



**Universitat  
Autònoma  
de Barcelona**



**Un modelo sintáctico para la representación,  
segmentación y reconocimiento de símbolos  
texturados en documentos gráficos**

Memoria de tesis presentada por  
**Gema Sánchez Albaladejo** a la  
Universitat Autònoma de Barcelona y la  
Université Henry Poincaré (Nancy 1)  
para la obtención del título de  
**Doctora en Informática.**

Bellaterra, 29 de noviembre de 2001

Director: **Dr. Josep Lladós i Canet**  
Universitat Autònoma de Barcelona  
Dept. Informàtica & Computer Vision Center

Co-director: **Dr. Karl Tombre**  
Laboratoire Lorrain de Recherche en Informatique et ses Applications. (LORIA)  
École de Mines de Nancy

a mis padres...



# Agradecimientos

Este es un capítulo especial en la tesis, no sólo porque es el más leído sino porque aún en las tesis científicas permite darle un matiz humano al trabajo realizado. Una tesis es un largo camino en el que te vas encontrando con problemas a solucionar, no sólo de índole académica. Aquí voy a intentar recordar a todas las personas que en algún momento y de alguna forma me han ayudado a seguir adelante, espero no olvidar a nadie.

En primer lugar quiero agradecer a Juan José Villanueva, director del Centro de Visión por Computador y jefe de la unidad de procesamiento de imágenes e inteligencia artificial del departamento de Informática de la U.A.B, el haberme dado la oportunidad de realizar esta tesis. Me contrató de una forma peculiar porque me hizo la entrevista por teléfono ya que yo estaba en Málaga y Lisboa, pero debo tener una voz de persona formal ya que parece que le convencí de que lo haría bien...

A mi director de tesis Josep Lladós, quiero agradecerle todo el apoyo que me ha ofrecido durante todo este tiempo aún antes de ser mi director. En todo momento ha estado ahí para resolver dudas y ponerse a trabajar conmigo en los problemas que nos han ido surgiendo en cada una de las etapas de la tesis. Creo que ha intentado inculcarme su naturaleza metódica y ordenada y espero no haber terminado con su paciencia con mi tendencia al Kaos.

También quiero agradecer a Karl Tombre el haberme acogido en Francia y haber accedido a ser mi director desde Francia. Quiero destacar el apoyo que he recibido de él en todo momento y la cualidad que tiene para infundir entusiasmo por el trabajo que se realiza a las personas que tiene a su alrededor, la mayoría de las veces entrar en su despacho a hablar sobre la tesis suponía salir con una carga renovada de entusiasmo.

Quiero agradecer a Enric Martí que en un inicio fue también mi director, el haberme iniciado en este mundo de la Universidad tanto a nivel de investigación como de docencia, y el haber continuado estando ahí cuando lo necesitaba.

A todos los miembros del tribunal, Juan José Villanueva, Jean-Paul Haton, Jaime López Krahe, Xulio Fernández Hermida, Philippe Dosch y Jordi Regincós por haber accedido a serlo.

Bueno y ahora llega el grupo más numeroso de compañeras y compañeros del cvc, a todos tengo algo que agradecer pero es muy probable que olvide a alguno así que esta primera línea sirve de agradecimiento a todos y ahora voy a detallar algunos aspectos más concretos. A mi compañero de despacho Ricardo Toledo por su ayuda con el Visual C++, las librerías STL, el latex, por sus charlas sobre la investigación, la vida en la Universidad y la vida en general, la historia, la política, y

en fin por esos momentos de distensión que ayudan a seguir trabajando, también por haber aguantado mis resoplidos cada vez más frecuentes y ruidosos a medida que se acercaba el momento de entregar la tesis. A Jordi Saludes por las funciones lineales a trozos. A Anna Salvatella por ayudarme con el francés tanto en el resumen como en la presentación oral de la tesis, por dedicarme su tiempo en un momento en que ella misma tenía mucho trabajo con su tesina. A Xavier Sánchez por haberse leído todo el capítulo de gramáticas y hacerme algunas correcciones. A Montse Cullere por haberme depositado la tesis mientras yo tenía que asistir a un congreso y haber tenido que hacer todo un papeleo que no le correspondía en absoluto. A Ramón Baldrich, Albert Pujol y Xavier Varona, por haber sufrido conmigo un Agosto "inolvidable" y haber hecho que fuera más llevadero el venir a trabajar de lunes a domingo  $n$  horas diarias, y haberme traído y llevado a la Vila. A Ernest y Felipe por ayudarme con la presentación. Y en fin a todos por vuestra colaboración, interés y por las charlas en la cafetería.

Quiero agradecer a los miembros del grupo ISA en el Centro Loria el apoyo y la buena acogida que me han dado siempre, la paciencia que han tenido conmigo y su ayuda. Especialmente quiero agradecer a mis dos compañeros de despacho, Christian Ah-Soon y Philippe Dosch el haber estado a mi lado desde el momento en que llegué a Nancy, haberme ayudado con la burocracia, la residencia, el laboratorio, el idioma y en fin durante toda la estancia que realicé allí y las posteriores visitas que he ido haciendo a lo largo de la tesis. Merci bien!

A mi familia, especialmente a mis padres y a mi hermano, por su ayuda y su apoyo, no sólo durante la tesis sino durante todos mis estudios, por estar siempre ahí aún cuando yo no me daba cuenta.

A mis amigos, por la gran paciencia y comprensión demostradas. En especial a Carme por estar siempre ahí cuando te necesitaba, por poder hablar siempre contigo de todo y por esa generosidad que te caracteriza. Y en general a todos los que habéis sufrido en mayor o menor medida esta tesis, con mis largas desapariciones y breves llamadas telefónicas, a Anna, Olga, Xavi, Thais, Dina, Jordi y Raul.

A Gustavo cuyas clases me fueron de gran ayuda en la recta final de este trabajo.

# Resumen

El análisis de documentos, después de varias décadas de actividad, sigue siendo una de las áreas científicas de referencia de la Visión por Computador. El análisis de documentos gráficos se ocupa de la identificación de entidades gráficas en un documento, ya sea impreso o en formato vectorial para su posterior almacenamiento, edición, validación o comprensión de sus características. Áreas como la arquitectura, cartografía, ingeniería, urbanismo, etc. se han beneficiado de los resultados científicos del análisis de documentos gráficos que son transferidos a entornos CAD cada vez más avanzados. Dentro del análisis de documentos gráficos el reconocimiento de símbolos es un problema complejo y en el que existen todavía retos importantes para la comunidad científica. El reconocimiento de símbolos puede definirse como la extracción de aquellas entidades gráficas con un significado determinado dentro del contexto en el que se enmarque el documento. Los símbolos los podemos dividir en dos clases principalmente, los basados en prototipos y los texturados, es decir símbolos que tienen alguna de sus partes formada por una textura estructurada.

Las texturas estructuradas se definen por uno o varios elementos o texels que se distribuyen en el espacio siguiendo unas reglas de emplazamiento. En nuestro modelo de textura estructurada consideraremos que estos texels son polígonos cerrados o segmentos de línea. Análogamente, un plano o documento a reconocer se modeliza por medio de un grafo de adyacencia de regiones, en el que los nodos representan los polígonos encontrados en el plano y las aristas las relaciones entre ellos.

El trabajo realizado tiene dos partes bien diferenciadas, la segmentación y el reconocimiento de texturas. En la primera, segmentamos texturas por medio de un clustering sobre el grafo que representa el plano. De esta manera obtenemos una serie de clusters de polígonos similares con reglas de emplazamiento similares. Así los clusters donde encontramos un número suficiente de elementos están formando una textura y su representante es el polígono y la vecindad medios calculados. Cuando es una textura compuesta y está formada por más de un texel encontramos clusters solapados, y cada uno de ellos nos indica uno de los texels que forman la textura. La segunda fase infiere una representación de la textura a partir de los representantes de los clusters. Esta representación es una gramática de grafo que permite producciones de error. Con ella es posible reconocer si un símbolo texturado existe en un grafo por medio de un parser. Además, el motor sintáctico de reconocimiento incorpora un modelo de error que permite refinar las fronteras de la textura gracias a sus producciones de error que controlan si en las fronteras aparecen texels partidos, y distorsiones que hacen que los texels queden unidos o partidos y que son producidas por ruido a la

hora de obtener la imagen o en los preprocesos como la vectorización.

El clustering utilizado es jerárquico sobre un grafo. Los elementos a agrupar son polígonos que están representados por medio de cadenas cíclicas que representan su contorno, por ello la función de distancia entre los representantes de los clusters es la edición de cadenas. En el proceso de agrupación es necesario calcular un representante de cada cluster, que determinará la regla de emplazamiento y el contorno, que representan al grupo de elementos agrupados. Esta forma y vecindad medias se calculan considerando las cadenas que definen los polígonos como funciones por trozos y calculando la cadena media como la media de esas funciones. El proceso permite ciertas distorsiones en las formas agrupadas y en sus vecindades pero no permite agrupar polígonos que estén partidos o fusionados entre sí.

El proceso de inferencia gramatical se realiza teniendo en cuenta los texels que forman la textura y sus reglas de emplazamiento. A partir de ellas se calculan de forma automática los ciclos cerrados que forman estas vecindades y se generan las reglas de la gramática de grafo. Al final se introducen producciones de error para tolerar texturas distorsionadas por texels partidos o fusionados. Una vez la gramática ha sido calculada se reconoce el símbolo representado por ella mediante un parser que analiza un grafo de entrada y recorre los nodos del mismo que cumplen las reglas de la gramática agrupándolos en uno o varios clusters que serán las ocurrencias del símbolo en el grafo presentado.

En resumen, en este trabajo proponemos soluciones para segmentar, mediante el clustering de formas, texturas estructuradas, y modelizar y reconocer un símbolo texturado mediante la gramática y el análisis sintáctico que podemos realizar con ella sobre un grafo, respectivamente.

**Palabras clave:** Análisis de documentos gráficos, reconocimiento de símbolos, texturas estructuradas, reconocimiento sintáctico de patrones, gramáticas de grafo.

# Abstract

After decades of activity, Document Analysis continues being one of the main scientific areas in Computer Vision. The aim of the graphical document analysis is to identify graphical entities in a document, printed or in a vectorial format, to store, to edit, to validate or to understand its characteristics. Several fields as the architecture, cartography, engineering, town planning, etc have taken advantage of the scientific results of the graphical document analysis that are transferred to CAD environments more and more advanced. Symbol recognition is a complex problem in Graphical Document Analysis, and it still have important challenges for the scientific community. Symbol Recognition can be defined as the extraction of graphical entities which have a special meaning in the context of the analyzed document. Symbols can be mainly divided in two classes, the prototyped-based symbols and the textured symbols, that means symbols with one or more of its parts formed by a structural texture.

Structural textures are defined by one or more elements or texels distributed in the space following placement rules. In our structural texture model we consider texels as closed polygonals or line segments. On the other hand a document or plan to be analyzed is modeled by means of a Region Adjacency Graph, where the nodes represent polygonals found in the original document and the edges the relations among them.

This dissertation has two parts well defined, the segmentation and the recognition of textures. In the first part, textures are segmented by means of a clustering over the graph that represents the plan. In that way a set of clusters, formed by similar polygonals placed following similar rules, are obtained. Then the clusters with a certain number of elements are considered to form a texture and their representative is presented by the mean polygonal and the mean neighbourhood, both computed. When the texture is composed of more than one kind of texels, the clusters appear overlapped, one for each kind of texel forming the texture. The second part is devoted to infer the texture representation from the representative of the clusters. This representation is a graph grammar with error productions. This grammar allows the recognition of a textured symbol in a graph by means of a parser. The syntactic recognition engine has an error model to allow the recognition of the texture borders because of its error productions that control if in the borders appear cut texels, or distortions that make texels to appear merged or cut and that are due to the noise in the input process or in some preprocess as the vectorization.

The clustering used in this work is a hierarchical one on a graph. The elements to be clustered are polygonals represented by means of cyclic strings representing

their contour, for that reason the distance between the representatives of the clusters is the string edit distance. In the clustering process it is necessary to compute a representative for each cluster, it determine the placement rule and the contour that represents the set of grouped elements. This mean shape and neighbourhood are computed by transforming the strings into piece-wise functions and computing the mean string as the mean among these functions. The process allows certain kind of distortions in the grouped shapes and in its neighbourhoods but it does not allow to group polygonals cut of merged.

The grammar inference process is done bearing in mind the texels forming the texture and their placement rules. From these rules the closed loops are computed automatically and the rules of the graph grammar are generated. At the end the error productions are introduced to tolerate textures distorted by cut or merged texels. Once the grammar is computed the symbol represented by it is recognized by parsing an input graph and visiting all the nodes following the rules of the grammar, and grouping them in one or several clusters which are the instances of the symbol in the input graph.

To summarize, in this work a solution to segment structural textures, by means of shape clustering, is proposed. Then solutions to modelize a textured symbol by means of a grammar, and to recognize it by parsing with that grammar over an input graph, are presented.

**Key words:** Graphical Document Analysis, Symbol Recognition, Structural Textures, Syntactical Pattern Recognition, Graph Grammars.

# Résumé

Après plusieurs dizaines d'années d'activité, l'analyse de documents reste l'un des domaines scientifiques de référence en vision par ordinateur. L'analyse de documents graphiques s'occupe de l'identification d'entités graphiques dans un document imprimé ou bien au format vectoriel pour le stockage, l'édition, la validation ou la compression de ses caractéristiques. Certains domaines comme l'architecture, la cartographie, l'ingénierie ou l'urbanisme, etc, ont bénéficié des résultats scientifiques de l'analyse des documents graphiques, qui sont transférés à des domaines d'utilisation CAO de plus en plus avancés. En analyse de documents graphiques, la reconnaissance de symboles est un problème complexe, qui pose encore des défis importants pour la communauté scientifique. La reconnaissance de symboles peut se définir comme l'extraction des entités graphiques qui ont un sens déterminé dans le contexte du document. Les symboles peuvent être classifiés en deux classes principales : ceux qui sont basés sur des prototypes, et ceux qui sont texturés, c'est-à-dire ceux qui ont une partie formée par une texture structurée.

Le point de départ de cette thèse s'inscrit dans le domaine de l'analyse des plans architecturaux, pour aborder comme objectifs fondamentaux la modélisation, la segmentation et la reconnaissance de symboles formés par des textures structurées. Même si les plans architecturaux ont été utilisés comme domaine d'expérimentation, les modèles proposés peuvent se généraliser à d'autres domaines où les éléments à reconnaître sont des textures structurées.

Les textures structurées se définissent comme un ou plusieurs éléments ou texels qui se distribuent dans l'espace en suivant des règles d'emplacement. Dans notre modèle de texture structurée, ces texels seront considérés comme polygones fermés ou segments de ligne. De manière analogue, un plan ou un document à reconnaître sera modélisé au moyen d'un graphe de régions d'adjacence, où les nœuds représentent les polygones trouvés dans le plan, et les arcs représentent les relations entre les nœuds.

Le travail réalisé est divisé en deux parties différentes : la segmentation et la reconnaissance des textures. Dans la première partie, les textures sont segmentées au moyen d'un regroupement sur le graphe qui représente le plan. Ainsi, une série de groupes de polygones similaires sont obtenus avec des règles d'emplacement similaires. Les clusters ayant un nombre suffisant d'éléments forment une texture, et leurs représentants sont le polygone et le voisinage moyen calculés. Quand la texture est composée de plusieurs texels différents, les clusters trouvés se superposent et chacun indique un des texels qui forment la texture. La deuxième phase infère une représentation de texture à partir des représentants des clusters. La représentation choisie

est une grammaire de graphe avec prise en compte des erreurs. Avec cette représentation, il est possible de détecter si un symbole texturé existe dans un graphe au moyen d'un analyseur syntaxique. Le moteur syntaxique de reconnaissance incorpore en plus un modèle d'erreur qui permet de raffiner les frontières de la texture, grâce à la modélisation des erreurs, qui contrôle si des texels coupés apparaissent aux frontières et si des distorsions dues au bruit de numérisation ou de prétraitement (comme la vectorisation) font que les texels restent unis ou sont divisés.

Le regroupement utilisé est hiérarchique sur un graphe. Les éléments qu'il faut regrouper sont des polygones représentés par des chaînes cycliques qui représentent leurs contours ; pour cette raison, la fonction de distance entre les représentants des groupes est le coût d'édition des chaînes. Dans le processus de regroupement, il faut calculer un représentant de chaque groupe qui déterminera la règle d'emplacement et le contour, représentant ainsi le groupe d'éléments groupés. Cette forme et ce voisinage moyens sont calculés en considérant les chaînes qui définissent les polygones comme une fonction par parties et en calculant la chaîne moyenne comme la moyenne de ces fonctions. Le processus autorise certaines distorsions dans les formes regroupées et leurs voisinages, mais ne permet pas le regroupement de polygones divisés ou fusionnés.

Le processus d'inférence grammaticale tient compte des texels qui forment la texture et de leurs règles d'emplacement ; à partir de ces règles, les cycles fermés que forment les voisinages sont calculés automatiquement, et les règles de la grammaire de graphe sont générées. À la fin, les productions d'erreur pour tolérer les textures avec distorsion par des texels divisés ou fusionnés sont introduites. Quand la grammaire a été calculée, le symbole qu'elle représente est reconnu au moyen d'un *parser* qui analyse un graphe d'entrée et qui parcourt les nœuds qui vérifient les règles de la grammaire, en les regroupant en un ou plusieurs groupes qui seront les occurrences du symbole dans le graphe présenté.

En résumé, nous proposons des solutions pour segmenter les textures structurées par le moyen du regroupement des formes, et pour modéliser et reconnaître un symbole texturé au moyen de la grammaire et de l'analyse syntaxique que nous pouvons réaliser avec elle sur un graphe et respectivement.

**Mots-clés:** Analyse de documents graphiques, Reconnaissance de symboles, Textures structurées, Reconnaissance syntaxique, Grammaires de graphe.

# Índice General

<b>Agradecimientos</b>	<b>i</b>
<b>Resumen</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Résumé</b>	<b>vii</b>
<b>1 Introducción</b>	<b>1</b>
1.1 Análisis de documentos . . . . .	1
1.1.1 Preprocesamiento . . . . .	3
1.1.2 Segmentación de bloques . . . . .	3
1.1.3 Análisis de bloques de texto . . . . .	4
1.1.4 Análisis de bloques de imágenes . . . . .	4
1.1.5 Análisis de bloques de gráficos . . . . .	4
1.1.6 Análisis de la composición de la página . . . . .	4
1.2 Análisis de documentos gráficos . . . . .	4
1.2.1 Preprocesamiento . . . . .	5
1.2.2 Vectorización . . . . .	5
1.2.3 Separación de texto y gráficos . . . . .	6
1.2.4 Reconocimiento de entidades simbólicas . . . . .	6
1.2.5 Interpretación del documento . . . . .	6
1.2.6 Texturas estructuradas del análisis de documentos gráficos . . . . .	6
1.3 Reconocimiento de símbolos . . . . .	8
1.3.1 Reconocimiento estadístico de símbolos . . . . .	9
1.3.2 Reconocimiento estructural de símbolos . . . . .	10
1.3.3 Reconocimiento de símbolos texturados . . . . .	11
1.4 Análisis de texturas estructuradas . . . . .	12
1.4.1 Definición de textura . . . . .	12
1.4.2 Propiedades . . . . .	13
1.4.3 Objetivos del análisis de texturas . . . . .	14
1.4.4 Texturas repetitivas . . . . .	14
1.4.5 Modelos para texturas fuertemente ordenadas . . . . .	21
1.4.6 Segmentación de texturas estructuradas en documentos gráficos . . . . .	22
1.5 Presentación general del trabajo realizado . . . . .	22

1.5.1	Ámbito y objetivos . . . . .	22
1.5.2	Definición de un modelo estructural para la representación de documentos . . . . .	24
1.5.3	Definición formal de símbolo texturado . . . . .	27
1.5.4	Visión general del trabajo . . . . .	28
1.5.5	Organización de la tesis . . . . .	31
<b>2</b>	<b>Correspondencia de formas 2D y cálculo de la forma media</b>	<b>33</b>
2.1	Las formas 2D en este trabajo . . . . .	33
2.2	Métodos existentes para resolver el problema de correspondencia de formas. . . . .	34
2.3	Correspondencia de cadenas . . . . .	34
2.3.1	Algunas definiciones de la teoría de cadenas . . . . .	34
2.3.2	Representando una forma por medio de una cadena . . . . .	35
2.3.3	Diversas aproximaciones de la correspondencia de cadenas . . . . .	36
2.4	Cálculo de la forma media . . . . .	41
2.4.1	Introducción . . . . .	41
2.4.2	Definición formal de la cadena media . . . . .	45
2.4.3	Un algoritmo para el cálculo de la cadena media que represente formas poligonales. . . . .	47
2.4.4	Comparación de los dos métodos . . . . .	50
2.4.5	Análisis de la complejidad . . . . .	57
2.5	Resultados . . . . .	59
2.5.1	Resultados del cálculo de la forma media entre dos formas sintéticas . . . . .	59
2.5.2	Resultados de calcular la forma media entre dos formas reales. . . . .	62
2.5.3	Resultados de calcular la forma media entre un grupo de formas. . . . .	66
2.6	Conclusiones . . . . .	80
<b>3</b>	<b>Clustering</b>	<b>83</b>
3.1	Presentación del problema y definiciones . . . . .	83
3.2	Características de los datos . . . . .	86
3.2.1	Características de los datos numéricos . . . . .	86
3.2.2	Dimensionalidad . . . . .	87
3.3	Medidas de distancia entre datos . . . . .	87
3.3.1	Formatos para recoger los datos . . . . .	88
3.3.2	Normalización y estandarización de los datos . . . . .	89
3.4	Métodos de clustering . . . . .	89
3.4.1	Clustering jerárquico . . . . .	92
3.4.2	Clustering particional . . . . .	94
3.4.3	Clustering basado en encontrar una distribución de los datos . . . . .	94
3.4.4	Clustering basado en el vecino más cercano . . . . .	95
3.4.5	Clustering basado en lógica difusa <i>fuzzy</i> . . . . .	95
3.4.6	Clustering basado en redes neuronales . . . . .	96
3.4.7	Clustering basado en algoritmos genéticos . . . . .	96
3.4.8	Clustering estocástico . . . . .	96

3.5	Validación e interpretación . . . . .	97
3.6	Diferentes métodos de clustering en análisis de patrones . . . . .	98
3.7	Un algoritmo de clustering para la segmentación de texturas estruc- turadas . . . . .	99
3.7.1	Agrupación según forma: Clustering de formas 2D . . . . .	103
3.7.2	Agrupación según estructura: Clustering de vecindades . . . . .	105
3.7.3	Texturas formadas por más de un tipo de polígono . . . . .	106
3.8	Resultados . . . . .	106
3.9	Conclusiones . . . . .	107
<b>4</b>	<b>Modelización de símbolos texturados</b> . . . . .	<b>111</b>
4.1	Introducción . . . . .	111
4.2	Gramáticas . . . . .	112
4.2.1	Extensión de las gramáticas de cadena, en mayor número de dimensiones . . . . .	115
4.2.2	Gramáticas de vector . . . . .	119
4.2.3	Gramáticas de grafo . . . . .	121
4.2.4	Gramáticas de árbol . . . . .	123
4.2.5	Generalizaciones de gramáticas . . . . .	124
4.3	Corrección de errores . . . . .	125
4.3.1	Gramáticas con información probabilística . . . . .	126
4.3.2	Gramáticas con analizador sintáctico con corrección de errores . . . . .	127
4.4	Análisis sintáctico. (Parser) . . . . .	127
4.4.1	Algoritmo de análisis sintáctico: Earley . . . . .	130
4.4.2	Algoritmo de análisis sintáctico: Earley con corrección de errores . . . . .	132
4.5	Inferencia gramatical . . . . .	135
4.6	Modelización y reconocimiento de símbolos texturados . . . . .	140
4.6.1	Definición del problema . . . . .	140
4.6.2	Definición de una gramática de grafo para representar símbolos texturados . . . . .	143
4.6.3	Inferencia de una gramática de grafo para representar símbolos texturados . . . . .	144
4.6.4	Ampliación de la gramática de grafo con producciones de error . . . . .	145
4.6.5	Reconocimiento de símbolos texturados . . . . .	152
4.6.6	Conclusiones . . . . .	158
<b>5</b>	<b>Aplicación de la segmentación y modelización de símbolos textura- dos en planos de arquitectura</b> . . . . .	<b>159</b>
5.1	Aplicación de referencia . . . . .	159
5.2	Resultados sobre una aplicación real . . . . .	161
5.2.1	Resultados sobre el plano de la Cabaña . . . . .	166
5.2.2	Resultados sobre el plano de Tiro . . . . .	170
5.2.3	Resultados sobre el plano Gaudí . . . . .	179
5.2.4	Resultados sobre el plano Igualada . . . . .	207
5.2.5	Resultados sobre el plano Linaire . . . . .	214
5.3	Conclusiones de los resultados . . . . .	228

<b>6 Conclusiones y perspectivas</b>	<b>231</b>
6.1 Contribución del trabajo presentado y potencial en otras aplicaciones .	231
6.2 Trabajo futuro . . . . .	234
<b>Bibliografía</b>	<b>237</b>
<b>Publicaciones</b>	<b>245</b>

# Índice de Tablas

2.1	Cadena media entre 8 imágenes sintéticas, cogidas de dos en dos, usando el Método-X. . . . .	60
2.2	Cadena media entre 8 imágenes sintéticas, cogidas de dos en dos, usando el Método-L. . . . .	61
2.3	Forma media de 10 formas reales, cogidas de dos en dos, de la base de datos Rutgers tools usando el Método-X. . . . .	63
2.3	Forma media de 10 formas reales, cogidas de dos en dos, de la base de datos Rutgers tools usando el Método-X. . . . .	64
2.4	Forma media de 10 formas reales, cogidas de dos en dos, de la base de datos Rutgers tools usando el Método-L. . . . .	65
2.4	Forma media de 10 formas reales, cogidas de dos en dos, de la base de datos Rutgers tools usando el Método-L. . . . .	66
2.5	Tabla de distancias entre las poligonales <i>cuad2</i> , <i>halfst2</i> , <i>st2</i> , <i>hexa</i> y la forma media que las representa calculada usando el Método-X. . . . .	69
2.6	Tabla de distancias entre las poligonales <i>cuad2</i> , <i>halfst2</i> , <i>st2</i> , <i>hexa</i> y la forma media que las representa calculada usando el Método-L. . . . .	71
2.7	Tabla de distancias entre las poligonales <i>P15</i> , <i>so42</i> , <i>so41</i> , <i>P16</i> y su forma media correspondiente, calculada según el Método-X. . . . .	72
2.8	Tabla de distancias entre las poligonales <i>P15</i> , <i>so42</i> , <i>so41</i> , <i>P16</i> y su forma media correspondiente, calculada según el Método-L. . . . .	75
2.9	Tabla de distancias de edición entre todas las formas <i>ori3</i> , <i>pl6</i> , <i>so42</i> , <i>so41</i> y su forma media calculada usando el Método-X. . . . .	76
2.10	Tabla de distancias de edición entre todas las formas <i>ori3</i> , <i>pl6</i> , <i>so42</i> , <i>so41</i> y su forma media calculada usando el Método-L. . . . .	78
4.1	Operadores binarios de PDL. . . . .	118
5.1	Tabla de las dimensiones de cada grafo, en número de píxeles(x,y) y número de polígonos. . . . .	162
5.2	Tabla con el número de formas que representa cada uno de los clusters por área, forma y vecindad obtenidos en el plano llamado Gaudí. . . . .	182
5.3	Tabla con el número de formas que representa cada uno de los clusters por área, forma y vecindad obtenidos en el plano llamado Linaire. . . . .	215



# Índice de Figuras

1.1	Fases del análisis de documentos. . . . .	3
1.2	Ejemplos de texturas repetitivas estructuradas. . . . .	15
1.3	Ejemplos de símbolos texturados. . . . .	23
1.4	Fases del reconocimiento sintáctico de símbolos texturados. . . . .	25
1.5	Estructuras de datos para representar un plano de arquitectura. . . . .	26
1.6	Representación de símbolo texturado. . . . .	28
2.1	Matriz $D$ calculada mediante la edición y puesta en correspondencia de cadenas con 3 operaciones básicas de edición. . . . .	36
2.2	Matriz $D$ calculada usando el algoritmo de distancia de edición de cadenas usando cadenas cíclicas. . . . .	37
2.3	Matriz $D$ calculada por la puesta en correspondencia mediante la distancia de cadenas con cadenas cíclicas y la operación de edición de fusión. . . . .	38
2.4	Forma representada por medio de una cadena de atributos. Los atributos son la longitud $l$ y el ángulo $\phi$ . . . . .	40
2.5	El problema que aparece al usar la parametrización común según los valores de $x$ . . . . .	50
2.6	Fases del cálculo de la subcadena media $M$ : (a)Conjunto de aristas de $A : A_{i,i+3}$ . (b)Conjunto de aristas de $B : B_{k,k+2}$ . (c)Suma vectorial del conjunto de aristas de $A$ transformadas en vectores. (d)Suma vectorial del conjunto de aristas de $B$ transformadas en vectores. (e)Calculo de la media $m'$ entre la suma vectorial de $A$ y $B$ . (f)Alineación de $A$ y $B$ en $m'$ . (g)Subcadena media, en líneas más gruesas, usando el Método-X. (h)Subcadena media, en líneas más gruesas, usando el Método-L. . . . .	52
2.7	Pasos del algoritmo del cálculo de la forma media a partir de dos formas sintéticas. (a) $Pol1$ . (b) $Pol2$ . (c)Polígono media $Pol1-Pol2$ usando el Método-X. (d)Polígono media $Pol1-Pol2$ usando el Método-L. (e)Arista media ed0-ed0 (para Método-X y Método-L). (f)Arista media ed1-ed1, (para Método-X y Método-L). (g)Alineación de las aristas del $Pol1$ y el $Pol2$ . (h)Aristas medias ed2-Fusión(ed2-ed3) usando el Método-X. (i)Aristas medias ed2-Fusión(ed2-ed3) usando el Método-L. (j)Alineación de las aristas del $Pol1$ y el $Pol2$ . (k)Aristas medias 3-Fusión(4-5) usando ambos métodos. (l)Tabla de la distancia de edición de cadenas entre el $Pol1$ y el $Pol2$ . . . . .	55

2.8	el problema del Método-X en detalle. (a) <i>Pol1</i> . (b) <i>Pol2</i> . (c)Forma media entre <i>Pol1</i> y <i>Pol2</i> usando el Método-X. (d)Tabla de distancia de edición entre <i>Pol1</i> y <i>Pol2</i> . (e)Detalle de la alineación de las aristas 2 y 3 con las aristas 2,3 y 4. (f)Forma media entre <i>Pol1</i> y <i>Pol2</i> usando el Método-L. . . . .	56
2.9	Método progresivo para calcular la forma media. En el último paso podemos ver que la forma $\overline{S1234}$ tiene más influencia que la forma <i>S5</i> . La figura tachada con una cruz sería la forma media obtenida si <i>S5</i> y $\overline{S1234}$ tuvieran el mismo peso. . . . .	58
2.10	Dos ejemplos del mismo problema calculando la forma media de 2 formas reales de la base de datos Rutgers tools.(a) <i>ori1</i> .(b) <i>so41</i> .(c)Media entre <i>ori1</i> y <i>so41</i> usando el Método-X. (d) <i>ori1</i> .(e) <i>pl3</i> .(f)Media entre <i>ori1</i> y <i>Pl3</i> usando el Método-X. . . . .	67
2.11	Cadena media de 3 imágenes sintéticas calculada según el Método-X. (a) <i>cuad2</i> . (b) <i>halfst2</i> . (c) <i>st2</i> . (d) <i>hexa</i> . (e)Polígono media entre <i>cuad2</i> y <i>halfs2</i> . (f)Polígono media acumulado entre Polígono en (e) y <i>st2</i> . (g)Polígono media acumulado entre el Polígono en (f) y <i>hexa</i> . . . . .	68
2.12	Cadena media de 3 imágenes sintéticas calculada según el Método-L. (a) <i>cuad2</i> . (b) <i>halfst2</i> . (c) <i>st2</i> . (d) <i>hexa</i> . (e)Polígono media entre <i>cuad2</i> y <i>halfs2</i> . (f)Polígono media acumulado entre Polígono en (e) y <i>st2</i> . (g)Polígono media acumulado entre el Polígono en (f) y <i>hexa</i> . . . . .	70
2.13	Cadena media de 3 formas reales obtenidas de la base de datos de Rutgers, usando el Método-X. (a) <i>Pl5</i> . (b) <i>so41</i> . (c) <i>so42</i> . (d) <i>Pl6</i> ., (e)Polígono media entre <i>Pl5</i> y <i>so41</i> . (f)Polígono media entre el polígono media acumulado en (e) y <i>so42</i> . (g)Polígono media entre el Polígono media acumulado en (f) ay <i>Pl6</i> . . . . .	73
2.14	Cadena media de 3 formas reales obtenidas de la base de datos de Rutgers, usando el Método-L. (a) <i>Pl5</i> . (b) <i>so41</i> . (c) <i>so42</i> . (d) <i>Pl6</i> ., (e)Polígono media entre <i>Pl5</i> y <i>so41</i> . (f)Polígono media entre el polígono media acumulado en (e) y <i>so42</i> . (g)Polígono media entre el Polígono media acumulado en (f) ay <i>Pl6</i> . . . . .	74
2.15	Cadena media de 3 formas reales obtenidas de la base de datos de Rutgers tools, calculada usando el Método-X. (a) <i>ori3</i> . (b) <i>pl6</i> . (c) <i>so41</i> . (d) <i>so42</i> . (e)Polígono media entre <i>ori3</i> y <i>pl6</i> . (f)Polígono media entre el Polígono acumulado en (e) y <i>so41</i> . (g)Polígono media entre el Polígono media acumulado en (f) y <i>so42</i> . . . . .	77
2.16	Cadena media de 3 formas reales obtenidas de la base de datos de Rutgers tools, calculada usando el Método-L. (a) <i>ori3</i> . (b) <i>pl6</i> . (c) <i>so41</i> . (d) <i>so42</i> . (e)Polígono media entre <i>ori3</i> y <i>pl6</i> . (f)Polígono media entre el Polígono acumulado en (e) y <i>so41</i> . (g)Polígono media entre el Polígono media acumulado en (f) y <i>so42</i> . . . . .	79
3.1	Matriz de patrones. . . . .	88
3.2	Matriz de proximidad. . . . .	88
3.3	Dendrograma. . . . .	92
3.4	Clustering fuzzy. . . . .	96

3.5	Operación de cruce en los algoritmos genéticos de clustering. . . . .	97
3.6	Tres casos distintos de textura estructural. (a)Textura. (b)Polígono( $P = a_0a_1a_2a_3$ ). (c)Regla de emplazamiento ( $R = e_0e_1e_2e_3$ ). (d)Textura. (e)Segmento de línea. (f)Regla de emplazamiento ( $R = e_0e_1e_2e_3$ ). (g)Textura. (h)Punto. (i)Regla de emplazamiento ( $R = e_0e_1e_2e_3$ ). . .	101
3.7	(a)Vecindades con distorsión. (b)Vecindades Polígono $P_5, P_{10}$ . (c)Vecindades Polígonos $P_1, P_2, P_3, P_4, P_6, P_7, P_8, P_9, P_{11}, P_{12}$ . . . . .	107
3.8	Proceso de Clustering. . . . .	108
3.9	(a)Imagen vectorizada (texturas en líneas gruesas). (b)Texel del tejado horizontal. (c)Regla de emplazamiento de (b). (d)Texel del tejado vertical. (e)Regla de emplazamiento de (d). (f)Texel de las escaleras. (g)Regla de emplazamiento de (f). . . . .	109
4.1	Clasificación de símbolos mediante gramáticas. . . . .	115
4.2	Codificación de un patrón mediante una gramática PDL: (a)Patrón. (b)Primitivas a partir de las que generamos el patrón. (c) Patrón codificado usando PDL. . . . .	116
4.3	(a) Símbolo texturado. (b)Representación simbólica del símbolo texturado usando PDL. . . . .	117
4.4	Palabras del lenguaje generadas por una gramática PDL que genera cuadrículas que no se corresponden a la textura de cuadrícula esperada. . . . .	117
4.5	Codificación de un patrón mediante una gramática plex: (a)Patrón. (b)Primitivas a partir de las que generamos el patrón. (c) Patrón codificado usando Plex: (ver,hor,ver,hor)(2100,0220,0012,1001)(). . . . .	119
4.6	Símbolos texturados de diferentes tipos: (a) Con 1 primitiva y emplazamiento en 2D. (b) Con 2 primitivas y emplazamiento en 2D. (c) Con 1 primitiva y emplazamiento en 1D. (d) Con 1 primitiva y emplazamiento en 1D. . . . .	142
4.7	Gramática de grafo que representa el símbolo texturado de la Fig.4.6(a). . . . .	146
4.8	Gramática de grafo que representa el símbolo texturado de la Fig.4.6(c) y de la Fig.4.6(d). . . . .	147
4.9	Gramática de grafo que representa el símbolo texturado de la Fig.4.6(b). Parte (I) . . . . .	148
4.10	Gramática de grafo que representa el símbolo texturado de la Fig.4.6(b). Parte (II) . . . . .	149
4.11	Gramática de grafo que representa el símbolo texturado de la Fig.4.6(b). Parte (III) . . . . .	150
4.12	Posibles distorsiones en la obtención del grafo. (a) Forma distorsionada. (b) Reglas de emplazamiento distorsionadas. (c) Formas fusionadas. (d) Oclusión de formas. . . . .	150
4.13	Gramática de grafo que representa el símbolo texturado de la Fig. 4.6(b). Parte I . . . . .	153
4.14	Gramática de grafo que representa el símbolo texturado de la Fig. 4.6(b). Parte II . . . . .	154
4.15	Gramática de grafo que representa el símbolo texturado de la Fig. 4.6(b). Parte III . . . . .	155

4.16	Proceso de reconocimiento sintáctico de un símbolo texturado. . . . .	157
5.1	(a) Imagen original del plano del alzado de una casa de madera. (b) Resultado de la vectorización (Cabaña). . . . .	161
5.2	(a) Imagen original de un detalle del plano de un campo de tiro al arco. (b) Resultado de la vectorización (Tiro) . . . . .	163
5.3	(a) Imagen original de un detalle del plano de un piso. (b) Resultado de la vectorización (Gaudí) . . . . .	164
5.4	(a) Imagen original de un detalle del plano de un cementerio. (b) Resultado de la vectorización (Igualada). . . . .	164
5.5	(a) Imagen original de un plano vectorizado de una casa. (b) Resultado de la vectorización. (Linaire) . . . . .	165
5.6	Resultados del clustering por área. . . . .	167
5.7	(a) Cluster por área y forma, (b) Forma media asociada de (a). (c) Cluster por área y forma, (d) Forma media asociada de (d). (e) Cluster por área y forma, (f) Forma media asociada de (e). (g) Cluster por área y forma, (h) Forma media asociada de (g). . . . .	168
5.8	(a) Cluster por área, forma y vecindad. (b) Cluster por área, forma y vecindad. (c) Forma junto con su vecindad media asociada a (a). (d) Forma junto con su vecindad media asociada a (c). (e) Cluster por área, forma y vecindad. (f) Cluster por área, forma y vecindad. (g) Forma junto con su vecindad media asociada a (e). (h) Forma junto con su vecindad media asociada a (g). . . . .	169
5.9	Cluster por área . . . . .	171
5.10	Cluster por área. . . . .	172
5.11	(a) Cluster por área y forma, (b) Forma media asociada de (a). . . . .	172
5.12	(a) Cluster por área y forma, (b) Forma media asociada de (a). . . . .	173
5.13	(a) Cluster por área y forma, (b) Forma media asociada de (a). . . . .	173
5.14	(a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . .	174
5.15	(a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . .	174
5.16	(a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . .	175
5.17	(a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . .	175
5.18	(a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . .	176
5.19	(a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . .	176
5.20	(a) Cluster por área, forma y vecindad obtenido uniendo vecindades incluidas unas en otras, (b) Forma junto con su orientación media asociada a (a). . . . .	177
5.21	(a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . .	177

5.22 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	178
5.23 Cluster por Area	181
5.24 Cluster por Area	183
5.25 Cluster por Area	183
5.26 Cluster por Area	184
5.27 (a) Cluster por área y forma, (b) Forma media asociada de (a)	184
5.28 (a) Cluster por área y forma, (b) Forma media asociada de (a)	185
5.29 (a) Cluster por área y forma, (b) Forma media asociada de (a)	185
5.30 (a) Cluster por área y forma, (b) Forma media asociada de (a)	186
5.31 (a) Cluster por área y forma, (b) Forma media asociada de (a)	187
5.32 (a) Cluster por área y forma, (b) Forma media asociada de (a)	188
5.33 (a) Cluster por área y forma, (b) Forma media asociada de (a)	188
5.34 (a) Cluster por área y forma, (b) Forma media asociada de (a)	189
5.35 (a) Cluster por área y forma, (b) Forma media asociada de (a)	189
5.36 (a) Cluster por área y forma, (b) Forma media asociada de (a)	190
5.37 (a) Cluster por área y forma, (b) Forma media asociada de (a)	190
5.38 (a) Cluster por área y forma, (b) Forma media asociada de (a)	191
5.39 (a) Cluster por área y forma, (b) Forma media asociada de (a)	192
5.40 (a) Cluster por área y forma, (b) Forma media asociada de (a)	193
5.41 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	194
5.42 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	195
5.43 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	195
5.44 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	196
5.45 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	196
5.46 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	197
5.47 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	197
5.48 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	198
5.49 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	199
5.50 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	200
5.51 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	200
5.52 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	201
5.53 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	201

5.54 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	202
5.55 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	203
5.56 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	204
5.57 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	204
5.58 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	205
5.59 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	205
5.60 (a) Cluster por área, forma y vecindad, obtenido uniendo vecindades incluidas unas en otras. (b) Forma junto con su orientación media asociada a (a).	206
5.61 Cluster por Area.	208
5.62 Cluster por Area.	208
5.63 (a) Cluster por área y forma, (b) Forma media asociada de (a)	209
5.64 (a) Cluster por área y forma, (b) Forma media asociada de (a)	209
5.65 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	210
5.66 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	210
5.67 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	211
5.68 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a).	211
5.69 (a) Cluster por área, forma y vecindad, obtenido uniendo vecindades incluidas unas en otras. (b) Forma junto con su orientación media asociada a (a)., con mejora de unir orientaciones vecinas incluidas	212
5.70 (a) Cluster por área, forma y vecindad, obtenido uniendo vecindades incluidas unas en otras. (b) Forma junto con su orientación media asociada a (a).	212
5.71 (a) Cluster por área, forma y vecindad, con solapamiento de texturas. (b) Forma de la primera textura encontrada en el área junto con sus orientaciones medias asociadas. (c) Forma de la segunda textura encontrada en el área junto con sus orientaciones medias asociadas.	213
5.72 Resultados del clustering por Area.	216
5.73 Resultados del clustering por Area.	217
5.74 Resultados del clustering por Area.	217
5.75 Resultados del clustering por Area.	218
5.76 Resultados del clustering por Area.	218
5.77 (a) Cluster por área y forma, (b) Forma media asociada de (a)	219
5.78 (a) Cluster por área y forma, (b) Forma media asociada de (a)	219
5.79 (a) Cluster por área y forma, (b) Forma media asociada de (a)	220
5.80 (a) Cluster por área y forma, (b) Forma media asociada de (a)	220

5.81 (a) Cluster por área y forma, (b) Forma media asociada de (a) . . . . . 221

5.82 (a) Cluster por área y forma, (b) Forma media asociada de (a) . . . . . 221

5.83 (a) Cluster por área y forma, (b) Forma media asociada de (a) . . . . . 222

5.84 (a) Cluster por área y forma, (b) Forma media asociada de (a) . . . . . 222

5.85 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 223

5.86 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 223

5.87 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 224

5.88 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 224

5.89 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 225

5.90 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 225

5.91 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 226

5.92 (a) Cluster por área, forma y vecindad, (b) Forma junto con su vecindad media asociada a (a). . . . . 226

5.93 (a) Cluster por área, forma y vecindad, obtenido uniendo vecindades incluidas unas en otras. (b) Forma junto con su vecindad media asociada a (a). . . . . 227



# Capítulo 1

## Introducción

---

En este capítulo vamos a presentar el ámbito en el que se centra este trabajo, que es el análisis de documentos gráficos. Veremos que dentro de este análisis los dos temas que están directamente relacionados con el trabajo presentado son el análisis de símbolos gráficos y el de texturas estructuradas. Definiremos el problema planteado que es la segmentación, reconocimiento y modelización de símbolos gráficos texturados en planos de arquitectura. Esta clase de símbolos será definida formalmente así como las diferentes herramientas que serán necesarias para resolver el problema, como son: la comparación de cadenas, el clustering y las gramáticas formales. Al final se presenta una descripción general del proceso utilizado para segmentar, reconocer y modelizar los símbolos texturados en planos de arquitectura.

---

### 1.1 Análisis de documentos

Los documentos han sido siempre una fuente de información y conocimiento. Actualmente coexisten documentos sobre papel y en formato electrónico que es necesario tratar de una forma eficiente e integrada, por ello los ordenadores deberían poder leer tanto los formatos electrónicos como los que están sobre papel con la misma facilidad, y permitir recuperar esta información para su posterior modificación, almacenamiento, recuperación, reutilización y transmisión. Muchas veces la manera de extraer esta información necesita realizar algunos procesos a mano que resultan costosos en tiempo y dinero. Por esta razón surgió la necesidad de extraerla automáticamente mediante técnicas que utilizaban básicamente los ordenadores y los escáner para tales objetivos. Para poder extraer la información es necesario analizar las imágenes que contienen estos documentos. El análisis de imágenes de documentos, llamado también en su forma abreviada *análisis de documentos*, es el subcampo dentro del procesamiento de imágenes digitales que tiene como objetivo obtener la estructura simbólica de imágenes digitales ya sean escaneadas desde papel o directamente producidas por ordenador. Este tema dentro del procesamiento de imágenes tiene un gran interés y prueba de ello son los congresos y workshops dedicados exclusivamente a él así como el gran número de publicaciones existentes entre las que podríamos destacar, a modo de revisiones

generales, las referencias [21, 79, 80].

Aunque algunos autores, como Tang en [93], llaman al análisis y comprensión de documentos *procesado de documentos*, y dividen ese proceso en dos partes: La extracción de la estructura geométrica del mismo, a la que llaman análisis del documento, y la asociación de una estructura lógica a esa estructura geométrica, a la que llaman comprensión del documento, muchas veces la línea que divide ambos análisis depende del punto de vista, por ello nosotros hablaremos de *análisis de documentos* para referirnos a ambos procesos.

Un sistema de análisis de documentos tiene como objetivo analizar documentos sobre papel y obtener una estructura semántica de los mismos que guardan en un formato electrónico. Así estos sistemas suelen seguir la estructura presentada en la Fig. 1.1, en ella vemos las siguientes fases:

- *Adquisición del documento:* Dado un documento se obtiene una imagen del mismo mediante escáner o cámara.
- *Preprocesamiento:* Son los diversos filtrados que se realizan a la imagen, obtenida del paso previo, para mejorar su calidad, eliminando ruido o distorsiones debidas al propio sensor o a documentos originales degradados. El objetivo de estos procesos de bajo nivel es preparar la imagen para que los procesos posteriores puedan analizarla más claramente. Algunos de estos procesos de filtrado están explicados en Loce [65].
- *Segmentación de bloques:* Divide la imagen del documento en tres tipos de bloques básicos: zonas de texto, de imágenes o de gráficos, para su posterior análisis. Esto es debido a que cada uno de estos tipos de bloques tiene unas características especiales que hace que el análisis a realizar sobre ellos sea diferente.
- *Análisis de los diferentes bloques:*
  - *Análisis de texto:* Normalmente se segmenta en las distintas componentes constituyentes de un texto: como párrafos, líneas, palabras, y se analizan las palabras que se observan mediante reconocedores ópticos de caracteres, en inglés *Optic Character Recognition (OCR)*.
  - *Análisis de imágenes:* Se analizan las imágenes para obtener la información que interese en cada aplicación, por ejemplo segmentación de personas o de objetos, o simplemente una compactación de las mismas para su posterior almacenamiento.
  - *Análisis de gráficos:* Se analizan los gráficos que pueden ser: diagramas, pentagramas musicales, dibujos de ingeniería, planos, etc. Cada uno de ellos según unas características específicas.
- *Análisis de la composición de la página:* En él se combinan los resultados de los análisis de los diferentes bloques encontrados para obtener una descripción global de la página.

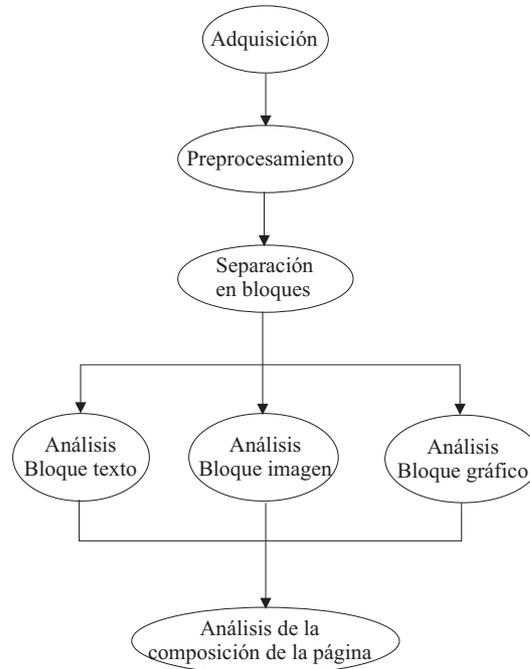


Figura 1.1: Fases del análisis de documentos.

### 1.1.1 Preprocesamiento

Consiste en realizar operaciones sobre la imagen para facilitar los procesos posteriores. Algunos de estos preprocesos se explican en [65, 79], de ellos podríamos destacar la binarización que pasa una imagen de grises a una en blanco y negro, o la detección de la orientación del documento que se realiza normalmente para detectar la orientación de las líneas de texto y reorientarlas para facilitar un posterior reconocimiento de los caracteres.

### 1.1.2 Segmentación de bloques

En esta etapa se identifican los diferentes bloques que aparecen en el documento y se clasifican según sean de texto, de figuras o de gráficos para un posterior análisis especializado en cada una de estas clases. El análisis de bloques se puede realizar según dos enfoques básicos: de arriba a abajo, en inglés *Top-down*, y de abajo a arriba, en inglés *bottom-up*. El primero se basa en el conocimiento, y parte de una predicción de la estructura del documento. Empieza dividiendo el mismo en regiones principales que se irán dividiendo en subregiones. El segundo está dirigido por los datos. La idea es partir de componentes geométricos básicos que se han extraído del documento e ir conectándolos en diferentes grupos en función de sus características, así los grupos se van combinando sucesivamente en grupos mayores.

### 1.1.3 Análisis de bloques de texto

En el análisis de bloques de texto se realiza a veces un proceso previo en el que se intenta determinar la orientación de las líneas de texto. Después se realiza una separación de los diferentes componentes del texto como son, las columnas, los párrafos, las líneas de texto, las palabras y las letras. Una vez hecho esto los *OCR*, se encargan de transformar la imagen de un carácter o una palabra en el código del carácter, como el ASCII o el UNICODE, correspondiente. Existe una amplia bibliografía sobre el tema de OCR. Resúmenes interesantes al respecto pueden consultarse en [21, 77]. Además de la información de las letras en este análisis se pretende obtener la información asociada a ellas como es el tipo de letra, su tamaño, si está o no en cursiva o en negrita, y si presenta algún tipo de estructura como tablas o fórmulas. A menudo estas tablas se consideran medio gráficos y medio texto, por la componente gráfica de sus celdas.

### 1.1.4 Análisis de bloques de imágenes

En este tipo de bloques se podrían realizar los análisis típicos de procesamiento de imágenes como detección de contornos, reconocimiento de formas, etc, pero normalmente se realiza una compresión de la imagen para su posterior almacenamiento.

### 1.1.5 Análisis de bloques de gráficos

Esta clase de bloques pueden presentar diversos tipos de gráficos como los dibujos de ingeniería, los mapas, las partituras musicales, o diversos diagramas. Por ello encontramos básicamente, dibujos lineales que pueden tener zonas en color o rellenas, y texto que normalmente se separa de los gráficos para su análisis por separado. Existen documentos en los que sólo hay bloques gráficos, o en los que éstos son mayoritariamente gráficos, a este tipo de documentos se les llama documentos gráficos. El trabajo presentado se centra en el análisis de este tipo de documentos, por ello hablaremos con mayor detenimiento de ellos en el apartado 1.2.

### 1.1.6 Análisis de la composición de la página

Este último paso relaciona la información obtenida de cada uno de los bloques de texto, de imagen o de gráficos para tener una única estructura del documento. Existen diferentes formatos para representar documentos como son XML, DXF, SGV, etc.

## 1.2 Análisis de documentos gráficos

Dentro del análisis de documentos podemos encontrar algunos que tienen sus bloques mayoritariamente de texto, y a los que se llama documentos de texto, mientras que otros tienen mayoritariamente bloques gráficos y se les llama documentos gráficos. Como hemos dicho anteriormente este trabajo se centra en el análisis de estos últimos y por ello vamos a estudiarlos con más detenimiento antes de entrar en lo que propiamente son objetivos de la tesis.

Existen diversos tipos de documentos gráficos como dibujos de ingeniería, planos del catastro, mapas topográficos, planos de arquitectura, etc. Un sistema de análisis de documentos gráficos lo podemos dividir en las siguientes fases:

- *Preprocesamiento*: Son las operaciones que se realizan sobre la imagen para mejorar su calidad.
- *Vectorización*: Es la extracción y organización de las primitivas que están formando un gráfico consistente en dibujos lineales. Consiste en la extracción y organización de los segmentos de arco y de línea, que aparecen en la imagen, para su posterior análisis.
- *Separación de texto y gráfico*: Consiste en extraer del documento aquellas zonas que corresponden a texto y separarlas de las zonas de gráficos para un análisis posterior por separado. Este paso es importante sobre todo en la extracción de cotas.
- *Reconocimiento de entidades simbólicas*: Los documentos gráficos pueden entenderse como un ensamblado de estructuras simbólicas con un significado dependiente de su contexto, además de anotaciones de texto. Estas estructuras simbólicas, o símbolos, pueden ser notas musicales, en partituras, parcelas, en planos de catastro, mobiliario en planos de arquitectura, etc. Generalmente están constituidas por primitivas con líneas, áreas sólidas, arcos, texturas o tramados, etc. Será sin duda la actividad más importante del reconocimiento del documento y a la que dentro de esta tesis dedicaremos mayor atención.
- *Reconocimiento de texto*: Se analizan las diferentes partes de texto mediante el uso de *OCR*.
- *Interpretación del documento*: Con toda la información del proceso de reconocimiento se determina si los elementos encontrados son coherentes con el entorno en que están, si, por ejemplo las cotas son consistentes en todo el documento, si los símbolos encontrados nos determinan el tipo de zona en que se encuentran etc.

### 1.2.1 Preprocesamiento

Esta fase consiste en realizar unas operaciones sobre la imagen para mejorar su calidad en vistas a los procesos de extracción de características que se realizarán sobre ella, como la vectorización. Algunas de las técnicas empleadas son la umbralización para determinar el nivel de gris que nos servirá de frontera para binarizar la imagen. Otras veces es necesario reducir el ruido de las imágenes mediante filtros, como los morfológicos.

### 1.2.2 Vectorización

El proceso de vectorización consiste en encontrar los segmentos de recta y curva que existen en el plano. A veces esta vectorización consiste solo en obtener segmentos de

línea, ya que los arcos son aproximados por pequeños segmentos de línea. Tombré en [94] explica los diversos pasos del proceso, de los que destacamos el de obtención de las líneas que puede realizarse según tres enfoques básicos: el de esqueletización de la imagen, el basado en poner en correspondencia los dos lados opuestos de la línea de la imagen, es decir, seguimiento de la línea sin esqueletización previa, y el basado en examinar las fronteras de unas ventanas distribuidas por la imagen.

### **1.2.3 Separación de texto y gráficos**

Otra parte importante de la extracción de primitivas es la separación entre las partes de texto y gráficos ya que, aún siendo una parte minoritaria, estas partes de texto aparecen a menudo cerca o incluso interconectadas con las partes de gráficos. En general se pueden discriminar las partes de texto de las de gráficos teniendo en cuenta la distribución espacial de sus componentes y su tamaño, pero cuando el texto está unido a la parte gráfica el problema resulta más complicado. A pesar de su antigüedad, el método de Fletcher y Kasturi [41] sigue siendo una de las referencias clave de esta actividad.

### **1.2.4 Reconocimiento de entidades simbólicas**

El reconocimiento de símbolos y la segmentación y reconocimiento de texturas se puede realizar utilizando diversas técnicas que comentaremos en los apartados 1.3 y 1.4 respectivamente, ya que el reconocimiento de símbolos integrados por texturas constituye uno de los objetivos fundamentales de esta tesis.

### **1.2.5 Interpretación del documento**

Esta fase corresponde al análisis semántico del documento. Dori en [33] hace un estudio sobre las diferencias entre el nivel sintáctico y el semántico en un proceso de reconocimiento de documentos gráficos. El análisis semántico consiste en asignar, a los símbolos encontrados en el documento durante el análisis sintáctico, un significado. Este proceso sería el que se realiza en los planos de arquitectura al comprobar que las cotas encontradas en un documento son coherentes entre sí, o al reconstruir un edificio en 3D a partir de sus vistas 2D.

### **1.2.6 Texturas estructuradas del análisis de documentos gráficos**

El trabajo presentado se centra en el análisis de documentos gráficos, en concreto en los planos de arquitectura donde aparecen dibujos lineales. Dentro de estos documentos gráficos podemos encontrar diferentes tipos de información como son: los símbolos y las texturas. Así el análisis de estos documentos se realiza siguiendo un enfoque bottom-up, es decir primero obtenemos los elementos básicos que componen los objetos compuestos para irlos agrupando y encontrando los objetos compuestos. Por ello el análisis pasa por una primera fase de escaneado, para tener los documentos en formato electrónico, para pasar a una fase de obtención de las primitivas básicas

que son los segmentos, esta fase es la vectorización, de esta forma obtenemos una primera estructura física del documento que correspondería a estos segmentos. Una vez obtenida y almacenada en una estructura de datos se analiza la misma para reconocer los símbolos y las texturas que nos encontramos. Sobre este reconocimiento se puede realizar una interpretación posterior para obtener la estructura lógica que ha de poder interpretar el documento que tenemos sobre papel y generar una descripción de alto nivel para poder tratarla posteriormente. Así las fases de un sistema de análisis de documentos gráficos de este tipo son las siguientes:

- **Adquisición y preprocesamiento del documento gráfico** para obtener una imagen binaria, y en la medida de lo posible, corregida en cuanto a ruido y distorsiones.
- **Vectorización** de la imagen para obtener sus primitivas básicas que son: los segmentos de línea y curva que la forman.
- **Agrupación de las primitivas en bloques gráficos**, para obtener los bloques donde están los símbolos y las texturas.
- **Análisis** de cada uno de los bloques gráficos.
- **Obtención de la estructura simbólica** del documento.

De los diferentes sistemas de análisis de documentos gráficos que existen los que tratan con texturas acostumbran a analizar patrones de líneas paralelas, como en [63] donde utilizan la transformada de Hough, o en [9, 103], o líneas en las que aparece una textura como los puntos y líneas alternados que aparecen en las curvas de nivel de algunos mapas, como en [1, 49]. Pero nosotros no hemos encontrado referencias en las que analicen texturas estructuradas en general como realizamos en este trabajo.

Dentro del análisis de documentos gráficos, el objetivo principal de este trabajo es la segmentación interpretación y modelización de las texturas estructuradas que aparecen en un documento gráfico. El estudio se centra en una aplicación dentro del ámbito de la arquitectura. Durante el estudio de la segmentación de texturas estructuradas en planos hemos observado que éstas, en realidad, definen la estructura de algunos símbolos, es decir definen una parte o el todo de una entidad con un significado asociado. Hasta ahora el término *símbolo* lo hemos visto asociado a estructuras fijas, es decir, con un número de líneas determinado, pero creemos que este término se puede extender a ciertas entidades que están definidas, en parte o totalmente, por texturas, y que tienen también un significado asociado. Algunos ejemplos de esta nueva interpretación del concepto de símbolo serían las escaleras de un plano de arquitectura o las ventanas con diferente número de cristales, o los suelos o terrazas embaldosados, entre otros. En este capítulo vamos a introducir el concepto de reconocimiento de símbolos, haciendo una distinción entre los que están basados en un patrón fijo, que serían los "*símbolos basados en prototipos*" y los que abordamos en este trabajo a los que llamamos "*símbolos texturados*". En el apartado 1.3 veremos que las técnicas utilizadas para el reconocimiento de símbolos clásicos no pueden utilizarse para el de los símbolos texturados y veremos también que una primera aproximación del reconocimiento de estos últimos pasaría por la segmentación de texturas estructuradas. Por ello en el apartado 1.4 explicamos el concepto de textura estructurada

y presentamos algunos métodos para su reconocimiento. Para finalizar en el apartado 1.5 explicamos, de forma resumida, el trabajo que hemos realizado y justificamos los siguientes capítulos que se presentan.

### 1.3 Reconocimiento de símbolos

El reconocimiento de símbolos gráficos es un campo dentro del análisis de documentos gráficos que se encarga de detectar en qué puntos de un documento se encuentran uno o varios símbolos determinados. Un símbolo podría definirse como una entidad gráfica con un significado asociado. El reconocimiento de símbolos gráficos se aplica a diversos tipos de documentos como son:

- **Los esquemas electrónicos:** donde los símbolos pertenecen a un grupo de entidades interconectadas mediante líneas, algunos de los trabajos sobre este campo se encuentran en [47, 51, 58]. Generalmente los símbolos siguen una notación estandarizada y son relativamente fáciles de segmentar en base a las líneas de interconexión y a las estructuras de ciclos.
- **Los mapas:** cuyos símbolos están, a menudo, caracterizados no sólo por su forma sino también por su color. Entre ellos están los mapas del catastro donde aparecen a menudo áreas con texturas formadas por líneas paralelas, algunos ejemplos de reconocimiento de símbolos en este tipo de documentos los podemos ver en [9, 63]. Otro dominio son los mapas geográficos, [78, 83], donde aparecen líneas de nivel, ríos, montañas, y donde a menudo una leyenda indica el significado del símbolo. También podemos considerar dentro de esta amplia categoría los trabajos relacionados con el reconocimiento de instalaciones de servicios como agua, gas, teléfonos, etc [10, 3].
- **Las fórmulas matemáticas:** Son un campo a medio camino entre el reconocimiento de símbolos y los OCR. Donde una parte importante es el reconocimiento del texto de la fórmula, lo que estaría relacionado con los OCR, pero otra no menos importante sería el reconocimiento y posterior validación de la estructura de la misma, algunos ejemplos se encuentran en [53, 57, 59].
- **Los dibujos de arquitectura o de ingeniería:** Presentan símbolos formados por segmentos de líneas o de arco, así como texturas o entramados para definir diferentes características. Podemos encontrar un amplio conjunto de referencias algunas de las cuales aparecen en [2, 4, 62]. Este campo presenta fundamentalmente dos inconvenientes. Primero la inexistencia de una notación estándar, lo que hace que cada plano pueda representar una simbología distinta, y por tanto, diversificar los métodos de reconocimiento. En segundo lugar, los símbolos aparecen incrustados en el documento, es decir, una segmentación previa al reconocimiento es complicada y ambos pasos deben plantearse conjuntamente.
- **Las partituras musicales:** Diversas contribuciones se encuentran en [8, 39] donde sobre un pentagrama en el que aparece un conjunto finito y bien definido de símbolos que son principalmente las notas musicales, se realizan básicamente

tres operaciones. La primera es la segmentación de las líneas del pentagrama que permite la extracción de las notas. La segunda el reconocimiento individual de las notas, y la tercera la interpretación global de la partitura.

- **Los logos:** Pueden ser de diferentes estilos, como lineales o con color o rellenos, e incluso letras caracterizadas por una determinada fuente. Existen diversos trabajos sobre este tipo de símbolos como los mostrados en [23, 31, 90].

El reconocimiento de símbolos está dentro del reconocimiento de patrones en el que típicamente se clasifican los métodos existentes en dos enfoques que son: el estadístico y el estructural. Esta misma clasificación puede usarse para el reconocimiento de símbolos. A continuación vamos a revisar estos dos enfoques.

### 1.3.1 Reconocimiento estadístico de símbolos

En este tipo de reconocimiento cada clase de símbolos se representa mediante un vector  $n$ -dimensional de características obtenidas de la imagen. Así se crea un espacio de  $n$  dimensiones con todos los valores posibles de esas características, en el que cada símbolo es un punto y cada clase una nube de puntos, y se particiona el espacio de manera que cada partición representa una clase distinta. Clasificar un patrón desconocido en este espacio significa calcular su vector de características y ver en que partición está. Un punto clave en este tipo de reconocimiento es una buena selección de las características que nos permita diferenciar claramente unos patrones de otros. La idea es maximizar la distancia entre patrones de clases distintas y minimizarla entre los de una misma clase. Las características, como se explica en [64], pueden basarse en:

- **Tipos de características:**
  - **Píxeles de la imagen:** El vector de características tiene una componente por cada píxel de la imagen, y normalmente la imagen se normaliza primero para obtener un tamaño fijo. Este tipo de características son fáciles de obtener y corresponden a las características visuales de la imagen pero son sensibles al ruido y no son invariantes a rotación ni a escala.
  - **Características geométricas:** Se extraen las características geométricas del símbolo, como son, sus ejes de inercia, área, perímetro, intersecciones, agujeros, centro de gravedad, etc. El vector de características es menor que en el espacio anterior de los píxeles de la imagen. El punto clave, para discriminar correctamente los diferentes símbolos, está en encontrar el conjunto de características con propiedades más discriminantes entre los diferentes símbolos. Se pueden escoger características invariantes a transformaciones afines, pero el ruido afectará a la extracción de características que pueden afectar a la clasificación final del símbolo. Por ello se deberán realizar preprocesos y extracción de las características que sean robustos incluso con la presencia de ruido.
  - **Transformaciones de la imagen:** Se obtienen realizando operaciones sobre la imagen y obteniendo las características de la imagen resultante. Por ejemplo, la transformada de Fourier, la de Fourier-Mellin, etc.

Una vez que se han obtenido las características a tener en cuenta para clasificar los símbolos se debe escoger un proceso para realizar la clasificación propiamente dicha. Para realizarlo existen diversas posibilidades:

- **Tipos de procesos:**

- **Funciones de distancia:** Se basan en calcular distancias entre vectores de características. Para cada clase de símbolos se escoge un representante, es decir, se escoge un vector de características que la represente. Así dado un patrón desconocido, se calculará la distancia entre el vector de características que lo representa y cada uno de los vectores que representan a las diferentes clases de símbolos, asignándole la clase que esté más cerca.
- **k-vecinos más cercanos:** Es similar al enfoque anterior pero ahora se escogen varios representantes para cada clase de símbolos. Para clasificar un patrón desconocido se calculan las distancias del vector de características de éste a los grupos de vectores de las diferentes clases de símbolos, se escogen los  $k$  vectores más cercanos y se asigna la clase que tiene más representantes en ese conjunto.
- **Redes neuronales:** Se realiza un aprendizaje automático de los diferentes símbolos y se seleccionan los parámetros de la red para que, dado un vector de características de entrada, nos devuelva como resultado a que clase pertenece éste.
- **Árboles de decisión:** Son árboles en los que en cada nodo se evalúa una característica a tener en cuenta y se decide qué rama del árbol seguir. Las hojas del árbol corresponden a las diferentes clases de símbolos.

### 1.3.2 Reconocimiento estructural de símbolos

En este tipo de reconocimiento los símbolos se representan por medio de las primitivas simples que lo forman así como las relaciones entre éstas. Las primitivas son características geométricas del patrón como: regiones, segmentos de línea o de arco y puntos, entre otras. Las relaciones entre ellas son también relaciones topológicas de vecindad o inclusión. Esta estructura presenta, de una manera jerárquica, cómo está formado el símbolo, a partir de estas primitivas. Así reconocer un símbolo dado significa construir la jerarquía de formas primitivas que lo representan y comparar ésta con las jerarquías de las clases que queremos reconocer. Muchas veces las primitivas que se han de extraer son segmentos de línea o de arco, y para obtenerlas necesitamos un paso previo de vectorización, además del de obtención de la imagen que muchas veces es mediante escáner. Todo este preproceso puede provocar distorsiones que han de ser tenidas en cuenta, y aceptadas, por el proceso de reconocimiento. Dado que estas estructuras suelen ser de dos dimensiones, la estructura lógica para representarlas acostumbra a ser un grafo. En general la segmentación y el reconocimiento de los símbolos se puede realizar, principalmente, de cinco formas:

- **Isomorfismo de grafos:** Existen muchos trabajos que basan su representación en grafos, como [47, 55, 58, 61]. Una forma de reconocer símbolos representados

por grafos es el isomorfismo de grafos que consiste en buscar subgrafos que estén en correspondencia con un grafo que representa el prototipo del símbolo buscado. Debido a las distorsiones que antes comentábamos estos isomorfismos han de permitir un cierto grado de error ya que el símbolo ideal raras veces se encuentra en el grafo. El problema de este enfoque es el alto grado de complejidad de la puesta en correspondencia de subgrafos, por ello muchos trabajos han explorado la manera de reducir el tiempo de computación.

- **Gramáticas:** Las gramáticas formales permiten modelizar un símbolo mediante reglas gramaticales a partir de las primitivas que lo componen, algunos trabajos relacionados los encontramos en [16, 39, 57]. Estas reglas permiten generar y reconocer la clase del símbolo modelizado. Dado el carácter bidimensional de estos símbolos las gramáticas que se utilizan suelen ser las que permiten una estructura bidimensional, como lo son las de grafo básicamente, pero también existen otras que admiten una estructura similar como las *plex* o *PDL* o las de árbol, etc. Dado que nuestro trabajo se basa en una formulación sintáctica de los símbolos, en el capítulo 4 describiremos con detalle este enfoque.
- **Modelos deformables:** Estos métodos [102] están especialmente pensados para trabajar con entradas de patrones deformados. La idea es modelar una clase de símbolos con un patrón deformable, y buscar el que más se adapte a un símbolo desconocido de entrada. Para ello se deberá minimizar una función de energía del template, que tendrá una energía interna debida al prototipo del símbolo, y una externa debida al símbolo de entrada desconocido.
- **Métodos de propagación de restricciones:** Este método [78, 103] consiste en definir una serie de reglas que modelan las restricciones geométricas de las primitivas que forman un patrón. Estas reglas se definen mediante un lenguaje, y un kernel decide cuáles hay que aplicar en cada caso. En [4] las reglas se estructuran en una red de restricciones y los símbolos se identifican atravesando la red y testeando las reglas en cada uno de sus nodos.
- **Hidden Markov Models (HMM):** En este enfoque, ver [23, 53], los símbolos están representados por una secuencia de estados que generan las características extraídas de la imagen. El proceso de reconocimiento encuentra la secuencia de estados que permite generar, con mayor probabilidad, un símbolo de entrada desconocido. Son capaces de reconocer símbolos con distorsiones.

### 1.3.3 Reconocimiento de símbolos texturados

El trabajo presentado reconoce símbolos texturados dentro de documentos de dibujos de arquitectura o ingeniería. Este tipo de documentos viene definido por segmentos de línea y de curva, que representamos mediante un grafo que contiene estos segmentos y las relaciones entre ellos. En el subapartado 1.5.2 presentamos con detalle este grafo y los preprocesos que realizamos sobre él antes de reconocer los símbolos en los que estamos interesados. Respecto a los símbolos que queremos reconocer están formados por elementos compuestos formados a su vez por texturas y por patrones fijos, como vemos en la Fig. 1.3. Las texturas que aparecen en este tipo de documentos son las

estructuradas, es decir texturas en las que aparecen uno o más patrones que se repiten siguiendo una regla de emplazamiento. Así los elementos compuestos formados por patrones fijos estarán representados por subgrafos de segmentos, mientras que los formados por texturas estarán representados por estructuras que reflejan las primitivas que las forman junto con sus reglas de emplazamiento. La representación formal de un símbolo está descrita en el subapartado 1.5.3. Dado que la representación de los datos es estructural, ya que este es el aspecto que nos interesa de los mismos, su reconocimiento seguirá también un proceso estructural. Así, como los datos están descritos según un grafo, el proceso que utilizamos para reconocerlos y modelizarlos será un híbrido entre el isomorfismo de grafos y las gramáticas de grafos. En el apartado 1.4, describimos qué es una textura estructurada para poder definir formalmente qué es un símbolo texturado en el apartado 1.5.3. Posteriormente presentamos, de manera general, cuál es el proceso que seguimos para segmentar y modelizar estos símbolos, y los temas que se ven implicados en el proceso y por ello son objeto de explicaciones más detalladas en los siguientes capítulos.

## 1.4 Análisis de texturas estructuradas

En esta sección vamos a introducir el concepto de textura y en concreto las texturas objeto de nuestro trabajo que son las estructuradas.

Las texturas formadas por segmentos aparecen en muchos documentos gráficos como son: los planos de arquitectura, de dibujos de ingeniería, planos del catastro o mapas. Todos ellos presentan texturas que nos aportan información sobre la zona donde aparecen o que representan un símbolo, como podría ser una escalera, una pared, o una zona embaldosada. Por otro lado una textura formada por un número elevado de líneas necesita bastante espacio de almacenamiento. Estas dos circunstancias hacen que el estudio y segmentación así como la posterior representación de una textura de forma compacta sean de interés. Por un lado, segmentar la textura nos dará información sobre la zona en la que se encuentra y por otro, modelizarla de una manera compacta nos permitirá almacenarla utilizando menos recursos.

### 1.4.1 Definición de textura

Las texturas se han utilizado en diversas áreas como la ingeniería mecánica, el arte y el diseño, para identificar y describir diferentes objetos, pero en cada área significan una cosa diferente y por ello existen diversas definiciones de las mismas.

El diccionario de María Moliner [76] define una textura como *la manera de estar combinadas o trabadas entre sí las partes, partículas o elementos de una cosa*.

En el análisis de documentos gráficos sólo tratamos las características visuales. Pero aún así existen muchas interpretaciones de lo que es y no es textura. Así Haralick considera que una textura se caracteriza por sus propiedades tonales y por las relaciones espaciales que hay entre ellas. Mientras que Horn considera una textura como una estructura en la imagen que es demasiado fina para ser resuelta, pero lo suficientemente gruesa como para producir fluctuaciones notables en el nivel de gris de las celdas vecinas. Y Hawkins da una definición más detallada, afirmando que la noción de textura depende de tres ingredientes: (1) De algún orden local que se repite en una

región, que es grande en comparación con la medida del orden. (2) Del emplazamiento no aleatorio de partes elementales. (3) Estas partes son entidades uniformes gruesas que tienen aproximadamente las mismas dimensiones en toda la zona texturada.

### 1.4.2 Propiedades

Las texturas tienen un número de propiedades intuitivas que se consideran ciertas:

- Es una propiedad de áreas. La textura de un punto es indefinida.
- La textura está implicada en la distribución espacial de los niveles de gris.
- La textura en una imagen se puede percibir a diferentes escalas o niveles de resolución.
- Podemos percibir que una región tiene textura cuando el número de objetos primitivos es grande.

Las siguientes propiedades juegan un papel importante a la hora de describir una textura en la imagen:

- Uniformidad.
- Densidad.
- Rugosidad.
- Regularidad.
- Linealidad.
- Direccionalidad.
- Dirección.
- Frecuencia.
- Fase.

El hecho de que la percepción de las texturas tenga tantas dimensiones es una razón importante para que no exista un único método de representación que sea adecuado a una variedad de texturas. Por ello surge el problema de diseñar una representación que sea adecuada a una variedad de texturas, y definir paradigmas computacionales que inferan automáticamente esta estructura. Este problema tiene dos aspectos:

- **Cualitativo:** Es el que se refiere al diseño de una representación simbólica y semánticamente relevante de la textura.
- **Cuantitativo:** Se refiere a las medidas de la textura.

### 1.4.3 Objetivos del análisis de texturas

Cuando se analiza una textura se pueden tener los siguientes objetivos:

1. **Clasificación o reconocimiento de texturas:** Producir un mapa de zonas texturadas. Para cada textura de una zona, se extrae su descripción, y se compara ésta con la de las texturas conocidas.
2. **Segmentación de texturas:** En este caso sólo se encuentran los límites de las texturas, sin clasificarlas. Hay dos aproximaciones:
  - (a) *Basada en regiones:* Se intentan identificar las regiones con una textura uniforme. Así, los píxeles o las regiones locales pequeñas se unen en base a la similitud de alguna propiedad de la textura. Esta aproximación tiene la ventaja de dar fronteras cerradas, y que las regiones con la misma textura están siempre bien separadas. Su desventaja es que normalmente se han de especificar el número de texturas diferentes que hay en la imagen.
  - (b) *Basada en fronteras:* Detecta diferencias en las texturas de regiones adyacentes. La ventaja es que no se necesita saber el número de regiones que hay en la imagen. La desventaja es que las fronteras pueden no quedar cerradas, haciendo que dos zonas con diferentes texturas no queden separadas.
3. **Síntesis de texturas:** Se utiliza para comprimir texturas y en aplicaciones de gráficos por computador. El objetivo es reproducir la superficie de un objeto de la manera más realista posible.
4. **Forma a partir de texturas:** A partir de la descripción de la textura observada se puede encontrar la forma tridimensional de los objetos que aparecen en la imagen.

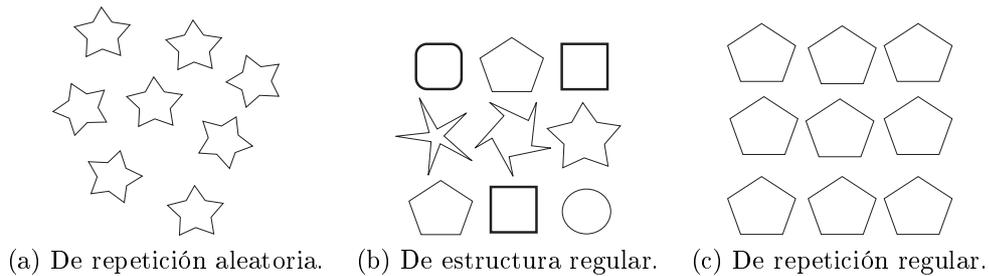
Nosotros nos centraremos en un tipo de texturas que son las fuertemente ordenadas (o texturas repetitivas). Están formadas por un **elemento primitivo** que se repite de acuerdo con unas **reglas de emplazamiento**. A continuación vamos a describirlas con mayor detalle así como a mostrar algunas formas de analizarlas.

### 1.4.4 Texturas repetitivas

Las texturas, en general, se pueden clasificar como:

- **Texturas estadísticas:** Son las que presentan una estructura muy fina, y se pueden describir mejor con propiedades estadísticas de los píxeles de la imagen.
- **Texturas estructuradas:** Se componen de elementos de textura que se organizan según unas reglas de emplazamiento. Dentro de ellas se encuentran las:
  - **De repetición aleatoria:** Se forman emplazando elementos similares de una manera aleatoria. Ver Fig.1.2(a).

- **De estructura regular:** Están formadas poniendo elementos diferentes sobre un patrón regular. Ver Fig.1.2(b).
- **De repetición regular:** Se forman emplazando elementos similares sobre un patrón regular. Ver Fig.1.2(c).



**Figura 1.2:** Ejemplos de texturas repetitivas estructuradas.

En este trabajo nos centraremos en las de repetición regular a las que llamaremos a partir de ahora por la abreviación de **texturas estructuradas o repetitivas**.

Una textura repetitiva tiene una definición cíclica: los elementos primitivos son patrones que se repiten con una cierta frecuencia de repetición, y la frecuencia de repetición es el espacio entre patrones. A veces existen diferentes interpretaciones de una misma textura en cuanto a los elementos que la forman, por ejemplo, según el patrón que escojamos tendremos una regla de emplazamiento u otra, y según la regla de emplazamiento un patrón u otro para una misma textura. Por ello hay técnicas para encontrar la estructura de una textura de estas características que se basan en encontrar primero los patrones y luego sus reglas de emplazamiento, mientras que otras realizan la operación inversa, es decir, primero buscan las reglas de emplazamiento y a continuación los patrones que se repiten. A continuación vamos a ver técnicas que siguen estos dos enfoques.

#### 1.4.4.1. Métodos para encontrar las primitivas

Hay dos tipos de métodos: los que se basan en encontrar el contorno que delimita la zona texturada y los que se basan en encontrar la región.

##### Basados en el contorno

Vilnrotter y otros [104] proponen un método basado en el contorno para construir una descripción simbólica estructural de una textura ordenada. Proponen el uso de un vector de repetición de fronteras, que es similar a una generalización de las matrices de coocurrencia para fronteras. El vector de repetición de contornos se define a lo largo de seis direcciones de la imagen. Es básicamente un histograma de la frecuencia de pares de contornos de orientación similar versus la separación de los mismos. Para texturas repetitivas este histograma muestra una periodicidad en la formación de picos a intervalos regulares. El esfuerzo más grande de este algoritmo

está centrado en describir y clasificar estos picos, para así extraer una descripción simbólica simple. Basándose en el análisis de la repetición de los picos en el vector de repetición de fronteras determinan qué contornos se repiten. Las primitivas de la textura se extraen rastreando la región entre los contornos repetitivos a lo largo de direcciones predeterminadas. Como las primitivas son direccionales han de aparecer para componer una primitiva. El algoritmo funciona bien para texturas naturales, pero tiene algunos problemas. Las direcciones de las fronteras y las del vector de repetición de las fronteras se discretizan en un conjunto pequeño de direcciones. Esto hace que no se explore todo, y que dé formas inusuales para algunos elementos de repetición. Otro problema derivado de la discretización es que no pueden tratar distorsiones sistemáticas, como las producidas por la perspectiva. Por último la reconstrucción de una textura dada no siempre se corresponde con la textura original.

### Basados en la región

Son los más usados. Los primeros métodos propuestos por Tomita, Shirai y Tsuji [95, 96], asumen que el elemento de textura es una región homogénea de nivel de gris. Usan técnicas de crecimiento de regiones simples para aislar estas regiones homogéneas. Leu y Wee [60] usan un método simple donde se hace un *threshold* sobre la imagen texturada original para obtener una imagen binaria.

Hay otros métodos que utilizan el filtro del Laplaciano de la gaussiana  $\Delta^2G$ , pero las interpretaciones del resultado de este filtro varían de las siguientes dos formas:

#### a) Haciendo un Threshold de la respuesta a $\Delta^2G$

Voorhees y Poggio [105] usan esta aproximación considerando los blobs como las partes de la imagen que no corresponden a contornos. Así mientras los contornos corresponden a los cruces por cero de  $I^*\Delta^2G$ , los blobs son las porciones no-cero de la respuesta a  $\Delta^2G$ . Primero hacen un *threshold* de la imagen para eliminar conexiones entre blobs producidas por ruido. Para superar el problema de los blobs conectados por una banda estrecha, usan una técnica de *threshold* múltiple. Así los blobs pequeños se extraen de los thresholds grandes. Estos blobs pequeños crecen sobre las regiones originales para segmentarlas. Las características geométricas: longitud, amplitud, y orientación se calculan para cada blob. Y no se hace ninguna suposición *a priori* de la forma del blob. Las características de este algoritmo son que hace múltiples thresholds a una única escala, y que no se hace ninguna suposición de la forma de los blobs.

#### b) Resolviendo la respuesta a un disco a múltiples escalas

Blostein y Ahuja [14], y Hamey[42] usan esta aproximación. Consideran que los blobs son regiones circulares de intensidad constante. Así la aproximación implica analizar la respuesta de  $\Delta^2G$  a un disco de intensidad circular, y a la inversa, obtener el diámetro y el contraste del disco a partir de la respuesta. El algoritmo se basa en cálculos de las respuestas de una imagen de disco a  $\Delta^2G$  y a  $\frac{\partial}{\partial \sigma}\Delta^2G$ . Así la aproximación de la solución de la respuesta al centro de un disco es:

$$D = 2\sigma \sqrt{\frac{\sigma \left( \frac{\delta}{\delta\sigma} \Delta^2 G * I \right)}{(\Delta^2 G * I)}} + 2 \quad (1.1)$$

$$C = \frac{2\sigma^2}{\pi D^2} e^{\frac{D^2}{8\sigma^2}} (\Delta^2 G * I) \quad (1.2)$$

Donde  $D$  es el diámetro y  $C$  el contraste del disco, y las convoluciones se hacen sobre el centro del disco. Estas ecuaciones se aplican a los extremos de  $\Delta^2 G * I$ . Esto es debido a que a una escala apropiada los extremos locales corresponden a localizaciones del centro del disco. Como hay discos de diferentes diámetros, el algoritmo utiliza un banco de filtros de  $\Delta^2 G$  a diferentes escalas. La ecuación anterior se considera que tiene sentido a una escala dada  $\sigma$  sólo si  $|D - 2\sqrt{2}\sigma| < 2$  píxeles. Esto da un criterio para combinar el resultado de diferentes escalas.

Hamey [42] hace una extensión del método descartando selectivamente las respuestas falsas a crestas y valles de la imagen, pero esto implica hacer un *threshold a priori*, que se podría hacer automáticamente con el método de Voorhees y Poggio. Las características especiales de este algoritmo son: que asume que los blobs son circulares y que el análisis se hace a múltiples escalas de filtros.

#### 1.4.4.2. Métodos para encontrar las reglas de emplazamiento

Como se ha dicho al inicio de este apartado entender un patrón repetitivo es un problema cíclico. Se necesita determinar qué primitivas lo forman para calcular sus reglas de repetición, pero por otro lado, estas primitivas se calculan en función de la naturaleza de la repetición de las mismas. La mayoría de los métodos extraen primero las primitivas, pero Hamey [42] realiza este proceso de forma explícita mediante el concepto de la característica dominante. Primero extrae la característica dominante, y después deriva la estructura repetitiva de las mismas.

Existen diversos métodos para extraer la estructura de las primitivas:

##### 1. Aproximaciones sintácticas

Lu y Fu en [67], una de las primeras referencias sobre análisis sintáctico, utilizan un método sintáctico para analizar las texturas estructuradas. El elemento primitivo se considera que es un píxel único. La estructura se extrae determinando si la textura dada pertenece o no a un lenguaje generado por una gramática hecha *a priori*. La imagen original se divide en ventanas de una medida predeterminada. Para cada ventana se construye un árbol con los valores de los píxeles, y se compara con el árbol generado por la gramática predefinida. Esta aproximación se hace usando la fuerza bruta. Este método tiene algunos inconvenientes: Por un lado sólo se puede aplicar a imágenes binarias, dado que debido a la explosión combinatoria que hace es impracticable para imágenes de niveles de gris. Además es muy sensible al ruido. Por último es dependiente de la medida de la ventana que se escoja.

Subramanian, Revathi y Siromoney en [92] definen las gramáticas Siromoney que son de matriz, y dan entre sus aplicaciones la descripción de texturas de dos

dimensiones formadas por uno o varios patrones representados por su contorno y emplazados según una matriz en el espacio.

## 2. Histogramas del vecino más cercano

Davis [28] presenta un método que calcula la estructura espacial de los elementos de textura. El método trata sólo con patrones de textura que sean puntos. El método tiene un procedimiento heurístico que mide la repetitividad del histograma de la distancia entre un punto y sus  $k$ -vecinos más cercanos. El problema de encontrar el elemento estructurante no se resuelve, y además sólo se consideran rejillas rectangulares y cuadradas.

Tomita, Shirai y Tsuji [95, 96] clasifican las primitivas basándose en las propiedades de: brillo, área, longitud, y curvatura. Los elementos de textura se extraen haciendo un análisis por regiones. Usan la distancia entre centroides de los elementos para derivar una descripción detallada de su emplazamiento. Esto se realiza haciendo un histograma de los vectores de desplazamiento entre elementos de textura. Bajo la suposición de que hay una frecuencia de repetición constante, el histograma muestra clusters de los vectores de desplazamiento que corresponden a las frecuencias de repetición de la textura. Este algoritmo tiene las limitaciones derivadas de tener un esquema de segmentación flojo, y que el método deriva reglas de emplazamiento demasiado simples.

Leu y Wee [60] usan una variante del método del histograma del vecino más cercano, empleando un análisis de agrupación (*clustering*) para abarcar los dos vectores que describen la repetición. Se obtienen buenos resultados para muestras pequeñas de texturas del álbum de Brodatz.

## 3. Matrices de coocurrencia

Es un método estadístico. La mayoría de la investigación hecha en análisis de texturas se basa en este método porque utiliza estadísticas de segundo orden, que es parecido a lo que hace el ojo humano.

Las matrices de coocurrencia de niveles de gris de una imagen forman un espacio 4-dimensional de estadísticas descriptivas indexadas por un vector 2-dimensional  $\delta$  y los escalares  $I_1$  i  $I_2$ .  $C_\delta(I_1, I_2)$ : Es la probabilidad de que la imagen contenga un par de píxeles con intensidades  $I_1$  y  $I_2$  separadas la una de la otra por un vector de desplazamiento  $\delta$ .

Connors y Harlow [26] describen un método para hacer un análisis estructural de texturas usando características que están definidas en matrices de coocurrencia. Observan que las matrices de coocurrencia son altamente diagonales si  $\delta$  es la frecuencia de la textura repetitiva. Esto se deduce de la definición de matriz de coocurrencia, y de que si  $\delta$  es la frecuencia de repetición, entonces los píxeles separados por  $\delta$  han de tener intensidades parecidas. Así calculan la medida de no-linealidad (inercia) en las matrices de coocurrencia para varias  $\delta$ , y localizan el mínimo local de esta función como la frecuencia repetitiva de esta textura. Se pueden obtener resultados similares definiendo medidas directamente sobre la función de correlación dos-dimensional como hacen Rao y Jain [82].

Zucker y Terzopoulos [110] observan lo mismo que los anteriores y usan el test estadístico estándar del *chi cuadrado* para detectar altos niveles de estructura en

las matrices de coocurrencia. Aplican esta técnica con éxito sobre cinco texturas de Brodatz.

Chetverikov [25] utiliza la función del primer momento para medir la diagonalidad de las matrices de coocurrencia. Derivan la respuesta esperada de esta función a una textura binaria, que es una mezcla de elementos de medida regular e irregular. Los parámetros de esta textura se acomodan a texturas repetitivas regulares e irregulares. Chetverikov adapta las curvas derivadas teóricamente a curvas observadas para tres texturas tomadas de Brodatz. Esta técnica determina con éxito la frecuencia de texturas repetitivas regulares.

Davis, Johns y Aggarwal en [29] introducen el concepto de las matrices de coocurrencia generalizadas. Sugieren el cálculo de relaciones de coocurrencia entre características de la imagen en vez de sobre las intensidades de los píxeles.

Starovoitov, Jeong y Park en [91] estudian la extracción de la estructura de una textura con niveles de grises utilizando 22 características basándose en las matrices de coocurrencia binarias. Primero binarizan las imágenes por medio de un threshold, y observan que es posible extraer los vectores de desplazamiento de la textura incluso cuando ésta presenta ruido. Entre las 22 características estudiadas hay una a la que llaman *JF*, *joint feature* que ha sido creada por los autores. Concluyen que el hecho de utilizar matrices de coocurrencia binarias en vez de con niveles de gris reduce la complejidad del problema y permite igualmente detectar la estructura.

Oh, Lee y Shin puntualizan en [81] que las matrices de coocurrencia tienen un alto coste computacional. Así indican que, el cálculo de las matrices de coocurrencia para todos los vectores de desplazamiento, para una textura de medida  $n \times m$ , consume un tiempo del orden  $O(m^2, n^2)$ . Por ello proponen una función de distancia de puesta en correspondencia que sustituya la inercia de una matriz de coocurrencia. Así, dado un vector de desplazamiento y una textura, el valor de la función de la textura con respecto al vector es equivalente a su correspondiente matriz de coocurrencia. Pero en este caso presentan un método eficiente para evaluar la función, ya que para una textura de medida  $m \times n$ , los valores de todas las funciones se pueden calcular en tiempo  $O(mn \log mn)$ .

Kim y Park en [50] describen un método para encontrarla disposición de texturas estructuradas. Suponen una distribución de los elementos de textura en forma de paralelepípedo, cuando se cumple esta característica los elementos de la textura se emplazan siguiendo dos ángulos prominentes. Así, si se obtiene la información de la proyección de la imagen sobre estos ángulos se verá que existen unos picos prominentes. De esta forma se seleccionan los dos ángulos que tienen los picos más prominentes. Una vez hecho esto calculan las matrices de coocurrencia para estos dos ángulos y diferentes distancias, obteniendo de esta forma las distancia de la regla de emplazamiento. Una vez obtienen los vectores de periodicidad encuentran los elementos que se repiten y por consiguiente la representación de la textura. Comprueban que este método es robusto a distorsiones, pero suponen una distribución en forma de paralelepípedo.

Estos métodos y los del histograma del vecino más cercano son muy parecidos.

El cuerpo de estos análisis es la caracterización de la periodicidad y los picos en los histogramas.

#### 4. Transformada de Fourier

La Transformada de Fourier calcula una descripción de toda la imagen basada en la frecuencia. Intuitivamente esta frecuencia ha de capturar la frecuencia repetitiva de una textura regular. Esta transformada mide la energía espacial de las frecuencias de la imagen. Pero si la imagen contiene muestras de repetición con la misma frecuencia, pero con diferentes fases, estas muestras tienden a cancelarse entre ellas.

Bajcsy [11, 6] describe las texturas examinando los *anillos* y los *sectores* del *Espectro de Fourier*. Esta es la base para una clasificación de las texturas en: nodireccionales, monodireccionales o bidireccionales y homogéneas o con blobs. Con esta clasificación las texturas repetitivas regulares se clasifican como bidireccionales.

Matsuyama y otros [71] calculan el *Espectro de Fourier* de una muestra de texturas repetitivas y extraen los picos de este espectro como las frecuencias espaciales de la textura. Para que sirva para texturas con una frecuencia básica estructural con energía baja, se usan los armónicos de frecuencia alta y las frecuencias básicas, para escoger el par apropiado de frecuencias básicas.

#### 5. Aproximaciones basadas en grafos

Las aproximaciones anteriores hacen un análisis global de la textura repetitiva. Pero, en general, no son aplicables a imágenes del mundo real porque se basan en dos suposiciones implícitas:

- La imagen sólo tiene una textura.
- La frecuencia de repetición regular es constante en toda la imagen. Si la frecuencia varía en la imagen, la transformación global combina las diferentes frecuencias en una única transformación. Aunque encontráramos la frecuencia repetitiva los resultados son sólo una descripción del patrón medio de la textura. Además en el mundo real acostumbran a aparecer distorsiones.

Hamey [42] propone un método robusto de la formulación del problema de extraer la descripción estructural a partir de los elementos primitivos. Formaliza ideas que aproximaciones anteriores habían utilizado, y da un significado matemático del uso del vecino más cercano. Usa conocimiento explícito del proceso estructurante para derivar su algoritmo. La idea es crear un grafo de vecindad para las características, y extraer los enlaces repetitivos de este grafo, basándose en restricciones de la teoría de rejillas y de la característica prominente. El algoritmo empieza agrupando las características que parece que van juntas. Esto se hace dando pesos a los arcos del grafo que tienen características como nodos. Los pesos dependen de la prominencia de la característica y de la proximidad de las características que se relacionan. En el siguiente paso sólo quedan los arcos fuertes que determinan la repetitividad local. Esto se hace considerando tres tipos de repetición local: los cruces, las *T*, y las *L*. Se utiliza

una función de evaluación basada en la perpendicularidad de los vectores de la frecuencia fundamental y de acuerdo con los datos, para retener las repeticiones más fuertes. Por último se aplica un algoritmo de relajación para resolver las incompatibilidades. Algunas clases de estructuras repetitivas no las encuentra, y el problema de describir detalladamente las primitivas no lo resuelve.

### 1.4.5 Modelos para texturas fuertemente ordenadas

Existen diversos modelos explícitos para texturas fuertemente ordenadas.

Zucker [109] describe un modelo basado en la noción de primitivas, texturas ideales y texturas observadas. Las primitivas son polígonos regulares. Las texturas ideales se generan concatenando primitivas de manera que cada una está idénticamente rodeada. Esta clase de textura ideal es muy restrictiva.

Ahuja y Rosenfeld [7] describen una clase de modelos de imagen, llamada **mosaico**, basada en procesos aleatorios geométricos. Consideran dos tipos de modelos: los de estructura de celda, y los de cobertura. Los primeros hacen un mosaico del plano con polígonos convexos. El segundo se obtiene posicionando aleatoriamente un conjunto de figuras geométricas en el plano.

Matsuyama [72] propone un modelo en el que definen unos vectores que determinan el emplazamiento de las primitivas. A partir de estos vectores pueden ir a la imagen original para intentar recuperar los elementos de textura que no han podido reconocer en una primera fase. No queda claro como comparan texturas ya que a veces una misma textura en función de los elementos que tengan de la misma puede dar vectores de emplazamiento diferentes.

Hamey [42] propone tres modelos: El de rejilla que se define por dos vectores con un error de distribución del emplazamiento. El modelo de emplazamiento relativo consistente, que es como el anterior pero el error se va acumulando de un emplazamiento a otro. Y el modelo de repetición local, que usa sólo vectores locales. En este trabajo analizan texturas repetitivas que se encuentran en imágenes reales, describiendo el elemento de textura y la estructura repetitiva. Las texturas que ellos admiten pueden tener elementos compuestos de diferentes subelementos, y tienen en cuenta deformaciones provocadas por la perspectiva.

Tuceryan y Jain en [100] segmentan texturas que aparecen en imágenes naturales, y formadas por elementos de textura que llaman *tokens*. Para segmentar la textura calculan los Polígonos de Voronoi, y para cada polígono calculan un vector de características, y estas características se utilizan para etiquetar los tokens identificando las zonas interiores de textura y las de frontera.

Jain y Karu en [44] analizan texturas naturales y las clasifican según unas texturas conocidas, utilizando redes neuronales para construir filtros óptimos. La clasificación se basa en combinar las salidas de estos filtros aplicados.

Lam y Ip en [56] segmentan texturas utilizando un clustering de zonas con un mismo nivel de gris que tengan unas características de área, perímetro, redondeo, ejes de inercia y orientación similares. Para detectar las zonas de un nivel de gris similar utilizan los polígonos de Voronoi, y así calculan los polígonos que aparecen en la imagen. A continuación calculan las características de los mismos y los guardan en una estructura de grafo sobre la que realizan el clustering o agrupación mediante el

uso de una pirámide irregular.

#### 1.4.6 Segmentación de texturas estructuradas en documentos gráficos

Dentro de los documentos gráficos las texturas que aparecen son las estructuradas, y dentro de ellas existen diversos trabajos sobre texturas formadas por líneas paralelas [63] o texturas lineales en las que aparece una línea con una textura formada por puntos y líneas alternadas [1, 49]. Aunque nuestro trabajo se centra en el análisis de los símbolos texturados en documentos gráficos. Estos símbolos están compuestos en su totalidad o en parte por texturas estructuradas de repetición regular, que nosotros llamaremos a partir de ahora, y por abreviar, *texturas estructuradas*. Es decir que consideramos texturas más complejas que las que sólo están formadas por líneas paralelas y diferentes a las lineales. En el siguiente apartado haremos una descripción general del trabajo realizado. Veremos que el método que utilizamos para detectar texturas estructuradas se basa en buscar los elementos que se repiten para posteriormente deducir su regla de emplazamiento. También veremos que el objetivo del trabajo no es sólo segmentar la textura encontrada sino modelizarla para su posterior reconocimiento. También cabe destacar que las texturas no tienen un patrón fijado *a priori* sino que son texturas estructuradas en general, y que podemos encontrar un número indeterminado de ellas en el documento, es decir no esperamos que haya alguna, ni tampoco sólo una, como algunos de los ejemplos que hemos presentado en este apartado.

### 1.5 Presentación general del trabajo realizado

En este apartado vamos a explicar los objetivos de este trabajo así como las diversas técnicas empleadas para desarrollarlo. También introducimos los capítulos que forman esta tesis.

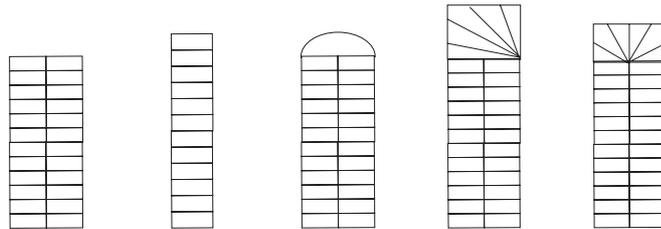
#### 1.5.1 Ámbito y objetivos

Este trabajo se centra en el análisis de documentos gráficos, en concreto planos de arquitectura. Este tipo de documentos se caracteriza por tener principalmente componentes gráficos formados por segmentos de línea y de arco. Estas componentes simples forman estructuras compuestas como son los símbolos gráficos, además del texto. Nosotros nos ocuparemos de los símbolos.

Dentro de los símbolos gráficos hemos visto en este capítulo que existían de dos tipos diferenciados. Unos se basan en un prototipo fijo formado por un conjunto de segmentos que admiten una cierta distorsión, y a los que hemos llamado *símbolos basados en prototipos*, otros están definidos por algunas zonas en las que aparece una textura estructurada, y es ella la que los define, pudiendo tener mayor o menor número de segmentos según la instancia de la misma. A estos últimos símbolos los hemos llamado *símbolos texturados*, y podemos ver algunos ejemplos en la Fig.1.3. Es evidente que los métodos utilizados para reconocer y representar los símbolos basados en prototipos no son aplicables a los texturados, dado que estos últimos no tienen un

número determinado y fijo de primitivas. Estos símbolos texturados deberían reconocerse y representarse mediante técnicas usadas normalmente para el reconocimiento de texturas, aunque ello tampoco es completamente cierto ya que pueden estar formados en parte por zonas que siguen un prototipo además de las texturadas. Por todo ello presentamos un método para reconocer, segmentar, y modelizar símbolos texturados que es un híbrido entre las técnicas utilizadas para los símbolos basados en prototipos y las usadas para las texturas estructuradas. Así los objetivos de esta tesis quedan determinados de la siguiente forma:

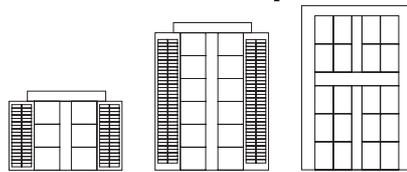
- Definir un modelo para texturas estructuradas.
- Segmentar texturas estructuradas desconocidas *a priori*.
- Inferir el modelo de símbolo texturado automáticamente.
- Reconocer símbolos texturados.
- Aplicación en un ámbito de arquitectura.



(a) Representando diferentes tipos de escaleras.



(b) Representando diferentes tipos de empedrado.



(c) Representando diferentes tipos de ventanas.

**Figura 1.3:** Ejemplos de símbolos texturados.

Este trabajo se enmarca dentro de un trabajo general de análisis de planos. Un esquema general del mismo se presenta en la Fig. 1.4. En ella vemos que hay una primera fase de escaneado del plano y una vectorización del mismo. La vectorización utilizada es la que presentan en [94]. A continuación se construye un grafo,  $H$ , que representa los segmentos obtenidos de la vectorización así como las relaciones entre los mismos. Este grafo está definido en el apartado 1.5.2. Sobre este grafo se realiza

una búsqueda de ciclos cerrados y se construye un nuevo grafo,  $\mathcal{H}$ , que representa estos ciclos cerrados, que son polígonos junto con las relaciones existentes entre ellos, este grafo también lo describimos con detalle en el apartado 1.5.2. Sobre este último grafo es sobre el que hacemos el análisis de los símbolos texturados. Es decir, nuestra estructura geométrica estará representada por este grafo  $\mathcal{H}$ , y por tanto nuestros elementos primitivos básicos serán los polígonos que aparecen en él, mientras que la estructura lógica la obtendremos al analizar este grafo. Los grafos utilizados en este trabajo han sido implementados utilizando la librería LEDA [73].

Antes de definir el proceso general vamos a describir con detalle los dos grafos nombrados anteriormente, y vamos a definir formalmente lo que es un símbolo texturado para finalmente describir el proceso de segmentación, análisis y modelización de los símbolos texturados.

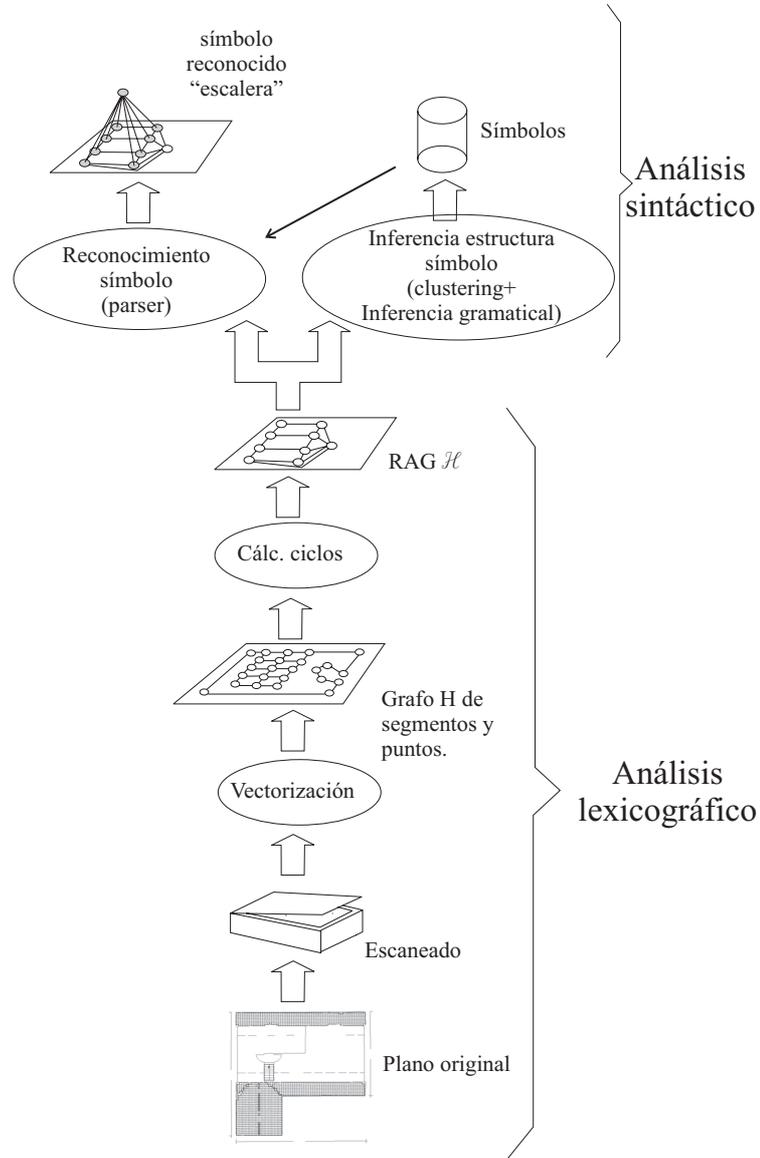
### 1.5.2 Definición de un modelo estructural para la representación de documentos

Dado un plano de arquitectura se vectoriza el mismo, utilizando la vectorización presentada por el grupo de Tombre en [94] que se basa en adelgazar las líneas del plano y seguirlas aproximándolas por segmentos de línea o de arco. Estos segmentos están emplazados en el espacio en dos dimensiones. Como el objetivo es obtener la estructura de los elementos que tenemos en el plano, nos interesa guardar esta estructura  $2D$  que existe entre los segmentos obtenidos, por ello utilizamos un grafo para guardarlos. Así, estos segmentos se guardan en una estructura de grafo etiquetado, en el que los nodos representan los puntos característicos, es decir: puntos terminales, de inflexión y múltiples, mientras que las aristas son los segmentos. Esta estructura puede verse en la Fig. 1.5. Así este grafo quedaría definido de la siguiente manera:

**Definición 1.1** El *grafo de representación de los segmentos del plano* es un grafo etiquetado y con atributos  $H = (V, E, LV, LE, FE)$ , donde:

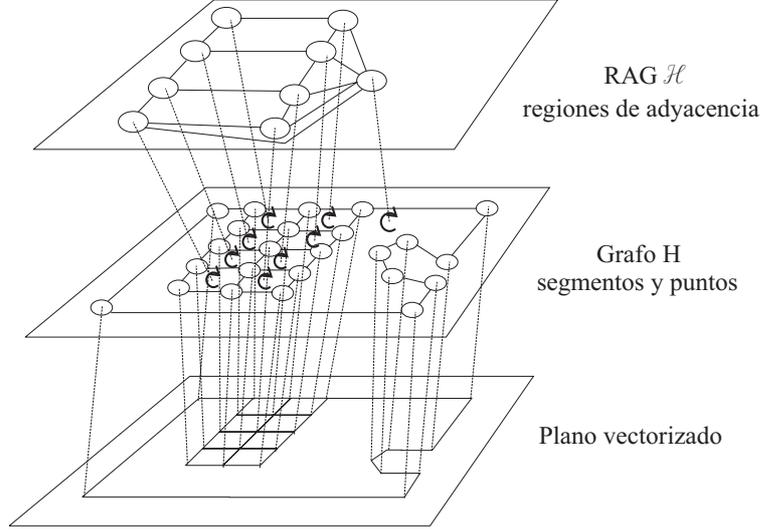
- $V$  es el conjunto de nodos del grafo que representan los puntos característicos del plano.
- $E$  es el conjunto de aristas del grafo, que representan los segmentos del plano.
- $FE$  es una función que le asigna a cada arista del grafo  $e \in E$  un atributo. Estos atributos son su longitud y su ángulo. Así  $FE$  se define como  $FE : E \rightarrow \mathbb{R}, [0, \pi]$ .
- $LV$  es la función que asigna etiquetas a los nodos del grafo, indicando el tipo de puntos que son: terminales, de inflexión o múltiples.
- $LE$  es la función que asigna etiquetas a las aristas del grafo, indicando si son segmentos de línea o de arco.

En este punto podría comenzar el análisis de símbolos buscando los conjuntos de elementos simples que los forman, considerando que estos elementos simples son los segmentos. Pero de hecho estos segmentos se agrupan para formar polígonos y son ellos, en última instancia, los que forman los símbolos que nos interesan. Es decir, una



**Figura 1.4:** Fases del reconocimiento sintáctico de símbolos texturados.

vez que tenemos los segmentos del documento en un grafo necesitamos encontrar los polígonos que están formando. De hecho, los polígonos del plano serán ciclos cerrados dentro del grafo que está representando los segmentos que forman el mismo. Por ello utilizamos el algoritmo de Jiang y Bunke presentado en [48] para encontrar ciclos cerrados en grafos. Estos ciclos cerrados serán los polígonos formados por segmentos. Aquí también nos interesa guardar las relaciones de vecindad existentes entre los



**Figura 1.5:** Estructuras de datos para representar un plano de arquitectura.

polígonos para poder realizar el análisis de su estructura, por ello los guardaremos en otro grafo de atributos. Este grafo tendrá como nodos los polígonos y como aristas las relaciones de vecindad entre ellos, es decir será un grafo de adyacencia de regiones.

**Definición 1.2** El *grafo de adyacencia de regiones* que representa los polígonos del plano se denota como  $\mathcal{H} = (\mathcal{V}, \mathcal{E}, \mathcal{L}\mathcal{V}, \mathcal{L}\mathcal{E}, \mathcal{F}\mathcal{V}, \mathcal{F}\mathcal{V}', \mathcal{F}\mathcal{E})$ , donde:

- $\mathcal{V}$  es el conjunto de nodos del grafo que representan los polígonos que aparecen en el grafo de segmentos, y por tanto en el documento vectorizado.
- $\mathcal{E}$  es el conjunto de aristas del grafo, que representan las relaciones de vecindad de estos nodos.
- $\mathcal{F}\mathcal{V}$  es una función que le asigna a cada nodo del grafo  $v \in \mathcal{V}$  un atributo. Este atributo es una cadena cíclica que representa la secuencia de segmentos que forman la primitiva, en este caso un polígono. Una explicación más detallada de esta representación de cadenas para polígonos la realizamos en el Capítulo 2, concretamente en el apartado 2.3.2. Aquí podemos decir que  $\mathcal{F}\mathcal{V}$  se define como  $\mathcal{F}\mathcal{V} : \mathcal{V} \rightarrow E^*$ , donde  $E$  es el conjunto de segmentos del documento vectorizado, es decir el conjunto de aristas del grafo definido en la Def. 1.1.
- $\mathcal{F}\mathcal{V}'$  es una función que le asigna a cada nodo del grafo  $v \in \mathcal{V}$  un atributo. Este atributo es una cadena no cíclica que representa la secuencia de segmentos que conectan el centro del polígono representado por  $v$  con cada uno de los centros de los polígonos que representan los nodos vecinos a  $v$ . Aquí podemos decir que  $\mathcal{F}\mathcal{V}'$  se define como  $\mathcal{F}\mathcal{V}' : \mathcal{V} \rightarrow C'^*$ , donde  $C'$  es el conjunto de segmentos de recta que unen los centroides de los polígonos vecinos de plano.

- $\mathcal{FE}$  es una función que asigna a cada arista del grafo  $(v, v')$  como atributo el segmento de recta que conecta los centros de los polígonos vecinos representados por  $v$  y  $v'$ . Así  $\mathcal{FE}$  se define como  $\mathcal{FE} : \mathcal{E} \rightarrow C'$ , donde  $C'$  es el conjunto de segmentos conectando los polígonos vecinos del grafo de adyacencia de regiones  $\mathcal{H}$ .
- $\mathcal{LV}$  es la función que asigna etiquetas a los nodos del grafo. Estas etiquetas son números naturales que nos servirán para determinar en los diferentes procesos diversos estados de estos nodos, como visitados o no visitados, a qué cluster pertenecen etc.
- $\mathcal{LE}$  es la función que asigna etiquetas a las aristas del grafo. Estas etiquetas son números naturales que se utilizarán en los diversos procesos de segmentación y reconocimiento de símbolos para determinar si corresponden a vecindades dentro de un mismo cluster, o si las regiones son adyacentes o están separadas, etc.

Esta estructura de grafo se ve reflejada en la Fig. 1.5, donde vemos el proceso completo del cálculo de la estructura de representación del documento, desde la fase en que el documento está escaneado pasando por la vectorización y obtención del primer grafo  $H$  explicado que representa los segmentos, para terminar por el cálculo de los ciclos cerrados del grafo, es decir de los polígonos que aparecen en el documento, y su representación mediante el último grafo explicado  $\mathcal{H}$ .

Una vez tenemos la estructura de representación de un documento vamos a ver qué es un símbolo texturado y cómo lo podemos describir de una manera formal sobre esta estructura de grafo. Una vez definido el símbolo texturado pasaremos a explicar el proceso de segmentación, reconocimiento e inferencia del modelo de los símbolos texturados en documentos gráficos.

### 1.5.3 Definición formal de símbolo texturado

Un *símbolo texturado* podría definirse como una estructura formada por una o más componentes y las relaciones existentes entre ellas. Al menos una de esas componentes debe estar definida por una textura estructurada y el resto pueden ser estructuras fijas o también texturas estructuradas. A su vez las partes con una estructura fija estarán definidas por una serie de regiones o polígonos conectados entre sí según unas relaciones de emplazamiento, y las partes de textura estarán definidas por uno o más polígonos que se emplazarán según unas reglas. Así dentro del ámbito de este trabajo definimos un símbolo texturado formalmente como:

**Definición 1.3** Un *símbolo texturado*  $X$  es un grafo  $X = (K, R)$  donde:

$K$  es el conjunto de nodos que representan componentes del símbolo. Tanto las componentes de estructura fija como las texturadas.

$R$  es el conjunto aristas que representan relaciones entre las componentes de un mismo símbolo o las componentes de otras clases.

Debemos notar que, a su vez, cada elemento de  $K$  puede ser un grafo  $K_i = (K, R)$  tal que:

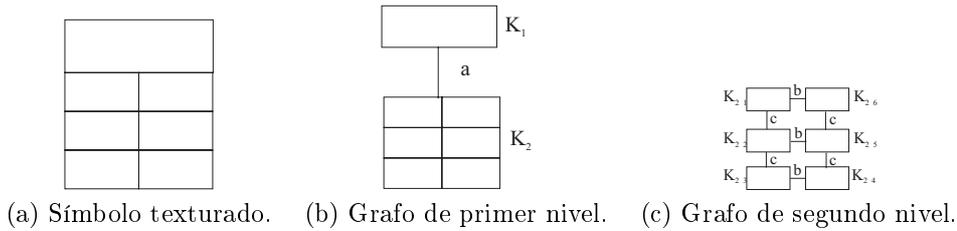
$K$  es el conjunto de nodos que representan:

Si  $K_i$  es una componente de estructura fija,  $K$  contendrá todas las componentes que definen  $K_i$  y  $R$  todas las relaciones existentes entre ellos.

Si  $K_i$  es una componente de textura,  $K$  será el conjunto de polígonos diferentes que forman la misma y  $R$  las relaciones entre ellos. Pero con estos dos conjuntos podrán definirse todas las ocurrencias que se quiera de este símbolo, añadiendo más o menos repeticiones de los emplazamientos de los polígonos que lo forman.

$R$  es el conjunto de aristas que representan las conexiones entre nodos.

Para clarificar esta explicación vamos a ver un ejemplo, en la Fig. 1.6(a) aparece un símbolo texturado que representa una escalera. La Fig. 1.6(b) presenta su representación en forma de grafo donde los nodos son  $K_1$  y  $K_2$ , y sus relaciones  $a$ . en este grafo vemos que  $K_2$  es una parte texturada que a su vez se representa por un grafo mostrado en la Fig. 1.6(c), donde  $K_{11}, K_{12}, K_{13}, K_{14}$  es el conjunto de nodos, y las aristas marcadas con  $b$  o  $c$  es el conjunto de aristas que definen sus vecindades.



**Figura 1.6:** Representación de símbolo texturado.

Una vez tenemos definida la estructura que representa los documentos que utilizamos y lo que es un símbolo, vamos a explicar de manera general el proceso que realizamos para segmentar, analizar y modelizar los mismos de una manera que nos sea útil para tales procesos.

#### 1.5.4 Visión general del trabajo

El problema que presentamos en este trabajo es el de segmentar un símbolo texturado dentro de un documento gráfico, para posteriormente modelizarlo, y de esta manera poderlo guardar de forma compacta y poderlo reconocer.

Como hemos explicado en este capítulo, partimos de un plano de arquitectura que escaneamos, y sobre el documento en formato electrónico que obtenemos, realizamos una vectorización para obtener un primer grafo  $H$  que representa una primera estructura geométrica del documento y que hemos definido en el apartado 1.5. A continuación calculamos los polígonos que existen en el plano, que son los ciclos cerrados que se encuentran en el grafo  $H$ , y los guardamos en un grafo de adyacencia de regiones  $\mathcal{H}$  que hemos definido en el apartado 1.5. Este grafo  $\mathcal{H}$  será la estructura geométrica que consideramos de nuestro documento, y sobre él realizaremos el análisis para obtener una estructura lógica del documento, la segmentación de texturas y el reconocimiento de los símbolos texturados.

El proceso queda como sigue: Primero dado un documento obtendríamos el grafo  $\mathcal{H}$ , a continuación sobre él podríamos realizar diversos procesos:

- Si no tenemos ningún símbolo texturado definido *a priori*:
  - Buscamos las texturas estructuradas que se encuentran en el mismo.
  - Inferimos su modelo
- Si tenemos un conjunto de símbolos texturados definido *a priori*
  - Buscamos todas las texturas estructuradas que aparecen en  $\mathcal{H}$ .
  - Reconocemos qué texturas pueden formar parte de qué símbolos texturados, en función de los polígonos que aparecen en las mismas y de sus reglas de emplazamiento.
  - Realizamos un análisis sintáctico de las zonas escogidas para determinar si pertenecen realmente al símbolo en cuestión y para refinar su segmentación permitiendo alguna distorsión.

De todos estos procesos los puntos clave y desarrollados en este trabajo son la segmentación de texturas estructuradas y la inferencia de su estructura para su posterior reconocimiento.

La segmentación de texturas ha de detectar qué nodos del grafo  $\mathcal{H}$  están formando la textura. Como hemos descrito en el apartado 1.4 el tipo de texturas que encontramos en los documentos que tratamos es de tipo estructurado de repetición regular. Estas texturas se caracterizan por uno o varios elementos, que en nuestro caso son polígonos o segmentos de línea, llamados *textones* que se emplazan siguiendo unas reglas. Por ello encontrar los nodos que forman textura dentro del grafo  $\mathcal{H}$  significa encontrar nodos que representan un polígono o segmento similar con unas vecindades parecidas, es decir, con unas aristas, que representan estas vecindades, parecidas. Esta búsqueda de nodos parecidos con aristas parecidas es, en realidad, una agrupación de nodos vecinos que tengan estas características similares. En realidad podemos llamarlo *clustering* o agrupación pero al tener la particularidad de ser elementos vecinos podemos llamarlo también *region growing* de nodos con unas características de forma y vecindad similares.

Para realizar este proceso de segmentación de texturas estructuradas necesitamos definir detalladamente cómo representamos la silueta de una forma y cómo podemos, en base a esta representación comparar dos formas para saber si son parecidas o no. Esto lo explicamos en el capítulo 2, donde vemos que las siluetas o polígonos los representamos mediante cadenas y la comparación entre ellas la hacemos mediante la comparación de cadenas. Además necesitamos definir cómo agrupamos los nodos usando esta distancia entre ellos. Este proceso de agrupación lo explicamos en el capítulo 3, donde vemos que es necesario tener un representante de cada uno de los grupos encontrados. Un representante de un grupo de polígonos representados por medio de cadenas es en realidad el polígono media, o la cadena media, la explicación del cálculo de esta cadena media de un grupo de cadenas dado se explica en el capítulo 2. Cada uno de los conjuntos de nodos agrupados por este proceso será un grupo de nodos que estará formando una textura estructurada. El representante de una

textura estructurada, ha de ser, como explicamos en el capítulo 3, una forma media, o polígono medio, y una regla de emplazamiento media.

En este punto tenemos la primera parte del proceso que es la segmentación de las texturas. Ahora necesitamos encontrar una forma de modelizar los símbolos texturados, es decir los símbolos que están formados en parte por una textura, para posteriormente reconocerlos. Como explicamos en el capítulo 5 este tipo de símbolos lo definimos mediante un grafo en el que cada nodo representa una parte del símbolo. Estas partes pueden ser partes fijas, o que siguen un patrón determinado, y por tanto se pueden representar a su vez por un grafo, o partes texturadas. La idea desarrollada en este trabajo es cómo modelizar las partes texturadas del símbolo. Una primera idea para realizar este proceso sería representarlas directamente mediante el polígono medio y la vecindad media del mismo. Pero esto tendría problemas a la hora de tratar con texturas distorsionadas, y además resultaría un proceso complicado a la hora de tratar con texturas formadas por más de un textón. Dado que una zona texturada de un símbolo la encontraremos en forma de grafo, la manera natural de representarla es una gramática de grafo, que nos permita generar y reconocer grafos que sean instancias de esa zona texturada. La definición de esa gramática está explicada en el capítulo 4. Una vez tenemos definida la gramática vemos que es necesario inferir esa estructura de forma automática para poder obtener la representación del símbolo texturado una vez lo tengamos segmentado. Para ello debemos inferir una gramática del tipo explicado. El proceso de inferencia gramatical se explica en el capítulo 4. Para finalizar, una vez tenemos la representación de un símbolo texturado necesitamos poder reconocerlo en un grafo  $\mathcal{H}$  dado. Para ello realizamos un isomorfismo de grafos, que se explica en el capítulo 4. En este isomorfismo, al encontrarnos con una zona que ha de ser texturada hemos de realizar un *parsing* de los nodos del grafo para ver si siguen las reglas gramaticales de la gramática que define esa zona texturada, y si es así, qué nodos exactamente son los que la están formando. El proceso de parsing se explica en el capítulo 4.

Así las fases generales de este proceso de reconocimiento sintáctico de símbolos texturados, presentadas en la Fig. 1.4 son las siguientes:

- Análisis lexicográfico: Sería el proceso de paso de documento en papel a documento electrónico y obtención de los elementos léxicos básicos:
  - Escaner de la imagen.
  - Vectorización del documento y obtención de los segmentos. Representación de esos segmentos mediante un grafo,  $H$ , como el explicado en el apartado 1.5.
  - Obtención de los ciclos cerrados del grafo  $H$ , y representación de los mismos mediante un grafo,  $\mathcal{H}$  descrito en el apartado 1.5.
- Análisis sintáctico: A partir de los tokens encontrados en el análisis lexicográfico, que son los ciclos cerrados o polígonos que están en  $\mathcal{H}$ , comprobamos si éstos siguen unas reglas estructurales definidas. Estas reglas se definen en función de un grafo, es decir comprobar si en un grafo  $\mathcal{H}$  dado existe una ocurrencia de un símbolo supondrá realizar un isomorfismo de grafos entre el mismo y la representación del símbolo, este isomorfismo está definido en el capítulo 5. Dentro de

este isomorfismo de grafos tendremos que para cada nodo que representa una parte texturada del símbolo hemos de realizar un parsing de la gramática que define esa textura. Este proceso de análisis sintáctico, o parsing, lo definimos en el capítulo 4.

### 1.5.5 Organización de la tesis

En los siguientes capítulos vamos a explicar los conceptos presentados brevemente en esta introducción, así como justificar su uso para la resolución de nuestro problema.

En el **capítulo 2** presentaremos la manera de representar la silueta de una forma, o un polígono por medio de una cadena. Veremos también como comprar estas cadenas, para finalmente explicar el algoritmo que hemos diseñado para calcular la cadena media, y por tanto la forma media, entendiendo por forma la silueta.

El **capítulo 3** explica la manera de agrupar polígonos, o nodos de un grafo que representan estos polígonos. La agrupación la hacemos de siluetas con una forma parecida y una regla de emplazamiento similar. También se explica como se calcula el representante de un conjunto de formas agrupadas. Entendemos por representante a la forma media que representa a todas las agrupadas junto a su regla de emplazamiento media.

El **capítulo 4** modeliza la parte de textura de un símbolo texturado mediante una gramática de grafo, y explica el proceso de inferencia del mismo a partir del grupo de formas medias con sus correspondientes vecindades medias que representan la textura. Además define el proceso de análisis sintáctico de un grafo para encontrar en él una ocurrencia de esta textura.

El **capítulo 5** presenta el proceso global de segmentación, modelización y reconocimiento de símbolos texturados, así como los resultados del mismo en un entorno concreto como es el de planos de arquitectura.

Para finalizar en el **capítulo 6** presentamos las conclusiones y las posibles vías de continuación del trabajo explicado.

