caso de la vectorización, las máscaras correspondientes a los casos particulares serían las mismas de 2-D, dado que son máscaras elementales, pero en mayor cantidad, dada la nueva dimensión. La gran ordenación del algoritmo propuesto de vectorización hace posible su generalización sin excesivos cambios. El algoritmo de suavizado está basado en operaciones vectoriales. El paso de 2-D a 3-D requiere sustituir los productos escalares 2-D por productos escalares 3-D, los productos cruzados 2-D por productos mixtos 3-D, y los vectores que definen segmentos 2-D por productos vectoriales 3-D representantes de estructuras definidas por dos segmentos 3-D.
 - La aplicación de los algoritmos de vectorización sobre imágenes que contengan caracteres alfanuméricos puede abrir nuevas facetas en el camino de reconocedores de este tipo de caracteres, de los que ya existe algún modelo simple sobre gráficos vectoriales (2,112) y otros sobre imágenes raster (108,109,110,111).
 - Se ha logrado crear un sistema de representación 3-D para estructuras gráficas en coordenadas paramétricas. El centro del sistema ha sido el algoritmo de líneas ocultas. Se pueden introducir algunas mejoras en las entradas al sistema con tratamientos de interpolación. Se ha escogido un modelado mixto de sólidos entre el modelo de alambre y el conjunto de parches. Queda la puerta abierta a muchas otras formas de modelado, ya que nadie ha dicho la última palabra al respecto. Incluso hay experimentos muy recientes en (81). El paso siguiente para dar mayores prestaciones al sistema de representación 3-D propuesto es el rellenado de parches con modelos de iluminación y sombras, al objeto de dar mayor realismo a los gráficos obtenidos. También queda, pues, la puerta abierta, para adaptar algún modelo de iluminación y sombras existente en la bibliografía

(94,95,96,97,99,100,101,102,103,104,105,106,107) al modelado mixto aquí propuesto, o crear alguno nuevo dentro de la filosofía del conjunto del sistema de representación 3-D.

2. Sobre las aplicaciones realizadas de los algoritmos de tratamiento de imágenes y generación de gráficos se puede profundizar a la vista de los resultados obtenidos:

- En el campo de la Arqueología el adelgazamiento y la vectorización aplicados con los parámetros adecuados pueden ser herramientas que sustituyan la penosa labor de dibujo de piezas por parte profesional de este campo. La representación tridimensional puede facilitar enormemente el estudio de conjuntos de elementos arqueológicos al permitir el estudio de secciones y almacenar estructuras básicas como primitivas de dibujo en bases de datos. Asimismo, la generación de curvas de nivel a partir de fotografías puede convertir en dibujos muchos elementos arqueológicos, facilitando la labor de investigación y los ficheros de los profesionales. Una profundización en las herramientas desarrolladas hacia un sistema de clasificación automática de estilos puede también facilitar enormemente el trabajo de un arqueólogo, a la vez que conseguir mayor exactitud en dichas clasificaciones.
- En el campo de la Física el estudio de los "stripe domains" y los mapas de energía de anisotropía pueden llegar a proporcionar conocimientos más profundos sobre las estructuras amorfas.
- En el terreno de la transmisión de firmas, el sistema puede llevarse a la práctica mediante su aplicación sobre terminales sencillos "ad hoc" con el diseño del hardware adecuado.
- En el campo de la Meteorología se ha comenzado a aplicar alguno de los algoritmos aquí expuestos, y puede ser un camino de investigación de nuevas herramientas según las necesidades de los meteorólogos.
- En Biología, el estudio de las secciones de un virus en 3-D con este nuevo sistema propuesto puede proporcionar conclusiones importantes en este campo, sobre todo dada su facilidad para el cálculo de intersecciones entre estructuras.

Apéndice A. Sistemas de adquisición de imágenes y gráficos

A.1 Adquisición de imágenes con cámara

Un sistema de adquisición de imágenes por cámara debe constar de las siguientes partes:

1. Cámara
2. Procesador de imagen
3. Controlador

A.1.1 La cámara

Un tubo de cámara capta la imagen, que puede ser directamente un objeto real, o una fotografía, diapositiva o negativo. Dependiendo de la fuente de imagen, se debe colocar la óptica adecuada delante del tubo de cámara. Un vidicon proporciona excelentes resultados, sobre todo en imágenes fijas. La cámara puede ser monocroma o en color.

A.1.2 El procesador de imagen

El sistema procesador de imagen ha de constar de las siguientes opciones básicas:

1. Adquisición de imagen
2. Almacenamiento de imagen
3. Presentación de imagen
4. Otras opciones

A.1.2.1 Adquisición de imagen

La imagen captada por el tubo de cámara es digitalizada a 8 bits/píxel con un sistema de muestreo y retención gobernado por un reloj. El reloj debe tener una frecuencia del orden de 10-14 MHz, y de esta forma la adquisición de imágenes tendrá lugar en tiempo real. Puede incorporarse una tabla de consulta de vídeo (VLT-Video Look-up Table) en este paso, de forma que la imagen adquirida sea corregida si es necesario.

A.1.2.2 Almacenamiento de imagen

La imagen, una vez adquirida, es volcada en RAM, al objeto de tenerla disponible para su proceso posterior.

A.1.2.3 Presentación de imagen

El sistema debe contar con un monitor de TV capaz de mostrar la imagen adquirida y residente en memoria, al objeto de comprobar si la imagen que se tiene es la deseada y está en perfectas condiciones de luz y enfoque. Asimismo, sirve para monitorizar la vista de la cámara hasta lograr la imagen que se desee adquirir.

A.1.2.4 Otras opciones

Un sistema de este estilo debe ser abierto. Se pueden incorporar otras muchas opciones, siendo la más relevante la que permite un tratamiento del color en tiempo real. No obstante, tratándose de imágenes fijas, este problema se puede resolver digitalizando sucesivamente tres imágenes iguales con los filtros rojo, verde y azul (R,G,B) correspondientes a los colores primarios teóricos o los prácticos de TVC.

A.1.3 El controlador

El controlador sirve para gobernar las otras dos partes principales: el procesador y la cámara. Ha de constar al menos de las siguientes partes:

1. CPU
2. RAM configurable
3. Periféricos

4. Controlador de periféricos

A.1.3.1 CPU

La CPU es el corazón del sistema, y debe gobernar el conjunto de tareas que realice el bloque completo.

A.1.3.2 RAM configurable

Los programas y tareas que al ejecutarse por la CPU gobiernan el sistema deben residir en memoria. Este es el objeto de la RAM configurable, que debe ser así para evitar que las posiciones de memoria se solapen con las que guardan la imagen.

A.1.3.3 Periféricos

Es necesario un sistema de almacenamiento masivo para los programas y las imágenes. Cualquier disco flexible, disco duro o cinta magnética son válidos.

Por otra parte, la comunicación de usuario con el sistema ha de tener lugar a través de un terminal de pantalla o teletipo.

A.1.3.4 Controlador de periféricos

Se impone la existencia de un controlador de periféricos, que, además de controlar los ya mencionados, debe contar con una salida RS-232 que permita volcar las imágenes a otros ordenadores capaces de hacer el tratamiento digital de la imagen.

Sobre este sistema se incluye un diagrama de bloques en la Figura 138 en la página 158.

A.2 Adquisición de imágenes por microdensitómetro

La segunda forma de adquisición de imágenes es el microdensitómetro. Proporciona una calidad mejor que la cámara, pero es mucho más lento, pues va sobre la base de un sistema mecánico servocontrolado, con dos motores dispuestos de tal forma que facilitan el movimiento de la plataforma del microdensitómetro en dos direcciones ortogonales.

La fuente de imagen ha de ser un negativo, diapositiva, o algo transparente, ya que el

Figura 138. Adquisición de imágenes raster por cámara

microdensitómetro analiza la transmitancia luminosa. Dicha fuente se coloca sobre cristales perfectamente pulidos, y unos fotodiodos enfrentados a un emisor de luz van tomando muestras de la luminancia de los píxeles de la imagen a medida que los motores mueven la plataforma con los cristales en las direcciones X-Y.

El sistema óptico del microdensitómetro incluye una serie de filtros de color, análogos a los descritos para la cámara, y cuyo uso se justifica ante la digitalización en color de una fuente de imagen. También permite el sistema óptico la elección de la ventana de muestreo.

Figura 139. Función típica de distribución de luminancia en una imagen

El sistema completo va gobernado por una CPU, en la que se programan las posiciones iniciales X-Y de la imagen y el período de muestreo, como datos más relevantes. No se emplea RAM intermedia, sino que la imagen se va volcando directamente a cinta magnética fila por fila, en forma de una fila igual a un registro, una imagen igual a un fichero.

La imagen suele salir codificada a 12 bits/píxel, por lo que en la mayoría de las ocasiones es necesaria una conversión a 8 bits/píxel, que es el formato más comúnmente adoptado. Esta transformación se hace en virtud de las propiedades de la función de distribución de la luminancia de una imagen. Esta función sube muy suavemente hasta cierto valor umbral bajo de luminancia, y luego sube bruscamente a lo largo de los niveles de luminancia intermedios, para bajar de nuevo la pendiente a partir de cierto nivel umbral alto de luminancia. Por debajo del umbral bajo apenas queda un 0.1% de píxeles, y lo mismo ocurre por encima del nivel umbral alto. De esta forma, el margen dinámico de luminancias que realmente da contraste a la imagen es menor que el original, y no se degrada la calidad apreciablemente por bajar de 12 bits/píxel. Se puede ver una gráfica esquematizada de la función típica de distribución de luminancia en la Figura 139. Algún detalle más sobre este punto puede contemplarse en (57).

Una descripción más detallada del microdensitómetro puede verse en (116). Un esquema del camino óptico del microdensitómetro se ve en la Figura 140 en la página 160.

Figura 140. Camino óptico de un microdensitómetro

A.3 Adquisición de gráficos por tablilla digitalizadora

Son varios los sistemas de adquisición de gráficos que existen en el mercado. En este apartado se mencionará el funcionamiento de la tablilla gráfica empleada. Alguna referencia a otros sistemas se puede ver en (7).

La tablilla gráfica lleva un bolígrafo conectado a un terminal de imágenes-gráficos. El terminal reproduce en pantalla el dibujo que se realiza sobre la tablilla con el bolígrafo. El gráfico es muestreado a una frecuencia gobernada por secuencias de escape enviadas desde un ordenador, donde se decodifican las coordenadas de cada punto y se almacena el gráfico.

Apéndice B. Sistemas de presentación y tratamiento digital de imágenes

En este apartado se va a hacer una relación de los medios utilizados para llevar a cabo los algoritmos expuestos. Los sistemas que se nombran son un modelo a seguir, pero, naturalmente, no son los únicos existentes.

Los algoritmos propuestos han sido realizados sobre dos sistemas diferentes.

1. El tratamiento de estructuras 2-D se ha llevado a cabo sobre un sistema VAX785 bajo VMS. El terminal de imágenes-gráficos es un Tektronix 4115-B. Los lenguajes de programación utilizados son PL/I y FORTRAN.
2. El tratamiento de estructuras 3-D se ha realizado sobre un IBM 4381, con un sistema de máquinas virtuales bajo CMS. Los terminales de imágenes son Ramtek 9351 y Hacienda 7352. El terminal gráfico es en este caso un Tektronix 612 con Graphic Attachment. El trazador de curvas utilizado para las figuras es un Benson 1302. El lenguaje de programación utilizado es PL/I.

Las imágenes han sido digitalizadas en un microdensitómetro Perkin-Elmer 1010-A de lecho plano. El intervalo de muestreo fue de $80\text{ }\mu\text{m}$ y la ventana de $100\text{ }\mu\text{m}$.

El sistema de adquisición de gráficos empleado ha sido una tablilla gráfica 4010 de Tektronix, soportada por el mencionado terminal 4115-B.

Apéndice C. Presentación de algunos sistemas prácticos propuestos

En este apéndice se expondrán conjuntos de menús que se utilizan para hacer las operaciones de entrada y salida de datos y la selección de opciones en los sistemas prácticos que se han desarrollado sobre los algoritmos propuestos.

C.1 Sistema propuesto de operación raster-vector

TRATAMIENTO DE ESTRUCTURAS LINEALIZABLES

=====

SALIDA _____	0
OPERACIONES DE ENTRADA/SALIDA _____	1
MONITOR DE IMAGENES _____	2
MONITOR DE VECTORES _____	3
BORRADO DE PANTALLA _____	4
DEFINICION DE SUBSECTORES _____	5
SEGMENTACION CON UMBRAL FIJO _____	6
SEGMENTACION DINAMICA _____	7
EXTRACCION DE BORDES _____	8
ADELGAZAMIENTO _____	9
VECTORIZACION _____	10
RECuento DE VECTORES _____	11
VISION EN FALSO COLOR _____	12
RETORNO AL COLOR ORIGINAL _____	13
CAMBIO DE TONO EN LA IMAGEN _____	14
ELIMINACION DE RAMAS _____	15
ELIMINACION DE LINEAS SUELTAS _____	16
MONITOR DE VT220, ADM36, VT100 _____	17

OPCION ELEGIDA ?

Figura 141. Operación de vectorización y su inversa

C.2 Sistema propuesto de mejora de imagen

MEJORA DE IMAGENES

=====

SALIDA	0
OPERACIONES DE ENTRADA/SALIDA	1
MONITOR DE IMAGENES	2
BORRADO DE PANTALLA	3
VISION EN FALSO COLOR	4
RETORNO AL COLOR ORIGINAL	5
CALCULO DE ENTROPIA/HISTOGRAMA	6
MODIFICACION DE HISTOGRAMA	7
MODIFICACION POR ZONAS	8
REALCE DE BORDES	9
EXTRACCION DE BORDES (SOBEL)	10
EXTRACCION DE BORDES (LAPLAC)	11
DIBUJO DEL HISTOGRAMA	12
REDUCCION DE IMAGENES	13
GUARDAR EN MEMORIA LA IMAGEN	14
RECUPERAR LA IMAGEN GUARDADA	15
IMAGEN EN DENSIDAD	16
CORRECCION EXPONENCIAL	17

OPCION ELEGIDA ?

Figura 142. Opciones generales

Tipo de tratamiento

1 = Uniforme

2 = Exponencial

3 = Rayleigh

4 = Hiperbolico (cubico)

5 = Hiperbolico (Logaritmico)

6 = Logaritmico sobre histograma

7 = Logaritmico sobre imagen

8 = Correccion de seno sobre el 1

9 = Correccion de seno sobre el 6

Figura 143. Opciones de tratamiento global de histograma

REALCE DE BORDES

=====

SALIDA _____	0
PARAMETROS PREVIOS _____	1
PARAMETROS DE REALCE _____	2
MONITOR _____	3
VISION EN FALSO COLOR _____	4
RETORNO AL COLOR ORIGINAL _____	5
BORRADO DE PANTALLA _____	6

OPCION ELEGIDA ?

Figura 144. Opciones de realce de bordes

EXTRACCION DE BORDES

=====

SALIDA	0
CALCULO DE PARAMETROS	1
TIPO DE PRESENTACION	2
MONITOR	3
VISION EN FALSO COLOR	4
RETORNO AL COLOR ORIGINAL	5
BORRADO DE PANTALLA	6
SEGMENTACION	7
NEGATIVO	8
CAMBIO DE ESCALA	9

OPCION ELEGIDA ?

PRESENTACION DE BORDES

=====

SALIDA	0
IMAGEN DE GRADIENTE	1
IMAGEN DE GRADIENTE/FONDO REAL	2
BORDES SOBRE FONDO REAL	3
IMAGEN DE GRADIENTE/FONDO NEGRO	4

OPCION ELEGIDA ?

Figura 145. Opciones de extracción y bordes y su presentación

C.3 Sistema propuesto de tratamiento de firmas adquiridas como gráficos

ADQUISICION DE FIRMAS	
=====	
SALIDA _____	0
ADQUISICION DE LA FIRMA _____	1
DIBUJO DE LA FIRMA _____	2
VISION POR TERMINAL CORRIENTE _____	3
SUAVIZADO _____	4
BORRADO DE PANTALLA _____	5
RECuento _____	6
FORMACION DE IMAGEN RASTER _____	7
SALIDA DE IMAGENES _____	8
MONITOR DE IMAGENES _____	9

OPCION ELEGIDA ?

Figura 146. Opciones de manejo de firmas como gráficos y su conversión a imágenes raster

C.4 Sistema propuesto de búsqueda de líneas cercanas y formación de mapas de distancias

```
RELLENADO DE IMAGENES DE RAYAS
*****

NO HAY UNIVERSO

Identificacion del fichero de entrada

Nombre =====> FICHERO
Tipo      =====> ENTRADA
Modo      =====> J1

Terminal

Ramtek     =====> - 1
Hacienda   =====> - 2 . Opcion ==> 1

Numero de puntos de la imagen
Horizontal ==> 512
Vertical   ==> 512

Operaciones

* Salida..... - PF3 -
* Rellenado en colores..... - PF4 -
* Rellenado en blanco y negro..... - PF5 -
```

Figura 147. Selección de imagen de entrada

TRATAMIENTO	PREVIO

Numero de lineas antes de la aproximacion.....==>	525
Numero de lineas despues de la aproximacion.....==>	491
Numero de segmentos antes de la aproximacion.....==>	1123
Numero de segmentos despues de la aproximacion.....==>	997

Umbrales

=====

Umbral de angulo para el aplanamiento.....==>	30.0
Umbral de longitud para el aplanamiento.....==>	0.25
Umbral de direccion para la separacion.....==>	10.0
Umbral de longitud para desaparicion.....==>	6.32

OPERACIONES

* Recuperacion del original.....	- PF2 -
* Fin de tratamiento previo.....	- PF3 -
* Aplanamiento.....	- PF4 -
* Separacion.....	- PF5 -
* Limpieza.....	- PF6 -
* Displayar lineas.....	- PF7 -
* Borrar.....	- PF8 -
* Actualizar umbral limpieza.....	- PF9 -

Figura 148. Tratamiento previo

Figura 149. Parámetros de recorrido

R E C O R R I D O

Numero de lineas de la imagen.....	==>	525
Numero de linea patron.....	==>	17
Numero de linea encontrada.....	==>	215
Numero de linea anterior.....	==>	215
Angulo de radiacion actual.....	==>	40.0
Numero de pasada con ese angulo.....	==>	2
Angulo inicial de radiacion.....	==>	10.0
Angulo final de radiacion.....	==>	80.0

Figura 150. Actualización de recorrido

PARAMETROS DE RESULTADO

Distancia maxima lineal.....==>	201.45
Distancia minima lineal.....==>	1.04
Distancia maxima puntual.....==>	204.32
Distancia minima puntual.....==>	2.02
Pendiente de transformacion...==>	54.27
Ordenada de transformacion....==>	-3.87
Numero de niveles histograma..==>	256

Operaciones	*	Recuperacion de la imagen inicial.....	- PF1 -
	*	Volver a continuar recorrido.....	- PF2 -
	*	Volver a recorrido nuevo.....	- PF3 -
	*	Mapa en grises con histograma.....	- PF4 -
	*	Mapa en colores con histograma.....	- PF5 -
	*	Carta de tonos.....	- PF6 -
	*	Reponer lineas.....	- PF7 -
	*	Poner marco.....	- PF8 -
	*	Borrar rayas y marco.....	- PF9 -
	*	Reinicializacion con otra imagen.....	-PF10 -
	*	Salida.....	-PF11 -
	*	Filtrado.....	-ENTER-

Figura 151. Parámetros de resultado

C.5 Sistema propuesto de representación y manejo de estructuras 3-D

```
SISTEMAS DE REPRESENTACION 3-D
*****

Terminal de salida
      G.A..... -1-
      Plotter..... -2-
      Ramtek..... -3-
      Hacienda..... -4-

      Opcion ==> 3
```

Figura 152. Inicialización de terminal

PARAMETROS DE ENTRADA

Superficie primitiva ==> 1

Puntos de lambda ==> 128 ** Puntos de beta ==> 128

Lambda maximo ==> 3.0 ** Beta maximo ==> 3.0
Lambda minimo ==> -3.0 ** Beta minimo ==> -3.0
Codigo ==> 1 /// 45 ** Codigo ==> 1 /// 45

Entrada por fichero de las coordenadas poner "*"
X ==> // // ** Y ==> // // ** Z ==> // * //

Línea primitiva ==> 2 ** Puntos de alfa ==> 1000

Entrada por fichero de las coordenadas poner "*"
X ==> // // ** Y ==> // // ** Z ==> // * //

Alfa maximo ==> 100.0 ** Alfa minimo ==> 0.
Codigo ==> 1 /// 43

Giro de	90.0		Escala	Translacion
De X	90.0	** X ///	1.0	** 20.0
De Y	0.0	** Y ///	1.0	** 40.0
De Z	90.0	** Z ///	1.0	** 20.0

Operaciones:

Salida.....	-PF3-	Superficie.....	-PF4-
Línea.....	-PF5-	Dibujo 3-D.....	-PF6-
Reinicializacion...	-PF7-	Corte de superficies.....	-PF8-
Curvas de nivel....	-PF9-	Dibujo 3-D curvas nivel..	-PF10-

Figura 153. Parámetros de entrada al sistema

L E C T U R A D E F I C H E R O S

EL FICHERO NO EXISTE

Filename =====> FICHERO

Filetype =====> IMAGEN

Filemode =====> J1

Tipo de formato

DEC FLOAT..... - 1
BIN FIXED..... - 2
CHAR(1)..... - 3
EBCDIC CON DECIMALES..... - 4
EBCDIC CON EXPONENTE..... - 5

Opcion ==> 3

Figura 154. Entrada por fichero o tablas

DIBUJO TRIDIMENSIONAL

Rotulos en los ejes: ** Parametros del giro:

- Eje X ==> /// LONGITUD /// ** - Azimut ==> 30.0
- Eje Y ==> /// ANCHURA /// ** - Elevacion ==> 40.0
- Eje Z ==> /// ALTURA /// ** - Rotacion ==> 15.0

Plano - centro ==> 50.0 ** Vista - centro ==> 200.0

Nombre del fichero a pintar ==> /// FICHERO /// DIBUJO

Codigo antiguo ==> 1 // 45 ** Magnificacion ==> 1
Codigo nuevo ==> 3 // 0 (Sin ejes = 2)
Nombre del fichero de salida ==> /// FICHERO /// SALIDA

PELIGRO DESTRUCCION

Operaciones:

Salida.....	- PF3-
Dibujo sin lineas ocultas.....	- PF4-
Dibujo de ejes.....	- PF5-
Borrar.....	- PF6-
Cambio de codigos en el fichero a pintar.....	- PF7-
Dibujo con lineas ocultas.....	- PF8-
Dibujo del fichero nombrado.....	- PF9-
Cambio del nombre del fichero de salida.....	-PF10-
Dibujo de marco.....	-PF11-

Figura 155. Dibujo tridimensional

CURVAS DE NIVEL

Nivel maximo de x ==>	3.257	exp ==>	1
Nivel minimo de x ==>	0.	exp ==>	1
Nivel maximo de y ==>	4.365	exp ==>	1
Nivel minimo de y ==>	0.	exp ==>	1
Nivel maximo de z ==>	2.203	exp ==>	2
Nivel minimo de z ==>	0.	exp ==>	1

Opciones:

Nivel maximo de x ==>	3.257	exp ==>	1
Nivel minimo de x ==>	0.	exp ==>	1
Nivel maximo de y ==>	4.365	exp ==>	1
Nivel minimo de y ==>	0.	exp ==>	1
Nivel maximo de z ==>	1.	exp ==>	2
Nivel minimo de z ==>	0.5	exp ==>	1

Numero de niveles ==>	23	Subcodigo automatico ==>	1
Codigo ==>	1	Subcodigo ==>	45

Operaciones:

Salida.....	- PF3-
Curvas de nivel de x constante.....	- PF4-
Curvas de nivel de y constante.....	- PF5-
Curvas de nivel de z constante.....	- PF6-

Figura 156. Generación de curvas de nivel

DIBUJO DE GRAFICOS CON EL TRAZADOR DE CURVAS BENSON 1302

Informacion general sobre el grafico

IDENTIFICACION DEL FICHERO GENERADO CON EL GRAFICO

Numero FORTRAN ==> 8

Nombre ==> FICHERO

Tipo ==> PLOTTER

Disco ==> J1

TAMANO EN CENTIMETROS

Horizontal ==> 40.00 Vertical ==> 30.00

Figura 157. Presentación en trazador de curvas

Apéndice D. Diccionario de términos técnicos

1. **Adelgazamiento** - Operación consistente en convertir una imagen en otra cuyas estructuras sólo tienen un píxel de anchura.
2. **Autopenetración** - Penetración de una superficie multiforme contra ella misma generando nuevas aristas en dicha penetración.
3. **Base de coordenadas** - Base de datos sustitutoria de la base de punteros y el universo, que en vez de contener listas de punteros sobre dicho universo contiene directamente las coordenadas de los puntos que serían señalados sobre él.
4. **Base de datos de líneas** - Base de datos que contiene todas las líneas que deben trazarse al hacer un dibujo 3-D.
5. **Base de punteros** - Base de datos que contiene listas ordenadas de números que señalan puntos de un universo de forma que uniendo los puntos por ese orden resultan las líneas de un dibujo.
6. **Base de datos de superficies** - Base de datos que contiene todas las coordenadas de los vértices de todos los parches alabeados de cuatro lados en que se dividen las superficies opacas 3-D a representar en un dibujo.
7. **Entorno** - Es una máscara de valor octal en una imagen adelgazada, indicativo de la posición de los píxeles vecinos de luminancia no nula con conectividad 8.
8. **Esclavo** - Es el segmento más próximo a un maestro en un punto dado de este último
9. **Gráfico vectorial** - Se emplea este término para designar el resultado de un tratamiento en un ordenador teniendo como entrada una descripción, y como salida un conjunto de vectores.
10. **Imagen de rangos iniciales** - Es una imagen donde cada elemento contiene un valor igual al rango del

correspondiente elemento de otra imagen original, antes de empezar una vectorización.

11. **Imagen de rangos modificados** - Es una imagen donde cada elemento contiene un valor igual al rango del correspondiente elemento de otra imagen original, antes de empezar una vectorización, y con alteraciones propias de casos particulares.
12. **Imagen de rangos actualizados** - Es una imagen donde cada elemento contiene un valor igual al rango del correspondiente elemento de otra imagen original, actualizado en cada momento de un proceso de vectorización.
13. **Imagen raster** - Se emplea este término para designar una imagen en el sentido normal de la palabra.
14. **Línea** - Quebrada compuesta por varios segmentos rectos uno a continuación de otro, siendo el vértice final de cada uno de ellos el punto inicial del siguiente.
15. **Línea de Búsqueda** - Es la línea que trazada desde un punto base de un segmento maestro intersecta los posibles segmentos esclavos en los puntos imagen correspondientes al punto base.
16. **Maestro** - Segmento maestro es el segmento desde el cual se busca el segmento más próximo a él.
17. **Mapa de distancias entre líneas** - Imagen cuyos elementos contienen un nivel de gris o un color proporcional a la distancia entre los dos segmentos rectos más próximos que tenga a su alrededor.
18. **Poligonalización** - Operación consistente en la formación de polígonos convexos entre las líneas más próximas cuando se quiere formar un mapa de distancias.
19. **Punto base** - Es un punto perteneciente a un segmento maestro, desde el cual se está haciendo la búsqueda de esclavo en un momento dado.
20. **Punto imagen** - Es el punto del segmento esclavo correspondiente a un punto base del segmento maestro durante una búsqueda.
21. **Rango** - Se llama rango de un píxel en una imagen adelgazada al número de vecinos, en un entorno de conectividad 8, que no tienen luminancia nula.
22. **Segmentación** - Operación consistente en convertir una imagen multinivel en otra imagen binaria.

23. **Suavizado** - Aproximación sobre una vectorización consistente en la rectificación de las quebradas que constituyen las líneas de un gráfico vectorial.
24. **Triángulos útiles** - Son los parches triangulares planos no degenerados, resultado de proyectar parches alabeados de 4 lados en el espacio 3-D, previamente descompuestos en dos triángulos disjuntos en el mismo espacio 3-D, sobre un plano ortogonal al eje de visión.
25. **Universo** - Conjunto de puntos vértices de vectores en algún sistema de representación.
26. **Vectorización** - Operación consistente en convertir una imagen raster en un gráfico vectorial.

Referencias

La siguiente clasificación de referencias se ha hecho en orden al tema por el cual son citadas en el texto. No se excluye, por tanto, que algunas de ellas pudieran clasificarse en apartados distintos, como de hecho ocurre con aquéllas que tratan un espectro amplio de temas relacionados con los gráficos por ordenador y el tratamiento digital de imágenes.

Gráficos por ordenador

1. A.Rosenfeld y A.C.Kak, "Digital Picture Processing", Academic Press, New York, 1976.
2. W.M.Newman y R.F.Sproull, "Principles of Interactive Computer Graphics", (Sec edition) McGraw Hill, New York, 1979.
3. J.C.Beatty y S.B.Kellog, Tutorial: Computer Graphics (Sec edition), I.E.E.E. Computer Society, Parte VI, New York 1982.
4. J.D.Foley y A.van Dam, "Fundamentals of Interactive Computer Graphics", Addison-Wesley Pub., Reading, 1982.
5. D.H.Ballard y C.M.Brown, "Computer Vision", Prentice-Hall Inc, New Jersey, 1982.
6. T.Pavlidis, "Algorithms for Graphics and Image Processing", Springer-Verlag, Berlín, 1982.
7. J.L.Navalón, "Introducción a los Gráficos Interactivos por ordenador", IBM-Madrid, Centro Científico, Publicación Interna, 1984.
8. Proceedings of the Siggraph, ACM, Vols. 17, 18, 19, núm 3, 1983,1984,1985.

Tratamiento digital de imágenes

9. R.H.Wallis, "An Approach for the Space Variant Restoration and Enhancement of Images", Proceedings of the Symposium on Current Mathematical Problems in Image Science, Monterrey, California (USA), 1976.
10. R.C.Gonzalez y P.Wintz, "Digital Image Processing", Addison-Wesley Pub., Londres, 1977.
11. H.C.Andrews y B.R.Hunt, "Digital Image Restoration", Prentice-Hall Inc., New Jersey, 1977.
12. W.Frei y C.C.Chen, "Fast Boundary Detection: A Generalization and a New Algorithm", I.E.E.E. Trans. on Computers, Vol 26, núm. 10, págs. 988-998, 1977.
13. W.K.Pratt, "Digital Image Processing", John Wiley and Sons, New York, 1978.
14. J.Jiménez y N.García, "Introducción al Tratamiento Digital de Imágenes," IBM-Madrid, Centro Científico, PCI 01.82, 1982.

15. A.V.Oppenheim y R.W.Schafer "Digital Signal Processing", Prentice-Hall Inc, New Jersey, 1975.
16. A.V.Oppenheim et al., "Signals and Systems", Prentice-Hall Inc., New Jersey, 1983.

Matemáticas para gráficos por ordenador y tratamiento digital de imágenes

17. A.Papoulis, "Probability, Random Variables and Stochastic Processes", Mc-Graw Hill Inc., Londres, 1965.
18. D.F.Rogers y J.A.Adams, "Mathematical Elements for Computer Graphics", Mc-Graw Hill, New York, 1976.
19. R.A.Liming, "Mathematics for Computer Graphics", Aero Publishers Inc, Fallbrook (California-USA), 1976.
20. S.H.Chasen, "Geometric Principles and Procedures for Computer Graphics Applications", Prentice-Hall, New Jersey, 1978.
21. Baker y Philips, "The Numerical Solution of Non-Linear Problems", Clarendon Press, Oxford, 1981.
22. T.W.Sederberg y D.C.Anderson, "Implicit Representation of Parametric Curves and Surfaces", Computer Vision, Graphics, and Image Processing, Vol. 28, págs. 72-84, 1984.

23. J.Kautsky et al., "Smoothed Histogram Modification for Image Processing", Computer Vision, Graphics, and Image Processing, Vol. 26, núm 3, págs. 271-291, Junio 1984.
24. Fu-Nian Ku, "The Principles and Methods of Histogram Modification Adapted for Visual Perception", Computer Vision, Graphics, and Image Processing, Vol. 26, núm 1, págs. 107-117, Marzo 1984.

Segmentación, Adelgazamiento, Vectorización, Suavizado y Unión de estructuras

25. R.Stefanelli y A.Rosenfeld, "Some Parallel Thinning Algorithms for Digital Pictures," Journal of the ACM, núm.18, págs 255-264, 1971.
26. A.Rosenfeld "A Characterization of Parallel Thinning Algorithms", Information and Control, núm.29, págs 286-291, 1975.
27. H.Tamura, "A Comparison of Line Thinning Algorithms from Digital Geometry Viewpoint", Proceedings of the 4th International Joint conference on Pattern Recognition, Kyoto, págs. 715-719, 1978.
28. J.Jiménez y J.L.Navalón, "The Structure of Queries on Geometric Data", in Data Base Techniques for Pictorial Applications", A.Blaser ed., Springer-Verlag, núm. 6, págs 219-232, Berlín, 1979.
29. T.Pavlidis, "A Thinning Algorithm for Discrete Binary Images," Computer Vision, Graphics and Image Processing, Vol.13, núm.2, págs 142-157, 1980.

30. P.C.Chen y T.Pavlidis, "Image Segmentation as an Estimation Problem", *Computer Graphics and Image Processing*, núm. 12, págs. 153-172, 1980.
31. M.J.B.Duff, "Neighbourhood Operators", *Proceedings of the International Symposium Organised by the Rank Prize Funds*, editados por Braddick y Sleigh en *Physcal and Biological Processing of Images*, Londres (UK), págs 53-72, Septiembre 1982.
32. J.Jiménez y J.L.Navalón, "Some Experiments in Image Vectorization", *IBM Journal of Research and Development*, Vol 26, núm. 6, págs. 724-734, Noviembre 1982.
33. J.L.Navalón y J.Jiménez, "Extracción, visualización y codificación de estructuras geométricas," *Actas V Congreso de Informática y Automática*, págs. 791-797, Madrid, 1982.
34. M.Cogollos, J.L.Navalón y J.Jiménez, "Visualización, reconstrucción y manipulación de objetos tridimensionales en dos dimensiones", *Actas V Congreso de Informática y Automática*, págs. 509-514, Madrid, 1982.
35. J.Jiménez y J.L.Navalón, "The Map Manipulation System: an Experiment in Image Vectorization," *IBM-Madrid, Centro Científico, PCI 03.82*, 1982.
36. D.Barba y J.Ronsin, "Image Segmentation using new measure of the texture feature", *Proceedings of the International Conference on Digital Signal Processing*, págs. 749-753, Florencia (Italia), Septiembre 1984.
37. R.W.Conners, M.M.Trivedi y C.A.Harlow, "Segmentation of a High-Resolution Urban Scene Using Texture Operators", *Computer Vision, Graphics, and Image Processing*, Vol. 25, núm 3, págs. 273-310, Marzo 1984.
38. M.Nadler, "Document Segmentation and Coding Techniques", *Computer Vision, Graphics, and Image Processing*, Vol. 28, núm 2, págs. 240-262, Noviembre 1984.
39. J.Rodríguez, D.Barba, "Picture coding scheme based on a two-component model", *Proceedings of the International Conference on Digital Signal Processing*, Florencia (Italia), págs. 498-503, Septiembre 1984.
40. K.Wall y Per-Erik Danielson, "A Fast Sequential Method for Polygonal Approximation of Digitized Curves", *Computer Vision, Graphics, and Image Processing*, Vol. 28, núm 2, págs. 220-227, Noviembre 1984.
41. R.M.Haralick y L.G.Shapiro, "Image Segmentation Techiques", *Computer Vision, Graphics, and Image Processing*, Vol. 29, núm. 1, págs. 100-132, Enero 1985.
42. S.Boukharouba, J.M.Rebordao y P.L.Wendel, "An Amplitude Segmentation Method Based on the Distribution Function of an Image", *Computer Vision, Graphics, and Image Processing*, Vol. 29, núm. 1, págs. 47-59, Enero 1985.
43. F.Cohen, "Adaptative Hierarchical Algorithm for Accurrate Image Segmentation", *Proceedings of the ICASSP85*, Vol. 2, págs. 897-900, Tampa (Florida-USA), Marzo 1985.
44. R.Cristi et al. "Segmentation of Multilevel Images Using Gibbs Distribution", *Proceedings of the ICASSP85*, Vol. 2, págs. 901-904, Tampa (Florida-USA), Marzo 1985.
45. J.L.C.Sanz, "Fast Object Segmentation in Texture Backgrounds", *Proceedings of the ICASSP85*, Vol. 2, págs. 905-908, Tampa (Florida-USA), Marzo 1985.
46. M.del Sordo y T.Kasvand, "A Near-Neighbor Processor for Line Thinning", *Proceedings of the ICASSP85*, Vol. 4, págs. 1523-1526, Tampa (Florida-USA), Marzo 1985.
47. M.S.Landy y Y.Cohen, "Vectorgraph Coding: Efficient Coding of Line Drawings", *Computer Vision, Graphics, and Image Processing*, Vol. 30, núm.3 págs. 331-344, Junio 1985.

48. M.P.Martínez y J.L.Navalón, "Un algoritmo de adelgazamiento de estructuras", *Actas VI Congreso de Informática y Automática*, págs. 65-68, 1985.
49. G.Cisneros, A.Vicente, G.Abella y N.García, "Método de codificación de imágenes para transmisión por segmentación y adelgazamiento de estructuras linealizables", *Proceedings del II Simpósio de Electrónica das Telecomunicações*, Lisboa, Mayo 1986 (en prensa).
50. G.Cisneros y N.García, "A New Procedure for Image Vectorisation", *Proceedings of the EUSIPCO*, La Haya Septiembre 1986 (en prensa).
51. A.Rosenfeld, "Picture Processing by Computer", Academic Press, New York, 1969.
52. J.E.Bresenham, "A Linear Algorithm for Incremental Digital Display of Circular Arcs", *Communications of the ACM*, págs. 100-106, 1977.
53. K.Yokokawa y T.L.Kunii, "A Definition of Neighborhood of a Region for Picture Processing", *Computer Vision, Graphics, and Image Processing*, Vol. 14, págs. 112-144, 1980.
54. Per-Erik Danielson, "Euclidean Distance Mapping", *Computer Vision, Graphics, and Image Processing*, Vol.14, págs. 227-248, 1980.
55. A.Scher, M.Sheneir y A.Rosenfeld, "A Method for Finding Pairs of Antiparallel Straight Lines", *I.E.E.E., Trans. on PAMI*, Vol 4, núm.3, 1982.
56. A.Distante y N.Veneziani, "A Two-Pass Filling Algorithm for Raster Graphics", *Computer Vision, Graphics and Image Processing*, Vol. 20, págs 288-295, 1982.
57. G.Cisneros, "Extracción de información gráfica en imágenes de dominios magnéticos," Proyecto Fin de Carrera, E.T.S. Ingenieros de Telecomunicación, Universidad Politécnica, Madrid, 1983.
58. G.Cisneros, P.S.Sánchez, C.Aroca, J.L.Navalón y N.García, "Extraction of Graphic Information in Magnetic Domains Images", *Proceedings of the International Conference on Digital Signal Processing*, págs. 524-528, Florencia (Italia), Septiembre 1984.

Conversión de gráficos en raster

51. A.Rosenfeld, "Picture Processing by Computer", Academic Press, New York, 1969.
52. J.E.Bresenham, "A Linear Algorithm for Incremental Digital Display of Circular Arcs", *Communications of the ACM*, págs. 100-106, 1977.

Búsqueda de líneas y rellenado de polígonos

53. K.Yokokawa y T.L.Kunii, "A Definition of Neighborhood of a Region for Picture Processing", *Computer Vision, Graphics, and Image Processing*, Vol. 14, págs. 112-144, 1980.
54. Per-Erik Danielson, "Euclidean Distance Mapping", *Computer Vision, Graphics, and Image Processing*, Vol.14, págs. 227-248, 1980.

Sistemas 3-D en general

59. B.R.Heap, "Algorithms for the Production of Contour Maps over an Irregular Triangular Mesh", *National Physical Laboratory report*, NAC10, 1972.
60. G.Cisneros y N.García, "Three Dimensional Pictures with Curves and Surfaces in Parametric Coordinates", *Proceedings of the Mediterranean Electrotechnical Conference*, págs. 213-216, Madrid, Octubre 1985.
61. R.T.Farouki y J.K.Hinds, "A Hierarchy of Geometric Forms", *I.E.E.E. CG&A*, págs. 51-78, Vol.5, núm.5, Mayo 1985.
62. S.G.Satterfield y D.F.Rogers, "A Procedure for Generating Contour Lines from a B-Spline Surface", *I.E.E.E. CG&A*, págs. 71-75, Vol. 5, núm.4, Abril 1985.

63. Y.Ota, H.Arai y S.Tokumasu, "An Automated Finite Polygon Division Method for 3-D Objects", I.E.E.E. CG&A, págs. 60-70, Vol.5, núm.4, Abril 1985.

Intersección de superficies

64. M.Cyrus y J.Beck, "Generalized Two- and Three Dimensional Clipping", Computers and Graphics, Vol.3, núm.1, págs 23-28, 1978.
65. You-Dong Liang y B.Barsky, "An Analysis and Algorithm for Polygon Clipping", Communications of the ACM, Vol.26, núm.11, págs 868-877, 1983.
66. R.B.Lee y D.A.Fredericks, "Intersection of Parametric Surfaces and a Plane", I.E.E.E. CG&A, Vol.4, núm.8, págs 48-51, 1984.
67. You-Dong Liang y B.Barsky, "A New Concept and Method for Line Clipping", ACM Trans. on Graphics, Vol.3, núm.1, págs. 1-22, 1984.
68. F.Yamaguchi y T.Tokieda, "A Solid Modeler with a 4x4 Determinant Processor", I.E.E.E. CG&A, págs. 51-59, Vol.5, núm.4, Abril 1985.
69. D.F.Rogers y L.M.Rybak "On an Efficient General Line-Clipping Algorithm", I.E.E.E. CG&A, págs. 82-86, Vol.5, núm.1, Enero 1985.

Interpolación de superficies

70. P.Bezier, "Numerical Control- Mathematics and Applications", Wiley & Sons, Londres, 1972.
71. P.Bezier, "Mathematical and Practical Possibilities of UNISURF", Computer Aided Geometric Design, R.E.Baruhill y R.F.Riesenfeld editores, Academic-Press, New York, 1974.

72. Ch.de Boor, "A Practical Guide to Splines", Springer-Verlag, Berlín, 1978.

73. I.D.Faux y M.J.Pratt, "Computational Geometry for Design and Manufacture", Halsted Press (Wiley), New York, 1979.

74. T.C.Chen y R.J.P. de Figueiredo, "Two-Dimensional Interpolation by Generalized Spline Filters Based on Partial Differential Equation Image Models", I.E.E.E. Trans. on ASSP, Vol. 33, núm.3, págs 631-642, 1985.

Modelado de sólidos

75. A.A.G.Requicha, "Representation for Rigid Solids, Theory, Methods and Systems," ACM Computing Surveys,, págs 437-464, 1980.
76. D.Meagher, "Geometric modeling using octree encoding," Computer Vision, Graphics and Image Processing,, págs. 129-147, 1982.
77. A.A.G.Requicha y H.B.Voelcker, "Solid Modeling: A Historical Summary and Contemporary Assessment", I.E.E.E. CG&A, Vol.2 núm.2 págs 9-24, 1982.
78. A.A.G.Requicha y H.B.Voelcker, "Solid Modeling: Current Status and Research Directions", I.E.E.E. CG&A, Vol.3 núm.7 págs 25-37, 1983.
79. D.F.Watson y G.M.Philip, "Systematic Triangulations", Computer Vision, Graphics, and Image Processing, Vol. 26, págs. 217-223, 1984.
80. T.W.Sederberg y D.C.Anderson, "Steiner Surface Patches", I.E.E.E. CG&A, págs. 23-26, Vol.5, núm.5, Mayo 1985.
81. P.R.Wilson, "Euler Formulae and Geometric Modeling", I.E.E.E. CG&A, págs. 24-36, Vol.5, núm.8, Agosto 1985.

82. Mathematical Application Group Inc., "3-D Simulated Graphics", Datamation, Febrero 1968.
83. D.E.Knuth, "The Art of Computer Programing", Addison-Wesley Pub., Reading 1973.
84. I.E.Sutherland, I.E.Sproull y R.A.Schumacker, "A Characterization of Ten Hidden-Surface Algorithms", en ACM Computing Surveys, págs. 1-55, Marzo 1974.
85. K.Weiler y P.Altherton, "Hidden Surface Removal Using Polygon Area Sorting", Computer Graphics, 1977.
86. G.Hamlin y C.W.Gear, "Raster-Scan Hidden Surface Algorithm Techiques", Computer Graphics (Proc. Siggraph 77), Vol.11, núm.2, págs. 206-213, 1977.
87. J.F.Blinn, "Computer display of curved surfaces", Tesis Doctoral, Universidad de Utah, 1978.
88. T.Wright, "Visible Surface Plotting", en Collected Algorithms from ACM, Algoritmo núm. 475, Vol. 2, 1980.
89. S.L.Watkins, "Masked Three-Dimensional Plot Program with Rotations", en Collected Algorithms from ACM, Algoritmo núm 483, Vol. 2, 1980.
90. J.M.Lane, L.C.Carpenter, T.Whitted y J.F.Blinn, "Scan Line Methods for Displaying Parametrically Defined Surfaces", Communications of the ACM, Enero 1980.
91. S.Secrest y D.P.Greenberg, "A visible Polygon Reconstruction Algorithm", Computer Graphics (Proc. Siggraph 81), Vol.15, núm.3, págs. 206-213, 1981.
92. Y.Ohno, "A Hidden Line Elimination Method for Curved Surfaces", Computer Aided Design, Vol.15, núm 4, págs. 209-216, Julio 1983.
93. C.H.Séquin y P.R. Wensley, "Visible Feature Return at Object Resolution", I.E.E.E. CG&A, págs. 37-50, Vol.5, núm.5, Mayo 1985.

Modelos de iluminación

94. H.Gourand, "Continuous Shading of Curved Surfaces", I.E.E.E. Trans. on Computers, págs 623-628, 1971.
95. B.T.Phong, "Ilumination for Computer Generated Images", Communications of the ACM, Vol. 18, págs.311-317, 1975.
96. R.W.G.Hunt, "The Reproduction of Color", Wiley & Sons, New York, 1975.
97. J.F.Blinn y M.E.Newell, "Texture and Refection in Computer Generated Images", Communications of the ACM, págs. 177-184, 1976.
98. J.F.Blinn, "Models of Light Reflection for Computer Synthetized Pictures," Computer Graphics, págs. 192-198, 1977.
99. F.Crow, "Shadows Algorithms for Computer Graphics", Computer Graphics, págs 242-247, 1977.
100. T.Whitted, "An Improved Illumination Model for Shaded Display", Communications of the ACM, págs 343-349, 1980.
101. F.C.Crow, "A Comparison of Anti-Aliasing Techniques", I.E.E.E. CG&A, págs. 40-49, 1981.
102. J.F.Blinn, "Light Reflection Funtions for Simulation of Clouds and Dusty Surfaces", Computer Graphics, 1982.
103. A.Fournier, D.Fussell y L.Carpenter, "Computer rendering of stochastics models", Communications of the ACM, págs. 371-384, 1982.

104. W.T.Reeves, "Particle Systems- a Tehcnique for Modeling a Class of Fuzzy Objects", *Computer Graphics*, 1983.
105. H.Christiansen y M.Stephenson, "MOVIE.BYU Training Text", University Press, Brigham Young University, Provo (Utah), 1983.
106. T.Nishita, I.Okamura y E.Nakamae, "Shading Models for Point and Linear Sources", *ACM Trans. on Graphics*, págs. 124,146, Vol.4, núm.2, Abril 1985.
107. Lih-Shyang Chen, G.T.Herman, R.A.Reynolds y J.K.Udupa, "Surface Shading in the Cuberille Environment", *I.E.E.E. CG&A*, págs. 33-43, Vol.5, núm.12, Diciembre 1985.

Reconocimiento de estructuras alfanuméricas

108. U.Montanari, "Continuous Skeletons from Digitized Images", *Journal of the ACM*, núm.16, págs. 534-549, 1969.
109. M.Beun, "A Flexible Method for Automatic Reading of Handwritten Numerals", *Philips Technical Review*, núm.33, Part I, págs. 89-101, 1973.
110. W.Hattich, M.Tropf y G.Winkler, "Combination of Statistical and Syntactical Pattern Recognition-Applied to Classification of Unconstrained Handwritten Numerals", *Proceedings of the 4th International Joint Conference on Pattern Recognition*, Kyoto, págs. 786-788, 1978.
111. S.Naito, H.Arakawa y I.Masuda, "Recognition of Hand-Printed Alphanumeric and Symbols based on Centroid Lines", *Proceedings of the 4th International Joint Conference on Pattern Recognition*, Kyoto, págs. 797-801, 1978.

112. J.R.Ward y B.Blessner, "Interactive Recognition of Handprinted Characters for Computer Input", *I.E.E.E. CG&A*, págs. 24-37, Vol.5, núm.9, Septiembre 1985.

Varios

113. A.Palomeque, "La villa romana de la finca de Las Tamujas", *AEA*, núm.92, 1955.
114. I.Zamorano, "Caracteres del arte visigodo de Toledo", *Anales Toledanos*, núm.10, págs 140-143, 1974.
115. C.Aroca, P.S.Sánchez y E.López, "Magnetoelastic Effects in Amorphous $Fe_{40} - Ni_{40} - P_{14} - B_6$ alloys," *I.E.E.E. Trans. on Magnetism*, Vol 4, núm.17, págs. 1462-1467, 1981.
116. N.García, "Procesamiento digital automático de estructuras virales cristalinas", Tesis Doctoral, E.T.S. Ingenieros de Telecomunicación, Universidad Politécnica, Madrid, 1983.
117. J.J.Storch de Gracia, "Artes decorativas visigodas en Toledo", Tesis de Licenciatura, Universidad Complutense, Madrid, 1983.
118. J.M.Carazo, "Reconstrucción tridimensional del cuello del bacteriófago D29 ", Tesis Doctoral, 1984.
119. G.Cisneros y J.Pérez, "Computer Graphics Aided Large Signal Design Methods for Microwave Oscillators", *Proceedings of the Mediterranean Electrotechnical Conference*, Madrid, Octubre 1985.
120. G.Abella y A.Vicente, "Aplicación de un modelo visual con base biológica a la identificación de texturas y reconocimiento de formas de onda", *Actas de las Primeras Jornadas sobre Inteligencia Artificial*, Universidad Complutense, Madrid, Junio 1985.

121. J.J.Storch de Gracia, I.García, G.Abella, A.Vicente y G.Cisneros, "Digital Image Processing Applied to Archaeological Research", CESTA, Niza, Abril 1986 (en prensa).
122. I.García, J.J.Storch de Gracia, G.Abella, G.Cisneros y A.Vicente, "El proceso digital de imagen en la investigación arqueológica", Revista Universidad de Murcia, 1986 (en prensa).
123. J.Jiménez, A.Santisteban, J.M.Carazo, y J.L.Carrascosa, "Computer Graphic Display Method for Visualizing Three-Dimensional Biological structures", Science, 1986 (en prensa).
124. J.J.Storch de Gracia, I.García, G.Abella, A.Vicente y G.Cisneros, "Las técnicas de transmisión y codificación de imagen en la investigación arqueológica", Proceedings del II Simpósio de Electrónica das Telecomunicações, Lisboa, Mayo 1986 (en prensa).