

UNIVERSIDADE DO ALGARVE
FACULDADE DE CIÊNCIAS E TECNOLOGIA

**EXTRACÇÃO AUTOMÁTICA DE DADOS
GEORREFERENCIADOS A PARTIR DOS PLANOS
CADASTRAIS PORTUGUESES**

(Tese para a obtenção do grau de doutor no ramo de Engenharia
Electrónica e Computação, especialidade de Ciências de Computação)

TIAGO MIGUEL PEREIRA CANDEIAS

Tese orientada por:

PROF. DOUTOR HAMID REZA SHAHBAZKIA

FARO
(2009)

Orientador

Autor

Prof. Doutor Hamid Reza Shahbazkia

Tiago Miguel Pereira Candeias

Extracção Automática de Dados Georreferenciados a partir dos Planos Cadastrais Portugueses

Resumo

Os algoritmos de reconhecimento de imagem são usados para extrair automaticamente a informação de imagens no formato digital. Actualmente, existem numerosos sistemas que convertem documentos do formato de papel para a representação vectorial e têm sido constantemente melhorados durante as últimas duas décadas. Contudo, na digitalização de documentos complexos tais como nos desenhos técnicos, nenhum desses sistemas produziu resultados satisfatórios, uma vez que pretendem ser genéricos e não consideram a semântica do problema, sendo o pós-processamento bastante dispendioso. Esta dissertação apresenta um conjunto de técnicas que simplificam significativamente a extracção automática da informação cadastral a partir dos mapas cadastrais portugueses. A validade da abordagem é demonstrada através da realização de um sistema protótipo, que integra todos os algoritmos de reconhecimento, bem como a validação da informação reconhecida.

Palavras-chave : Reconhecimento de Imagem, Análise de Imagem de Documentos, Processamento de Imagem

Thesis advisor

Autor

Prof. Doutor Hamid Reza Shahbazkia

Tiago Miguel Pereira Candeias

Automatic Extraction of Georeferenced Data from the Portuguese Cadastral Maps

Abstract

Image recognition algorithms are used to extract information from digitized images automatically. Systems designed to convert paper documents into meaningful vectorial representations are numerous nowadays, and have been constantly improved over the two last decades. However, none of these systems seems to be able to provide satisfying results when it comes to convert complex documents such as technical drawings, usually semantic of the problem is not considered and post-processing costs remain high. This dissertation presents a set of techniques that greatly simplifies the automatic extraction of cadastral entities from the portuguese cadastral maps. The validity of the approach is illustrated designing a prototype system, joining all recognition algorithms and validating all information.

Key-words: Image Recognition, Document Image Analysis, Image Processing

Conteúdo

Título	i
Resumo	ii
Abstract	iii
Agradecimentos	viii
Dedicatória	xi
Lista de Figuras	xii
Lista de Tabelas	xix
1 Introdução	1
1.1 Motivação	2
1.2 Objectivos desta Tese	3
1.3 Organização da tese	3
1.4 Importância Nacional deste Trabalho	4
1.4.1 Conceito de Cadastro e sua Importância	4
1.4.2 Cadastro em Portugal	5
1.4.3 Vantagens da Informatização do Cadastro	7
1.4.4 Importância do Trabalho para o Cadastro Nacional	8
1.5 Contribuições	8
2 Visão Geral sobre a Análise de Imagem em Documentos	9
2.1 Introdução	9
2.2 Análise de Imagem em Documentos	10
2.3 Degradação de Documentos	11
2.3.1 Modelos de Degradação de Documentos	12
2.3.2 Modelo de Kanungo et al.	13
2.4 Processamento de Imagem Binária	14
2.4.1 Binarização	15
2.4.2 Propriedades Geométricas	17
2.4.3 Projecções	20
2.4.4 Algoritmos Binários	21
2.4.5 Operadores Morfológicos	31

2.5	Reconhecimento de Imagem	33
2.5.1	Segmentação Texto/Gráficos	34
2.5.2	Reconhecimento de Objectos Gráficos (Vectorização)	35
2.5.3	Reconhecimento de Gráficos Característicos	39
2.5.4	Reconhecimento de Símbolos	39
2.5.5	Reconhecimento de Caracteres	41
2.5.6	Avaliação de Performance	46
2.6	Sumário	49
3	Trabalho Relacionado	50
3.1	Desenhos Técnicos	50
3.1.1	Mapas Cadastrais Portugueses	50
3.2	Investigação na Área	54
3.3	Estratégias Fundamentais	56
3.3.1	Processamento em Imagem Binária	57
3.3.2	Extracção de Informação Sem Pré-Processamento	58
3.3.3	Análise Hierárquica	59
3.3.4	Dados de Treino	60
3.4	Sumário	62
4	Extracção de <i>Features</i>	63
4.1	Detecção de Cruzes	65
4.2	Detecção de Círculos	68
4.3	Detecção de Arcos	71
4.4	Extracção de Linhas a Tracejado	80
4.5	Detecção dos Marcos	87
4.6	Conclusões	91
5	Reconhecimento de Caracteres	92
5.1	Segmentação Texto/Gráficos	93
5.1.1	Mapas Coloridos	93
5.1.2	Mapas Monocromáticos	94
5.2	Métodos de Análise	95
5.2.1	Método de <i>Template-Matching</i>	97
5.2.2	Método de Zoning	99
5.2.3	Métodos de Aprendizagem (K-NN)	100
5.3	Segmentação de Caracteres Conexos	105
5.4	Agrupamento de Caracteres	107
5.5	Correcção Semântica	109
5.6	Conclusões	110

6	Extracção do Contorno	112
6.1	Introdução	112
6.2	Problemas na Extracção do Contorno	114
6.2.1	Descontinuidades	114
6.2.2	Cruzamento de Linhas	115
6.2.3	Elementos Conexos ao Contorno	117
6.3	Soluções para a Extracção do Contorno	118
6.3.1	Métodos de Processamento	118
6.3.2	Algoritmo de Preenchimento com Blocos	120
6.3.3	Algoritmo de Lohmann	122
6.3.4	Algoritmo de <i>Seed-Segment</i>	126
6.3.5	Algoritmo de Seguimento de Linhas	129
6.4	Algoritmo de Vectorização Rosin-West	131
6.5	Solução Geral - Método Composto	131
6.6	Conclusões	133
7	Integração e Interface	135
7.1	Problemas de Integração	136
7.1.1	Abordagem Global	136
7.1.2	Qualidade do Método	137
7.1.3	Generalidade do Método	138
7.1.4	Ligação dos módulos	139
7.1.5	Dificuldades na Implementação	139
7.2	Arquitectura do Software	140
7.3	Organização da Informação	142
7.4	Interface com o Utilizador	143
7.5	Exportação da Informação	144
7.6	Conclusões	146
8	Resultados e Discussão	148
8.1	Amostra	149
8.1.1	Amostra das Secções Cadastrais	151
8.2	Análise dos Módulos	155
8.2.1	Cruzes Georeferênciadas	156
8.2.2	Círculos	159
8.2.3	Arcos	161
8.2.4	Linhas a Tracejado	164
8.2.5	Marcos	168
8.2.6	Caracteres	172
8.2.7	Contorno	175
8.2.8	Taxa de Reconhecimento	177
8.3	Projecto Executado	182

8.3.1	Monchique	183
8.3.2	Tempo de Execução	184
9	Conclusões Sumárias	187
	Bibliografia	189
A	Taxas de Detecção	201

Agradecimentos

Gostaria de agradecer em primeiro lugar, ao Prof. Hamid Reza Shahbazkia pela sua orientação, compreensão e encorajamento em momentos críticos da realização do programa desta tese.

Agradeço à Vera, o amor e compreensão durante este trabalho, mas também o estudo aprofundado sobre a realidade do Cadastro em Portugal e na Europa, o que ajudou a compreender toda a dimensão do Cadastro.

Agradeço à Flávia Mendes, funcionária do Instituto Geográfico Português na delegação de Faro, toda a disponibilidade na demonstração de como é realizada a vectorização do Cadastro Geométrico Rústico de modo manual, assim como ao esclarecimento de muitas dúvidas relacionadas com o Cadastro.

Agradeço à Eng. Vanda Machado, responsável pela execução do cadastro nos distritos do Algarve, Beja e Évora, pela disponibilidade demonstrada em esclarecer algumas dúvidas sobre a vectorização do Cadastro Rústico e ao esclarecimento de muitas dúvidas sobre o Cadastro em Portugal.

Agradeço à Eng. Vanda Bica, responsável pela conservação do cadastro Português, a apresentação de uma grande variedade de mapas cadastrais existentes em Portugal, a explicação da sua semântica e a partilha das principais dificuldades na digitalização dos mapas cadastrais Portugueses.

Agradeço à Eng. Maria Gabriela Rocha, responsável pela execução cadastral na digitalização dos mapas cadastrais Portugueses, o esclarecimento sobre as especificações do Instituto Geográfico Português e a informação sobre problemas de digitalização e integração num sistema de informação geográfico.

Agradeço à Eng. Teresa Costa Mira, bem como às suas duas colegas a quem agradecei

anteriormente, a reunião que tivemos e as suas sugestões para melhorar o meu sistema protótipo de extracção automática.

O meu agradecimento vai também para a Prof. Ana Lopes pela disponibilidade no fornecimento de alguns mapas cadastrais e por explicações iniciais sobre o cadastro Português.

Gostaria de agradecer à Câmara Municipal de Monchique, nomeadamente ao Eng. Eduardo Duarte e ao Eng. Montez, toda a disponibilidade em mostrar o sistema cadastral da Câmara e à disponibilização de mapas cadastrais para teste.

Também agradeço aos meus colegas Filipe Tomaz e Rúben Oliveira pela colaboração e troca de ideias no início deste projecto.

Eu agradeço fundamentalmente aos meus pais pelo amor, compreensão, aconselhamento, encorajamento e suporte em todos os momentos da minha vida.

Os meus amigos merecem uma menção especial pelo encorajamento e suporte emocional.

Gostaria de agradecer ao Instituto Geográfico Português pelo fornecimento de mapas em formato raster e respectiva vectorização manual, o que se revelou fundamental para testar o sistema e verificar a taxa de reconhecimento do protótipo.

Também gostaria de agradecer ao Projecto ACID (Automatic Cadastral Information Digitalization) financiado pela FCT (Fundação para a Ciência e Tecnologia), pelo financiamento parcial durante a realização deste trabalho (contrato SRI-34257-99-00).

Finalmente, agradeço enormemente à iniciativa Neotec, administrada pela Agência de Inovação, S.A., todo o apoio financeiro prestado na fase final deste projecto. Possibilitando a formação em sistemas de informação geográfica, a aquisição de material de apoio,

consultadoria em gestão e ao financiamento para o início da actividade empresarial, dando assim origem à constituição da empresa Digital View, Unip. Lda.

*Dedicado à Vera,
à Mariana
e aos meus pais.*

Lista de Figuras

1.1	Esta imagem apresenta um mapa administrativo de Portugal Continental com as idades de execução das secções cadastrais.	6
1.2	Esta figura mostra no lado esquerdo a informação geográfica patente numa secção cadastral e no lado direito a informação alfanumérica numa ficha de prédio (imagens extraídas da apresentação <i>A Informatização do CGPR</i> , na conferência internacional, GIS Planet 2005, realizada no Estoril, de autoria da Eng. ^a Vanda Machado, in site www.igeo.pt).	6
2.1	Modelo de distorção morfológica local: (a) Representação ideal; (b) Transformada da distância do <i>foreground</i> (a); (c) Transformada da distância do <i>background</i> (a); (d) Imagem com o resultado da perturbação usando a distribuição exponencial de probabilidade $P(0 d, \beta, f) = P(1 d, \alpha, b) = e^{-\alpha d^2}$ e $\alpha = \beta = 2$, $\alpha_0 = \beta_0 = 1$, $\eta = 0$; (e) Resultado do <i>closing</i> em (d) por um elemento binário estruturante de tamanho 2×2	15
2.2	A imagem binária da palavra CA com a sua projecção horizontal e vertical.	20
2.3	As vizinhanças-4 e -8 de uma imagem rectangular. O pixel [i,j] está localizado no centro de cada figura.. . . .	22
3.1	Visão geral de uma secção cadastral digitalizada. No lado esquerdo podemos observar um mapa cadastral rústico digitalizado com tamanho de 10784×7853 pixels e resolução 300 dpi. No lado direito podemos visualizar os elementos que constituem um mapa cadastral, nomeadamente círculos, arcos, cruces, linhas contínuas, linhas a tracejado e texto. A cada elemento está associada uma semântica.	53

- 4.1 É possível distinguir uma cruz do *background* de diferentes maneiras, usando:
a) quatro segmentos de recta, b) dois segmentos de rectas oblíquos ou c) dois segmentos de recta perpendiculares. De notar também, que as linhas contínuas representam o *match* com os pixeis a preto, enquanto as linhas a tracejado representam o *match* com os pixeis a branco. São definidos quatro pequenos segmentos de recta nas extremidades da cruz para garantir a simetria deste elemento. 66
- 4.2 As cruzes são apagadas dos mapas cadastrais utilizando máscaras, isto é, seguindo o eixo de cada segmento de recta e apagando na direcção perpendicular ao segmento em análise, apenas se não existir qualquer elemento conexo a essa parte da cruz. 68
- 4.3 A detecção do círculo está baseada na extracção de características, nomeadamente a) a análise da percentagem de *match* (p_{match}), b) a análise da taxa de continuidade máxima (p_{cont}) e a taxa de descontinuidade máxima ($p_{descont}$) e c) a detecção de elementos dentro (p_{dentro}) e fora do círculo (p_{fora}). A combinação desta análise de características culmina na equação 4.5, que calcula a probabilidade de existência do círculo. 70
- 4.4 Esta figura mostra a sobreposição do traçado do círculo teórico sobre o arco detectado na fase anterior. Seguindo a trajectória do círculo são encontrados os pontos extremos do arco, nomeadamente p_i e p_f . Neste caso o arco apresenta-se com uma trajectória perfeita, não existindo descontinuidades no *match* entre o arco e o círculo. 75
- 4.5 Esta figura mostra que a sobreposição das trajectórias do círculo e do arco não é perfeita e apresenta algumas descontinuidades. No entanto, as pequenas descontinuidades são eliminadas de modo a obter-se o maior arco possível e assim determinar-se os seus extremos. 76
- 4.6 Na figura observamos que a partir dos pontos extremos do arco, A e B , encontramos um terceiro ponto pertencente ao arco (F); partindo de um segmento de recta CD que está perpendicular ao segmento AB e de seguida encontrando o ponto médio de DE situado no centro do traçado do arco. . . 77
- 4.7 Esta figura mostra o processo iterativo para a determinação do centro do arco. Partindo dos três pontos iniciais do arco A, B e C são encontrados centros O_n para as bissectrizes de intersecção dos pontos do arco. Analisando a dispersão dos centros é finalmente encontrado o centro mais provável. 78
- 4.8 A figura mostra o traçado teórico do arco com os parâmetros previamente encontrados. Esta trajectória é utilizada para calcular a percentagem de *match* entre os dois arcos. Percorrendo o traçado do arco de p_i até p_f através de p_k e analisando o arco candidato sobre as rectas l_{k1} e l_{k2} , são encontradas características do arco, nomeadamente a simetria central, interferência ao ruído e conexão com outros elementos. Finalmente, ocorre a decisão sobre a validade do arco candidato. 79

4.9	A detecção de linhas a tracejado está baseada em três factos, nomeadamente um tracejado tem um tamanho e forma característica, tem pelo menos um vizinho com uma orientação semelhante à sua e um conjunto de tracejados segue uma linha virtual.	82
4.10	Esta figura mostra a distinção relativa à densidade entre o carácter '1' e um tracejado. Considerando que o tracejado apresenta um densidade próxima de um rectângulo completamente preenchido é assim possível detectar estes elementos de acordo com esta característica.	83
4.11	A ligação dos tracejados elementares às junções (tracejados elementares compostos) é realizada considerando a proximidade dos extremos dos tracejados elementares aos extremos das junções.	86
4.12	Imagens do pré-processamento da extracção das linhas a tracejado.	87
4.13	Validação e vectorização das linhas a tracejado.	88
4.14	A detecção dos marcos em imagens a cinza é implementada através do processo da convulsão utilizando o <i>kernel K</i>	90
4.15	A detecção dos marcos em imagens coloridas é implementada através do processo da convulsão utilizando o <i>kernel K</i>	90
5.1	Imagens representativas da segmentação dos elementos gráficos baseado na cor.	94
5.2	Imagens representativas da segmentação de uma imagem monocromática baseada na área e conexão dos elementos constituintes da imagem.	96
5.3	A determinação dos pontos de <i>match</i> no método de <i>template matching</i> considera os pontos de esqueleto do carácter e os pontos do <i>outline</i> do carácter, mais a condição de que dois pontos de <i>match</i> devem estar a uma distância mínima de três pixels.	98
5.4	Exemplo da análise realizada pelo método de Zoning.	100
5.5	Primeiramente os elementos não conexos são identificados com vista à sua classificação.	102
5.6	Os caracteres são classificados manualmente, construindo-se uma base de dados de <i>templates</i> reconhecidos. De notar que cada carácter pode apresentar vários <i>templates</i> de modo a obter-se maior robustez.	102
5.7	Esta imagem apresenta uma lista de features que são utilizadas para classificar os símbolos.	103
5.8	O treino do sistema de reconhecimento de caracteres é realizado utilizando o classificador K-NN em conjunto com algoritmo genético para otimizar os pesos das <i>features</i>	104
5.9	Esta figura apresenta o resultado da classificação automática. Na tabela da figura estão agrupados os símbolos em classes, apresentando-se a verde os símbolos classificados manualmente e a rosa os símbolos classificados automaticamente.	105

5.10	Como resultado final é obtido um ficheiro XML com as classificações de cada símbolo, a sua probabilidade, bem como todos os valores do vector de <i>features</i> de cada um.	106
5.11	Esta figura apresenta o histograma vertical e horizontal dos caracteres conexos 'CA'. A análise é feita em ambos os eixos de modo a encontrar as linha potenciais de segmentação. Este procedimento é apenas aplicável nos casos em que os elementos apresentam na zona de conexão uma fraca densidade de pixels.	106
5.12	O algoritmo de agrupamento de caracteres considera o rectângulo que define os limites de cada caracter. Considerando ambos os eixos de forma independente, são agrupados os elementos que apresentam uma distância euclidiana próxima e uma orientação semelhante entre dois caracteres. No final ambos os caracteres são agrupados e é definido um novo rectângulo que os engloba no seu todo.	108
6.1	Exemplos de alguns problemas na extracção do contorno, nomeadamente descontinuidades no contorno, elementos conexos ao contorno, cruzamento de linhas e também é necessário associar a cada círculo um contorno. . . .	113
6.2	Esta imagem mostra dois tipos de descontinuidades no contorno: o lado esquerdo apresenta uma descontinuidade simples enquanto o lado direito tem duas descontinuidades muito próximas.	115
6.3	A detecção dos segmentos de recta através da transformada de Hough não resolve o problema da extracção do contorno devido ao contorno apresentar descontinuidades e curvas.	116
6.4	A aplicação do algoritmo de <i>skeletonization</i> origina distorções quando existem elementos conexos ao contorno.	117
6.5	Um exemplo da aplicação do algoritmo de preenchimento, utilizando blocos de diferentes tamanhos (N).	120
6.6	Um exemplo da aplicação do algoritmo de preenchimento utilizando blocos.	121
6.7	Na primeira iteração do algoritmo de Lohmann são detectados os extremos do círculo máximo centrado na posição do máximo local.	123
6.8	O resultado do algoritmo de Lohmann aplicado ao contorno com grandes descontinuidades.	125
6.9	O teste dentro do prédio verifica se um círculo está dentro ou fora da região.	126
6.10	O segmento padrão (<i>seed-segment</i>) é um rectângulo que descreve as características elementares de um segmento de recta, nomeadamente a largura de linha e a sua orientação.	127
6.11	Um exemplo da detecção de <i>seed-segments</i> descrevendo as características básicas da linha, nomeadamente a orientação e a largura. Estes <i>seed-segments</i> são os padrões básicos para realizar o seguimento da linha no processo de detecção das linhas rectas.	128

-
- 6.12 Esta figura mostra o resultado final do algoritmo de *seed-segments*, após a aplicação da segmentação dos elementos conexos relativamente ao contorno. 129
- 6.13 No algoritmo de seguimento de linha, é sempre tomada a mesma direcção em todos os pontos de intersecção. 130
- 6.14 Vectorização das linhas do contorno utilizando o algoritmo de Rosin-West. . 132
- 7.1 Esta imagem apresenta uma amostra com alguns detalhes das características dos mapas cadastrais. No processo de implementação, analisa-se o universo dos mapas cadastrais considerando todas as características e detalhes, identifica-se o que têm em comum, estabelecendo padrões de forma a generalizar e dar robustez ao sistema. 137
- 7.2 Esta imagem apresenta a ligação de todos os componentes de software do sistema de reconhecimento. Existe um *backend* com os métodos de análise que realizam a detecção de entidades básicas para os processos de detecção e que posteriormente enviam os dados para o interface, no sentido de visualizar e exportar essa informação para ficheiros no formato XML, vectorial ou de imagem. 142
- 7.3 A informação detectada é apresentada num ficheiro XML de modo estruturado. A detecção considera as entidades cadastrais com os seus atributos. De notar que a posição do círculo é a chave da entidade cadastral. 143
- 7.4 O interface apresenta a sobreposição da imagem cadastral com a informação detectada. Deste modo pode-se visualizar a detecção, realizar a validação e facilmente otimizar o desenvolvimento do sistema. 145
- 7.5 Esta imagem apresenta o resultado da vectorização integrado no sistema de informação geográfico QGIS. Os dados podem ser georeferenciados e sobrepostos à CAOP que contém os limites administrativos de Portugal. Deste modo a informação cadastral está pronta a ser utilizada para os mais diversos fins. 146
- 7.6 Nesta imagem estão evidenciados os atributos da entidade cadastral. É de notar que a área do prédio é atribuída automaticamente pelo SIG visto ser um *layer* de polígonos. Estes dados ao encontrarem-se colocados num sistema de informação permitem realizar consultas (*queries* à base de dados, de modo a obter a informação solicitada. 147
- 8.1 Esta figura apresenta a distribuição por concelhos das secções cadastrais presentes na tabela 8.1 no mapa administrativo de Portugal Continental. . . 151
- 8.2 Esta imagem mostra os períodos de execução das secções cadastrais por concelho em função das áreas administrativas de Portugal continental. . . . 153
- 8.3 Esta imagem apresenta a distribuição por freguesia da quantidade de secções cadastrais, no mapa administrativo de Portugal Continental. 154

- 8.4 O exemplo de uma cruz bem detectada e outro de uma cruz não detectada devido ao efeito da degradação da imagem. Para ser correctamente detectada a cruz necessita de estar bem formada e de se distinguir do *background*. 157
- 8.5 Nesta figura pode ser observada a detecção das cruzez georeferenciadas. A detecção destes elementos está representada com a cor amarela. Observar-se ainda que o conjunto das cruzez está disposta na forma de uma grelha. . . 158
- 8.6 Esta imagem apresenta um gráfico de barras que mostra as taxas de detecção e de falsas detecções no processo de reconhecimento das cruzez. 159
- 8.7 Exemplo da detecção dos círculos numa secção cadastral. Neste caso pode-se observar que os círculos apresentam uma forma elipsoidal e não a estrutura de círculos perfeitos. 160
- 8.8 Um exemplo da detecção global dos círculos. Estes elementos foram detectados e os seus parâmetros de detecção estão representados a azul. . . . 160
- 8.9 Esta imagem apresenta um gráfico de barras que mostra as taxas de detecção dos círculos. 161
- 8.10 Duas figuras representativas dos problemas na detecção dos arcos. 162
- 8.11 A detecção dos arcos está representada com círculos cor de rosa, que se apresentam conexos ao contorno do prédio. 163
- 8.12 Esta imagem apresenta os níveis de detecção e de falsos alarmes dos arcos. De notar que nos casos onde não está definida a percentagem de detecção deve-se a não existirem arcos nessas secções cadastrais ou a não terem sido considerados (ver tabela A.3). 164
- 8.13 Esta figura mostra um exemplo da detecção dos tracejados elementares. Esta detecção apresenta alguns traçados não detectados (que podem ser observados onde está o sinal de interrogação) nomeadamente no caso das junções e dos tracejados com forma de curva ou que não formam um rectângulo completamente preenchido. 165
- 8.14 Esta figura apresenta a visão geral sobre o processo de ligação dos tracejados elementares. De notar que os tracejados elementares não reconhecidos apresentam-se dentro de um rectângulo. 166
- 8.15 Nesta imagem pode observar-se as dificuldades da ligação dos tracejados elementares. Os casos problemáticos acontecem na ligação das junções e dos tracejados elementares não detectados. 167
- 8.16 Esta imagem apresenta um gráfico de barras que mostra as taxas de detecção dos tracejados. 168
- 8.17 Esta imagem apresenta uma boa detecção dos marcos. 169
- 8.18 Esta imagem apresenta a detecção dos marcos na secção 080301_n1. De notar que são detectados dois marcos muito próximos devido à dilatação do contorno nalgumas zonas. 169
- 8.19 Esta imagem apresenta a detecção dos marcos na secção 081101_o. Neste caso a detecção dos marcos não é aplicável devido às características excepcionais desta secção. 170

- 8.20 Esta imagem apresenta a detecção dos marcos na secção 1209035_o. De notar que sobre o contorno aparecem marcos com a cor vermelha escura e que depois da segmentação baseada na cor vermelha padrão, a cor de dentro do marco é perdida, por se apresentar muito próxima da cor preta. Assim, os marcos não são detectados. 171
- 8.21 Esta imagem apresenta um gráfico de barras que mostra as taxas de detecção dos marcos. 171
- 8.22 Exemplos do reconhecimento de caracteres. 172
- 8.23 Nesta figura é apresentado o processo de agrupamento de caracteres, sendo os caracteres agrupados nos eixos vertical e horizontal, tendo em consideração a vizinhança de cada um. 173
- 8.24 Exemplo do reconhecimento e classificação dos caracteres. 174
- 8.25 Esta imagem apresenta um gráfico de barras que mostra a taxa de detecção dos caracteres e a taxa de caracteres utilizados no treino do sistema. 175
- 8.26 Esta figura mostra a vectorização do contorno dos prédios. 176
- 8.27 A extracção do contorno é um processo problemático no caso dos prédios com uma área muito reduzida, onde a informação apresenta-se conexas. 177
- 8.28 Esta figura apresenta uma visão geral sobre a extracção do contorno. 178
- 8.29 Esta imagem apresenta os níveis das taxas de detecção dos contornos. 178
- 8.30 Esta figura mostra a vectorização das extremas de prédio numa secção cadastral do concelho de Monchique. 183
- 8.31 Esta imagem apresenta a sobreposição de vários secções cadastrais, com uma camada de polígonos sobreposta, que representa as extremas de prédios. 184
- 8.32 Nesta figura pode observar-se uma consulta no sistema de informação cadastral, de modo a obter os atributos de um polígono, que representa um prédio. 185
- 8.33 A sobreposição do mapa administrativo do Algarve permite identificar todas as vantagens da informatização do cadastro predial, ou seja, realizar consultas e encontrar os proprietários das propriedades de uma forma automática e centralizada. 185
- 8.34 Esta imagem apresenta um gráfico de barras que mostra os tempos de vectorização dos prédios. 186

Lista de Tabelas

2.1	A determinação do ângulo θ (orientação) é ambíguo, assim é necessário verificar $\mu_{2,0} - \mu_{0,2} \geq 0$ e $\mu_{1,1} \geq 0$ para encontrar o valor θ	19
8.1	Esta tabela apresenta a amostra das secções cadastrais utilizadas, identificando o código <i>DICOFRE</i> (o DICOFRE é a concatenação dos códigos do distrito,concelho e freguesia) nome da secção, largura, altura da imagem e período de execução cadastral (o período está identificado na figura 8.2, sendo o valor 1 correspondente a 1944-1954, o valor 2 a 1955-1966, o valor 3 a 1967-1979 e o valor 4 a 1980-1995).	152
8.2	Esta tabela mostra a quantidade das secções cadastrais em função dos períodos de execução, calculando a sua percentagem relativamente ao total, de modo a encontrar uma sub-amostra que se aproxime da representação da população.155	
8.3	Esta tabela identifica as nomenclaturas das secções presentes em cada estrato da amostra.	155
8.4	Tabela de dupla entrada, que apresenta as médias das taxas de detecção. . .	179
8.5	Tabela de dupla entrada, que apresenta as médias dos falsos alarmes na detecção.	181
8.6	Tabela que compara o tempo de execução da vectorização.	186
A.1	Esta tabela apresenta as taxas de detecção das cruze. A taxa de detecção média é de 88.75% e a taxa de má detecções médias é de 6,49%.	202
A.2	Esta tabela apresenta a taxa de detecção dos círculos. A taxa de detecção média é de 99.83% e a taxa de má detecção média é de 1,60%.	203
A.3	Esta tabela apresenta a taxa de detecção dos arcos. A taxa de detecção média é de 82.00% e a taxa de má detecção média é de 8,14%. (*) Nos casos onde não existem arcos ou a taxa de má detecção é superior ou igual a 100% a taxa de detecção não é contabilizada, devido às razões apresentadas anteriormente.	204
A.4	Esta tabela apresenta a taxa de detecção dos tracejados. A taxa de detecção média é de 75.40% e a taxa de má detecção média dos tracejados é de 0,32%.205	

A.5	Taxa de detecção dos marcos. A taxa de detecção média é de 85,57% e a taxa de má detecção média é de 11,57%. (*) Os valores que apresentam uma elevada taxa de más detecções ou que a secção não é aplicável nesta análise, identificam as características excepcionais dos mapas e não foram considerados nesta análise.	206
A.6	Taxa de detecção dos caracteres. A taxa de detecção média dos caracteres é de 91,60% e a percentagem média de utilização de caracteres de treino é de 40,87%.	207
A.7	Esta tabela apresenta o resumo da taxa de detecção do contorno dos prédios, nas secções onde o método é aplicável.	208
A.8	Tabela que apresenta o resumo das taxas de reconhecimento de cada módulo individualmente.	209

Capítulo 1

Introdução

O tema do presente trabalho enquadra-se na área da análise de imagem de documentos (Document Image Analysis) e estando orientado para a análise de mapas cadastrais Portugueses. O domínio de análise de documentos, tal como se conhece na actualidade, apareceu apenas nos anos 60, sendo que os sistemas de reconhecimento de caracteres (OCR ¹) os primeiros sistemas realizados neste campo. Esta área de investigação apresenta muitos métodos, a maioria deles bem conhecidos, e são principalmente utilizados para resolver problemas genéricos, contudo o conhecimento específico do domínio de análise é essencial para a interpretação de um documento, uma vez que depois da segmentação de uma imagem são necessários métodos cada vez mais específicos.

Os algoritmos de análise de documentos têm sido aplicados a uma enorme variedade de documentos, incluindo desenhos de engenharia, mapas, pautas de música, desenhos esquemáticos e gráficos organizacionais. Os mapas cadastrais Portugueses são documentos bastante ricos em problemas específicos e de interesse nesta área de estudo.

¹Optical Character Recognition

1.1 Motivação

Em Portugal a informação cadastral ainda não foi completamente digitalizada e existem cerca 24 mil mapas cadastrais que aguardam a conversão para o formato vectorial, de modo a serem posteriormente integrados num sistema de informação. As folhas cadastrais foram desenhadas desde 1920 e até 1994, mas no entanto, ano após ano estão a ser degradadas e deformadas pela erosão, resultando numa constante perda de informação. Todas estas limitações evidenciam a importância da informatização dos mapas cadastrais Portugueses.

Felizmente, a digitalização manual já começou, mas como essa tarefa é altamente repetitiva e cansativa, um operador necessita de bastante tempo para vectorizar um mapa cadastral.² Uma vez que o processo é demasiado lento, principalmente devido à enorme quantidade de mapas e ao seu enorme tamanho³ é necessário reformular esta solução, senão a digitalização poderá levar mais de 10 anos a realizar-se.

Contudo, todas essas tarefas repetitivas podem ser realizadas por um computador, permitindo automatizar o processo consideravelmente. O objectivo do sistema descrito nesta tese é precisamente extrair a maior quantidade de informação cadastral dos mapas, detectando e validando toda a informação. Uma vez que os mapas não são uniformes, a informação não é completamente detectada e continua a ser necessário a intervenção humana.

²Em média um operador demora um dia e meio a vectorizar uma folha cadastral completa, segundo fontes do Instituto Geográfico Português.

³Um mapa cadastral apresenta o tamanho de uma folha A0.

1.2 Objectivos desta Tese

O objectivo desta tese é extrair objectos georeferenciados a partir dos mapas cadastrais Portugueses de uma forma automática. Um modo de atingir este objectivo é desenvolver técnicas que simplifiquem o reconhecimento de todos os elementos cadastrais. O trabalho contido nesta tese demonstra que é possível realizar este reconhecimento de uma forma simples e eficiente. Com este estudo, alguns problemas do reconhecimento de imagem foram revisitados e considerando a experiência adquirida, foram dadas algumas contribuições na área de estudo.

1.3 Organização da tese

Esta dissertação está organizada como se segue.

No capítulo 1 é realizada uma introdução, descrevendo o tema escolhido, a motivação desta escolha, os objectivos, a organização da tese e a importância nacional do trabalho.

O capítulo 2 apresenta uma visão geral sobre a área de análise de documentos.

No capítulo 3 é enquadrado o tema deste trabalho e são analisados os trabalhos já desenvolvidos nesta área.

No capítulo 4 são estudados os algoritmos de extracção de *features* nas secções cadastrais Portugueses. As *features* extraídas das imagens cadastrais são: as cruces georeferenciadas, os círculos, os arcos, os tracejados e os marcos do terreno.

No capítulo 5 são estudados os algoritmos para o reconhecimento de caracteres, utilizando métodos de *Template-Matching*, *Zoning* e *K-NN*. Os caracteres reconhecidos isoladamente são agrupados e validados considerando a sua semântica.

No capítulo 6 são analisados os métodos de extração do contorno utilizando métodos simples e robustos. Os contornos são extraídos utilizando um método composto que utiliza os seguintes métodos: Block-Fill, Lohmann, Seed-Segment, Line-Following e a vectorização Rosin-West.

No capítulo 7 é descrita a integração dos módulos de análise e são descritas as principais dificuldades do processo de integração do sistema. Seguidamente é descrita a implementação do interface e o modo de exportação da informação.

No capítulo 8 são apresentados os resultados e as taxas de reconhecimento do sistema. Finalmente, no capítulo 9 são apresentadas as conclusões desta tese.

1.4 Importância Nacional deste Trabalho

Este trabalho apresenta importância a nível nacional, uma vez que pretende acelerar consideravelmente a resolução de um problema da administração pública, que vem constantemente a ser adiado. A solução presente neste trabalho pretende contribuir para o aceleramento da informatização do cadastro em Portugal, bem como oferecer soluções automáticas para a execução de tarefas extremamente repetitivas e morosas.

1.4.1 Conceito de Cadastro e sua Importância

Cadastro Predial é um registo ordenado contendo informações sobre os Prédios existentes numa determinada jurisdição, informações essas capazes de proceder à descrição da localização, forma e composição do Prédio (componente geométrica) e à sua situação tributária (componente fiscal) e / ou à identificação dos direitos, ónus e encargos que sobre

ele recaem (componente legal).

O Cadastro constitui-se como uma ferramenta indispensável para as políticas de ordenamento do território, ambiente, económicas (em particular a agrícola e a florestal), fiscal e de obras públicas. Deste modo, o conhecimento dos limites e da titularidade da propriedade é imprescindível para todas as actividades de planeamento, gestão ou decisão sobre a ocupação e uso do território, para a regulação da repartição das mais-valias fundiárias e para a gestão, controlo e desenvolvimento dos recursos naturais e obras públicas.

1.4.2 Cadastro em Portugal

Em Portugal, o maior esforço de execução cadastral ocorreu entre as décadas 30 e 90 do Século XX, tendo sido efectuado o Cadastro Geométrico da Propriedade Rústica (CGPR) em 126 Concelhos, localizados maioritariamente na região sul do continente, correspondentes a cerca de 55% da área total do País (ver figura 1.1). Esta informação cadastral encontra-se representada em secções cadastrais às escalas de 1:500, 1:1.000, 1:2.000, 1:2.500 e 1:5.000.

Uma secção cadastral é uma planta topográfico-cadastral que representa, a uma determinada escala, uma projecção do território abrangendo normalmente um ou mais prédios rústicos completos. Este conjunto de prédios está desenhado num suporte inextensível, de área útil 80 cm x 60 cm e segundo normas e especificações determinadas (ver figura 1.2).

Portugal é um dos países da União Europeia onde a situação da execução cadastral apresenta um significativo atraso, quer em termos da cobertura territorial, quer em termos do suporte em que se encontra a informação existente. Deste modo, antes de actualizado, grande parte deste cadastro necessita ainda de ser convertido do actual formato analógico

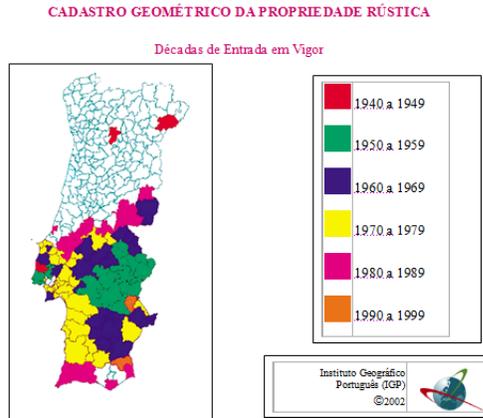


Figura 1.1: Esta imagem apresenta um mapa administrativo de Portugal Continental com as idades de execução das secções cadastrais.

para o formato digital.

Esta conversão, consiste na digitalização da informação contida nas secções cadastrais e na digitação da informação alfanumérica, contida nas fichas de prédio.

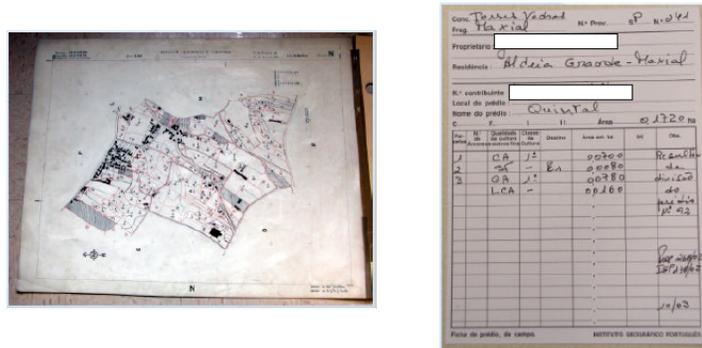


Figura 1.2: Esta figura mostra no lado esquerdo a informação geográfica patente numa secção cadastral e no lado direito a informação alfanumérica numa ficha de prédio (imagens extraídas da apresentação *A Informatização do CGPR*, na conferência internacional, GIS Planet 2005, realizada no Estoril, de autoria da Eng.^a Vanda Machado, in site www.igeo.pt).

1.4.3 Vantagens da Informatização do Cadastro

Informatizar os elementos gráficos e alfanuméricos do CGPR, constantes das cerca de 24 mil secções cadastrais, que cobrem cerca de 55% do País e as fichas de prédio que representam os cerca de 2 milhões de prédios existentes, permite:

- Produzir cópias de segurança da informação existente e evitar a sua deterioração aquando da constante manipulação das secções por consequência da alteração de prédios.
- Facilitar o seu armazenamento, a actualização da informação e manuseamento.
- Integrar os dados num sistema de gestão da informação cadastral potenciando as capacidades das entidades que trabalhem directamente com cadastro geométrico em vigor, tais como: Instituto Geográfico Português, Delegações Regionais do IGP, Municípios, Associações de Municípios, Direcção-Geral e Delegações Regionais de Agricultura e Desenvolvimento Rural e outros Organismos do Ministério da Agricultura e do Desenvolvimento Rural.
- Aproximar a informação existente à estrutura do cadastro predial, que posteriormente deverá ser executado segundo a Resolução de Conselho de Ministro N^o 45/2006 e o Decreto-Lei N^o 224/2007.

Nestes diplomas é criado o Sistema Nacional de Exploração de Informação Cadastral, designado SiNErGIC, que é um projecto concebido com o objectivo de proporcionar ao País uma gestão integrada e eficiente do Cadastro existente em conjugação com aquele que, no prazo de uma dezena de anos, será obtido para concluir a cobertura nacional.

1.4.4 Importância do Trabalho para o Cadastro Nacional

Sendo o resultado do presente trabalho, um software de vectorização capaz de reconhecer informação contida nos mapas cadastrais portugueses, existe assim a possibilidade de automatização de uma tarefa mecânica altamente morosa, traduzindo-se na eficiência e eficácia de todo o processo de informatização do Cadastro.

Actualmente o processo é realizado de forma manual, após a rasterização é incumbido a um operador a tarefa de vectorizar um mapa, ou seja, desenhar por cima da imagem, os contornos dos prédios, bem como os limites das parcelas e codificar posteriormente.

Pretende-se assim com este trabalho contribuir para o desenvolvimento sustentado do Cadastro em Portugal, nomeadamente para as regiões em regime de Cadastro Geométrico da Propriedade Rústica.

1.5 Contribuições

As principais contribuições desta tese são:

- Desenvolvimento de um sistema protótipo para o reconhecimento automático dos mapas cadastrais Portugueses;
- Solução para o problema da extracção de contornos fechados, para contornos com grandes discontinuidades, elementos conexos e existindo linhas cruzadas no contorno;
- Descrição de métodos simples e eficientes que resolvem os problemas do processo de reconhecimento das secções cadastrais portuguesas.

Capítulo 2

Visão Geral sobre a Análise de Imagem em Documentos

2.1 Introdução

Desde sempre, o Homem tentou criar novas ferramentas de modo a facilitar a sua vida, mas após o aparecimento do computador tentou ainda ir mais além, procurando desenvolver máquinas com capacidades semelhantes a si mesmo. A percepção através dos cinco sentidos foi desenvolvida em robôs usando sensores electrónicos e actualmente as máquinas podem ouvir, cheirar, falar, apalpar e ver. De qualquer forma, tão ou mais importante do que adquirir dados, é compreender e processar essa informação. Assim, desde as últimas décadas, inúmeros investigadores trabalham continuamente no desenvolvimento do processo de interpretação dos dados pela máquina.

A visão computacional revelou ser um dos campos com mais e maiores desafios na ciência computacional, uma vez que a percepção da máquina é distinta da do Homem. En-

quanto o Homem primeiramente analisa a imagem dum ponto de vista global, partindo do geral para o particular, a máquina, sem o mesmo entendimento, inicia uma análise local e só depois parte para o todo. Como a percepção computacional é uma tarefa difícil de sistematizar, a comunidade científica concentrou-se nalguns problemas específicos, separando a visão computacional em várias áreas, nomeadamente o processamento de imagem digital, a análise de documentos e a visão computacional.

Este capítulo apresenta uma visão geral sobre a evolução da análise de imagem em documentos. Primeiramente é feita uma revisão histórica sobre a análise de documentos de modo a compreender a origem e o estado actual desta área. Depois são apresentados os métodos que modelam a degradação de documentos e são revistos os fundamentos básicos do processamento de imagem binária. Finalmente são analisados alguns problemas do reconhecimento de imagem e são descritas soluções existentes na literatura para a resolução destes desafios.

2.2 Análise de Imagem em Documentos

A Análise de Imagem em Documentos (AID¹) é um campo do processamento de imagem digital e tem como objectivo converter imagens de documentos em formas simbólicas para posterior modificação, armazenamento, processamento, reutilização e transmissão. Ou tal como Nagy [86] sugere, a AID é a teoria e a prática de recuperar a estrutura simbólica de uma imagem digitalizada do formato de papel ou produzida por um computador.

O mesmo autor [86] faz uma breve descrição histórica do aparecimento de alguns sistemas de análise de documentos, desde os primeiros avanços até ao aparecimento desta

¹Na literatura esta área é conhecida como *Document Image Analysis*.

área do conhecimento, tal como hoje a conhecemos. Este campo surgiu ainda antes dos computadores conseguirem representar informação no tradicional formato de papel. Já no século XIX, existiam patentes relativas a sistemas de reconhecimento de caracteres ópticos (OCR²), estes sistemas permitiam ajudar pessoas cegas na leitura e na escrita de telégrafos. Em 1916, foram apresentados alguns destes modelos de trabalho. Os sistemas de OCR especializados em dígitos impressos (*fontes OCR*) foram primeiramente utilizados por empresas nos anos 50 e o primeiro leitor de endereços postal, tal como a primeira máquina para leitura de caracteres impressos em cartões de segurança social, foram instalados em 1965. Mas este campo, tal como hoje o conhecemos, apenas apareceu por volta dos anos 60.

Desde os anos 80, o preço da tecnologia desceu num factor de dez e nos anos 90, a capacidade de armazenamento aumentou milhares de vezes, assim o interesse em converter documentos para a forma digital aumentou rapidamente. Neste novo século, a transformação de imagens para o formato digital é um negócio que envolve milhões de euros, por exemplo algumas colecções de livros antigos apenas existentes nalgumas bibliotecas estão a ser digitalizados para consulta e pesquisa a partir da Internet³.

2.3 Degradação de Documentos

Os processos de duplicação de documentos, através da impressão, da fotocópia e do *scanning* degradam a qualidade de imagem dos documentos duplicados. A precisão dos algoritmos de um sistema de reconhecimento de documentos falha dramaticamente quando

²Optical Character Recognition

³Por exemplo, a empresa Google disponibiliza o serviço de visualização de livros online através do website <http://books.google.com>

a qualidade da imagem se degrada mesmo que ligeiramente, parecendo negligenciável para a vista humana. Na última década, alguns investigadores [56, 17] intensificaram o estudo numa forma explícita, quantitativa e parametrizada de modelos que introduzem defeitos na imagem, tais como os que ocorrem durante os processos de impressão e scanning.

A degradação é entendida como qualquer alteração mínima que diminua a representação ideal da imagem de um documento real. Existem muitas razões que justificam o estudo dos modelos de degradação de documentos, mas este estudo é fundamentalmente importante para calcular a performance dos algoritmos de reconhecimento. Por outro lado, os modelos de degradação permitem avaliar os algoritmos para níveis contínuos de degradação da imagem, desde os valores mais baixos até aos mais elevados.

2.3.1 Modelos de Degradação de Documentos

Baird [7] descreveu vários modelos propostos para o estudo da degradação de imagem, alguns motivados pelas leis físicas da formação de imagem e outros pelas superfícies estatísticas da distribuição da imagem. Esses modelos geram imagens sintéticas que permitem investigar propriedades fundamentais do reconhecimento de imagem. No entanto a validação desses modelos é ainda um problema em aberto.

Quase todos os modelos de degradação de imagem foram desenvolvidos no contexto do OCR. Os modelos de degradação propostos por Li et al. [72] consideram que um modelo é validado, se os erros do OCR induzidos pelo modelo são indistinguíveis dos erros encontrados na digitalização real dos documentos. O principal problema é que o seu método valida os modelos de OCR e não apenas o seu método, pois se o OCR for robusto, a taxa de reconhecimento não representa a validação do modelo de degradação.

O primeiro modelo com validação foi proposto por Kanungo [56, 58]. O teste de validação é um método estatístico realizado para estudar a hipótese de que dois conjuntos de imagens (digamos um real e outro sintetizado) foram desenhados da mesma distribuição normal. No contexto da parametrização dos modelos de degradação, a hipótese pode ser analisada como uma função dos parâmetros do modelo.

Foram propostos muitos métodos estatísticos para validar os modelos de degradação [59, 60]. Existem dois cenários distintos mas relacionados. O primeiro é o modelo de calibração, onde é assumido que são conhecidas a imagem ideal de entrada e a imagem de saída, já com o processo de degradação. O segundo é um problema de estimação geral [106, 139], onde apenas a imagem de saída do processo de degradação é conhecida.

2.3.2 Modelo de Kanungo et al.

O método de Kanungo [58] é aceite como suficientemente sensível à diversidade dos fenómenos físicos da degradação de imagem, para servir como modelo base na estimação da qualidade de uma imagem [7]. Este modelo é usado para introduzir distorções locais em imagens sintéticas. O modelo de degradação local dos documentos realiza: 1) inversão de pixels (do *background* para o *foreground* e vice-versa) que ocorrem independentemente em cada pixel devido às flutuações da intensidade de luz, sensibilidade dos sensores e ao nível de *threshold* e 2) *blur*, que ocorre devido à função do *point-spread* do sistema óptico do *scanner*.

O modelo de degradação tem seis parâmetros: $\Theta = (\eta, \alpha_o, \alpha, \beta_o, \beta, k)^t$. Esses parâmetros são usados para degradar uma imagem, realizando os seguintes passos:

1. Calcular a distância d de cada pixel desde o limite do carácter.

- Alterar cada pixel do *foreground* com uma probabilidade

$$p(0|1, d, \alpha_0, \alpha) = \alpha_0 e^{-\alpha d^2} + \eta. \quad (2.1)$$

- Alterar cada pixel do *background* com uma probabilidade

$$p(0|1, d, \beta_0, \beta) = \beta_0 e^{-\beta d^2} + \eta. \quad (2.2)$$

- Realizar a operação morfológica de *closing* usando um elemento com estrutura de disco e diâmetro k .

Este método considera a probabilidade da troca dos pixels do *background* como uma função exponencial da distância entre um pixel e o pixel mais próximo no limite (ver figura 2.1). O parâmetro α_0 é o valor inicial da exponencial, enquanto que o parâmetro α controla a velocidade de queda da exponencial. A distância entre os pixels do *foreground* e do *background* é calculada usando a transformada da distância standard [12]. As probabilidades de troca dos pixels do *foreground* são similarmente controladas por β_0 e β . O parâmetro η é uma constante de probabilidade da troca de todos os pixels. Finalmente, o parâmetro k , é o tamanho do disco usado na operação morfológica de *closing*, considerado como sendo a correlação introduzida pela função de *point-spread* do sistema óptico.

2.4 Processamento de Imagem Binária

O processamento de imagem é uma área bem estudada na literatura. As suas técnicas normalmente permitem transformar uma imagem noutra imagem e é deixado ao utilizador a tarefa de extrair informação. Este campo, inclui tópicos tais como realce de imagem, restauração, segmentação, filtros, extracção de *features*⁴, compressão, operadores mor-

⁴Termo técnico para representar características.

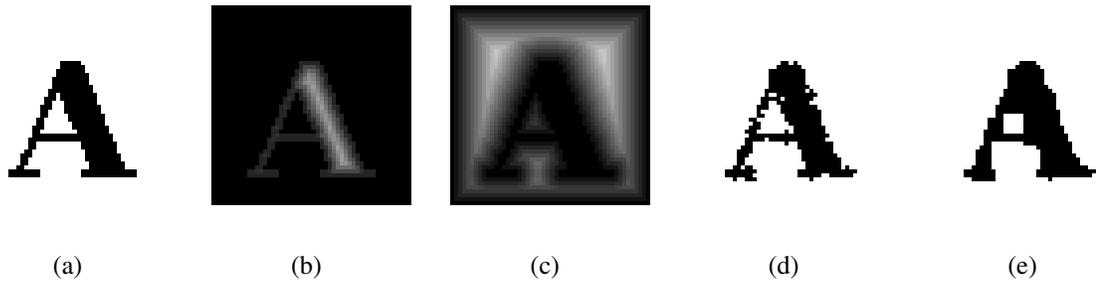


Figura 2.1: Modelo de distorção morfológica local: (a) Representação ideal; (b) Transformada da distância do *foreground* (a); (c) Transformada da distância do *background* (a); (d) Imagem com o resultado da perturbação usando a distribuição exponencial de probabilidade $P(0|d, \beta, f) = P(1|d, \alpha, b) = e^{-\alpha d^2}$ e $\alpha = \beta = 2$, $\alpha_0 = \beta_0 = 1$, $\eta = 0$; (e) Resultado do *closing* em (d) por um elemento binário estruturante de tamanho 2×2 .

fológicos e descrição de imagem. Os algoritmos de processamento de imagem são úteis principalmente na fase inicial dos sistemas de visão computacional, para evidenciar alguma informação particular ou para remover o ruído.

Uma imagem contém valores de intensidade contínuos que são quantificados para obter uma imagem digital. Para interpretar uma imagem, devem ser analisadas as variações dos valores de intensidade.

2.4.1 Binarização

A *binarização* (*thresholding*) é a primeira etapa na maioria dos sistemas de análise de documentos e refere-se à conversão da imagem cinza numa imagem binária. A binarização é uma etapa fundamental no processamento de imagem dos documentos, uma vez que uma boa binarização é a base para uma boa segmentação, necessária ao reconhecimento de objectos. Numa visão simplista, a binarização distingue entre os objectos do *background* e os do *foreground*.

Na literatura existem principalmente dois tipos de binarização, nomeadamente a binarização

global e a binarização *local adaptativa*. Os métodos de binarização global calculam um valor simples de *threshold* para uma imagem inteira. Os pixels que têm um valor cinza mais escuro que os valores de *threshold* são colocados a preto (*foreground*), caso contrário ficam com a cor branca (*background*). Por outro lado, os métodos de binarização local adaptativa, calculam um *threshold* para cada pixel com base na informação contida nos pixels vizinhos. Se um pixel (x,y) que está na imagem de entrada tem um valor cinza maior do que o *threshold* na superfície do ponto (x,y) , então o pixel (x,y) na saída terá a *label* do *background*, caso contrário será colocado como *foreground* (preto ou cinza).

Na literatura, Trier et al. [131] propuseram uma metodologia para avaliar sobre a binarização e também estudaram onze métodos locais adaptativos de binarização para imagens *gray-scale* com baixo contraste, intensidade do *background* variável e ruído [131]. Eles concluíram que o método de Niblack [88] com a adição do passo de pós-processamento do método de Yanowitz e Bruckstein [140] melhora a performance ao máximo, para além de ser o método de binarização mais rápido.

Mais recentemente, Gatos et al. [41] propuseram um novo esquema para a binarização de imagens digitais, usando cinco passos distintos. A sua metodologia tem algumas optimizações usando filtros e um pós-processamento final. Contudo, estes autores consideram que a melhor abordagem para a binarização adaptativa são os métodos de Niblack [88] e de Sauvola [102].

O algoritmo de Niblack [88] calcula o *threshold* de um pixel deslocando uma janela rectangular pela imagem. O *threshold* T para o pixel central da janela é calculado usando a média (m) e a variância (s) dos valores cinza da janela.

$$T = m + ks \tag{2.3}$$

onde k é uma constante igual a -0.2 . O valor de k foi calculado e corresponde à quantidade do limite do objecto impresso sendo considerado como parte do dado objecto. Este método pode distinguir efectivamente o objecto do *background* nas áreas próximas dos objectos. Contudo, o ruído que está presente no *background* continua dominante na imagem binária final. Consequentemente, se os objectos estão separados numa imagem, será deixado muito ruído no *background*.

O método de Sauvola [102] resolve este problema adicionando uma hipótese sobre os valores cinza do texto e dos pixels do *background*, do qual resulta a seguinte formula para o *threshold*:

$$T = m + (1 - k(1 - s/R)) \quad (2.4)$$

onde R é o intervalo dinâmico do desvio padrão igual a 128 e k igual a 0.5. Este método dá melhores resultados ao nível da binarização para imagens de documentos.

2.4.2 Propriedades Geométricas

Os momentos geométricos [65] descrevem características geométricas dos objectos. Este método clássico é uma das primeiras abordagens ao problema do reconhecimento de padrões [4].

A definição geral dos momentos é dada por

$$m_{p,q} = \sum_{i=1}^n \sum_{j=1}^m i^p j^q B[i, j], \quad (2.5)$$

onde $B[i, j]$ é uma imagem binária.

Os momentos centrais são expressos pela seguinte fórmula

$$\mu_{p,q} = \sum_{i=1}^n \sum_{j=1}^m (i - \bar{i})^p (j - \bar{j})^q B[i, j]. \quad (2.6)$$

Das equações 2.5 e 2.6 resulta que $\mu_{0,0} = m_{0,0}$ e $\mu_{1,0} = \mu_{0,1} = 0$.

Os momentos centrais de primeira ordem ou maior podem ser derivados directamente dos momentos espaciais por

$$\mu_{p,q} = \frac{m_{p,q}}{m_{0,0}} - \left(\frac{m_{1,0}}{m_{0,0}} \right)^p \times \left(\frac{m_{0,1}}{m_{0,0}} \right)^q. \quad (2.7)$$

A desvantagem dos momentos espaciais e dos momentos centrais é a sua dependência relativamente ao tamanho do objecto. Dividindo os momentos centrais $\mu_{p,q}$ com potência de A , obtemos os *momentos centrais normalizados* $\nu_{p,q}$

$$\nu_{p,q} = \frac{\mu_{p,q}}{m_{0,0}^{\frac{p+q}{2}+1}}, \quad (2.8)$$

invariante do tamanho do objecto.

Características dos objectos baseado em momentos

A área A de um objecto é dada pelo momento de ordem zero ($(p, q) = (0, 0)$)

$$A = m_{0,0} = \sum_{i=1}^n \sum_{j=1}^m B[i, j]. \quad (2.9)$$

Os momentos de primeira ordem contêm informação sobre o centro de gravidade de um objecto

$$\bar{i} = \frac{m_{1,0}}{A} = \frac{m_{1,0}}{m_{0,0}} \quad (2.10)$$

$$\bar{j} = \frac{m_{0,1}}{A} = \frac{m_{0,1}}{m_{0,0}} \quad (2.11)$$

A orientação de um objecto é definida como o ângulo entre o eixo do X e o eixo sobre o qual o objecto pode ser rodado com inércia mínima.

$$\theta = \frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} \quad (2.12)$$

Existe uma ambiguidade na determinação do ângulo θ , mas pode ser resolvido considerando a tabela 2.4.2.

$\mu_{2,0} - \mu_{0,2}$	$\mu_{1,1}$	θ	
Zero	Zero	0°	
Zero	Positivo	$+45^\circ$	
Positivo	Zero	0°	
Negativo	Zero	-90°	
Positivo	Positivo	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}}$	$0^\circ < \theta < 45^\circ$
Positivo	Negativo	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} + 90^\circ$	$-45^\circ < \theta < 0^\circ$
Negativo	Positivo	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} + 90^\circ$	$45^\circ < \theta < 90^\circ$
Negativo	Negativo	$\frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} - 90^\circ$	$-90^\circ < \theta < -45^\circ$

Tabela 2.1: A determinação do ângulo θ (orientação) é ambíguo, assim é necessário verificar $\mu_{2,0} - \mu_{0,2} \geq 0$ e $\mu_{1,1} \geq 0$ para encontrar o valor θ .

A circularidade κ pode ser facilmente calculada dividindo o quadrado do perímetro p pela área A .

$$\kappa = \frac{p^2}{A} \quad (2.13)$$

A centricidade ϵ pode ser directamente calculada a partir dos momentos centrais de segunda ordem por

$$\varepsilon = \frac{(\mu_{2,0} - \mu_{0,2})^2 - 4\mu_{1,1}^2}{(\mu_{2,0} + \mu_{0,2})^2} \quad (2.14)$$

A centricidade ε pode variar num intervalo de valores entre 0 (um objecto perfeitamente redondo) e 1 (um objecto com forma de linha).

2.4.3 Projecções

A projecção de uma imagem binária sobre uma linha pode ser obtida dividindo a linha em zonas de memória e encontrando o número de pixeis de cor preta que estão nas linhas perpendiculares a cada zona. As projecções são representações compactas da imagem, dado que, muita informação útil pode estar contida na projecção. Contudo, as projecções não são únicas, pois várias imagens podem ter a mesma projecção. As projecções horizontais e verticais podem ser facilmente obtidas encontrando o número de pixeis 1 para cada zona na direcção vertical e horizontal, respectivamente como mostra a figura 2.2.



Figura 2.2: A imagem binária da palavra CA com a sua projecção horizontal e vertical.

A projecção $H[i]$ ao longo das linhas e a projecção $V[j]$ ao longo das colunas de uma imagem binária são dadas por

$$H[i] = \sum_{j=1}^m B[i, j] \quad (2.15)$$

$$V[j] = \sum_{i=1}^n B[i, j] \quad (2.16)$$

Nalgumas aplicações, as projecções podem ser utilizadas como características de reconhecimento dos objectos. As projecções oferecem uma representação compacta e permitem realizar algoritmos rápidos.

2.4.4 Algoritmos Binários

Os algoritmos de processamento digital de imagem encontram-se muito bem estudados e na literatura existem variados livros [52, 90, 116, 92, 37] que constituem a referência base de todos os trabalhos. Nesta secção são revistos conceitos base e também alguns algoritmos clássicos do processamento digital de imagem binária, uma vez que estes constituem a base de qualquer algoritmo de reconhecimento de imagem.

A segmentação entre os pixels do objecto e os pixels do *background* é um problema difícil. Vamos assumir que podemos definir um objecto e colocar-lhe uma *label* usando um predicado, i.e., detectando todos os pontos de uma imagem pertencendo ao objecto. O problema é então agrupar todos os pontos de uma imagem, que têm a mesma *label* dos pontos do objecto, numa imagem do objecto. Vamos assumir que tais pontos estão espacialmente próximos. Esta noção de *proximidade espacial* requer uma definição mais precisa, tal que um algoritmo agrupe pontos espacialmente próximos para formar um componente. Para este fim, vamos introduzir algumas definições.

Definição 2.4.1 Um pixel numa imagem digital está espacialmente próximo de outros pixels. Numa imagem digital representada por uma grelha de quadrados, um pixel tem limites comuns com quatro pixels e partilha um canto com quatro pixels adicionais. Dizemos que dois pixels são vizinhos-4 se eles partilham um limite comum. Similarmente, dois pixels são vizinhos-8 se partilharem pelo menos um canto. Por exemplo, o pixel com a localização $[i, j]$ é vizinhos-4 de $[i + 1, j]$, $[i - 1, j]$, $[i, j + 1]$ e $[i, j - 1]$. Os vizinhos-8 do pixel incluem os vizinhos-4 mais os pontos $[i + 1, j + 1]$, $[i + 1, j - 1]$, $[i - 1, j + 1]$ e $[i - 1, j - 1]$. Um pixel é dito estar conexo-4 aos seus vizinhos-4 e conexo-8 aos seus vizinhos-8 (ver figura 2.3).

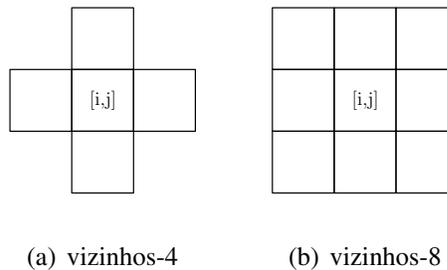


Figura 2.3: As vizinhanças-4 e -8 de uma imagem rectangular. O pixel $[i, j]$ está localizado no centro de cada figura..

Definição 2.4.2 O caminho desde o pixel $[i_0, j_0]$ até ao pixel $[i_n, j_n]$ é uma sequência de índices de pixels $[i_0, j_0], [i_1, j_1], [i_2, j_2], \dots, [i_n, j_n]$ tal que o pixel em $[i_k, j_k]$ é vizinho do pixel em $[i_{k+1}, j_{k+1}]$, para todo k com $0 \leq k \leq n - 1$. Se a relação de vizinhança utiliza uma conexão-4, então o caminho é um caminho-4; para uma conexão-8, o caminho é um caminho-8.

Definição 2.4.3 O conjunto de todos os pixels a 1 numa imagem é chamado de foreground e é representado por S .

Definição 2.4.4 Um pixel $p \in S$ é chamado conexo a $q \in S$ se existir um caminho de p até q usando apenas pixels pertencentes a S .

De notar que a conectividade é uma relação de equivalência. Para quaisquer três pixels p, q e r em S , temos as seguintes propriedades:

1. O pixel p está conexo a p (reflexibilidade).
2. Se p está conexo a q , então q está conexo a p (comutabilidade).
3. Se p está conexo a q e q está conexo a r , então p está conexo a r (transitividade).

Definição 2.4.5 Um conjunto de pixels no qual cada pixel está conectado a todos os outros pixels é chamado um componente conexo.

Definição 2.4.6 Ao conjunto de todos os componentes conexos de \bar{S} (o complemento de S) que tenham pontos sobre a borda de uma imagem é chamado de background. Todos os outros componentes de \bar{S} são chamados de buracos.

Definição 2.4.7 O limite de S é o conjunto de pixels de S que têm vizinhança-4 em \bar{S} . O limite é usualmente representado por S' .

Definição 2.4.8 O interior é o conjunto de pixels de S que não estão no seu limite. O interior de S é $(S - S')$.

Definição 2.4.9 A região T contém a região S (ou S está dentro de T), se qualquer caminho-4 de qualquer ponto de S ao limite do desenho intersecta T .

Labeling dos Componentes

Uma das operações mais comuns em processamento de imagem é encontrar os componentes conexos numa imagem. Os pontos dentro de um componente conexo formam uma região candidata para representar um objecto. O algoritmo de *labeling* dos componentes encontra todos os componentes conexos numa imagem e assinala uma única *label* para todos os pontos do mesmo componente.

Wu et al. [138] propuseram recentemente duas novas estratégias para aumentar a velocidade do algoritmo de *labeling* dos componentes conexos. A optimização está em reduzir o número de vizinhos examinados considerando componentes conexos-8 numa imagem com duas dimensões e usando *arrays* para guardar a informação equivalente entre as *labels*.

Filtro pelo Tamanho

É muito comum utilizar um *threshold* para objectos numa imagem binária. Na maioria dos casos, existem algumas regiões numa imagem que são devidas ao ruído. Normalmente essas regiões são pequenas. Em várias aplicações, é conhecido que os objectos de interesse têm um tamanho maior do que T_0 pixels. Em tais casos, podemos realizar um filtro em função do tamanho para remover o ruído depois do *labeling* dos componentes. Todos os componentes com tamanho menor do que T_0 são removidos, modificando os pixels correspondentes para 0. Este mecanismo de filtragem simplificado é extremamente eficiente na remoção de ruído.

Número de Euler

Em muitas aplicações, o *genus* ou *número de Euler* é usado como *feature* de um objecto. O *genus* é definido como o número de componentes menos o número de buracos. Assim,

$$E = C - B \quad (2.17)$$

onde E , C e B são respectivamente o número de Euler, o número de componentes conexos e o número de buracos. Isto disponibiliza uma característica topológica simples que é invariante à translação, rotação e escala.

Limite da Região

O limite de um componente conexo S , é o conjunto de pixels de S que são adjacentes a \bar{S} . Uma operação local simples pode ser utilizada para encontrar os pixels do limite. Na maioria das aplicações, é pretendido seguir os pixels do limite numa ordem particular. Uma abordagem comum é seguir todos os pixels de uma região na sequência dos ponteiros do relógio.

O algoritmo de seguimento do limite selecciona um pixel inicial $s \in S$ e segue o limite até voltar ao ponto inicial, assumindo que o limite não é um lado da imagem.

Área e Perímetro

Tal como já foi referido anteriormente, a *área* é o número de pixels em S . Se existem vários componentes S_1, S_2, \dots, S_n , então a área de cada componente é o número de pixels nesse componente. O número de pixels em cada componente pode ser obtido através da *labeling* dos componentes. Num caso geral, a área de cada um dos n componentes pode ser

obtida numa verificação da imagem.

O *perímetro* de um componente pode ser definido de várias formas diferentes. Algumas definições usuais são:

1. A soma dos comprimentos das linhas de separação entre os pixels de S e os pixels de \bar{S} . Uma linha de separação contém dois pares de pixels p e q tal que $p \in S$ e $q \in \bar{S}$.
2. O número de passos levados pelo algoritmo de seguimento do limite.
3. O número de pixels do limite de S .

O perímetro medido será diferente dependendo da definição aplicada. Em geral, o perímetro obtido usando a definição 1 é muito maior do que o perímetro medido usando as outras duas definições.

Compactividade

É bem conhecido que a *compactividade* de uma figura geométrica contínua é medida pela inequação isoperimétrica

$$\frac{P^2}{A} \geq 4\pi \quad (2.18)$$

onde P e A são o perímetro e a área, respectivamente. O círculo é a figura mais compacta (i.e. tem o menor valor de compactividade) considerando esta medida.

Outra forma de observar a compactividade é que uma região mais compacta contém uma grande quantidade de área para um dado perímetro. De notar que um quadrado é mais compacto que um rectângulo com o mesmo perímetro.

Medidas de Distância

Em muitas aplicações, é necessário encontrar a distância entre dois pixels ou dois componentes de uma imagem. Infelizmente, não existe um único método de definir distâncias em imagens digitais. As distâncias podem ser definidas de muitas maneiras diferentes. Para todos os pixels p , q e r , qualquer métrica de distância deve satisfazer todas as três seguintes propriedades:

1. $d(p, q) \geq 0$ e $d(p, q) = 0$ se $p = q$
2. $d(q, p) = d(p, q)$
3. $d(p, r) \leq d(p, q) + d(q, r)$

Muitas funções de distância têm sido usadas na geometria digital. Algumas das funções de distância mais comuns são:

Euclidiana

$$d_{Euclidean}([i_1, j_1], [i_2, j_2]) = \sqrt{(i_1 - i_2)^2 + (j_1 - j_2)^2} \quad (2.19)$$

City-block

$$d_{city} = |i_1 - i_2| + |j_1 - j_2| \quad (2.20)$$

Chessboard

$$d_{chess} = \max(|i_1 - i_2|, |j_1 - j_2|) \quad (2.21)$$

Transformada da Distância

A transformada da distância foi primeiramente introduzida por Pfaltz e Rosenfeld [99] e produz um mapa de distâncias a partir de uma imagem binária. Para cada pixel no interior

dos objectos de uma imagem binária, o pixel correspondente no mapa de distâncias tem o valor igual à distância mínima até ao *background*. De um modo similar, pode-se definir a distância para o *background*.

Algumas métricas diferentes têm sido definidas para o plano discreto, por exemplo as distâncias *city-block* e *chessboard*. Essas distâncias são substancialmente diferentes da distância Euclidiana. Uma vez que os custos computacionais da transformada da distância Euclidiana são relativamente elevados, muitos algoritmos de processamento rápido têm sido desenvolvidos para aproximarem-se dessa distância. A maior classe de tais algoritmos é baseada nas métricas de chamfer [12]. O termo *chamfer* refere-se à sequência de dois passos da transformada da distância desenvolvido por Pfaltz e Rosenfeld [99] e mais tarde melhorado e generalizado por Borgefors [12].

Em certas aplicações, tais como no reconhecimento de caracteres, a distância mínima é utilizada para calcular a distância entre o *background* e o pixel de um objecto. Assim, para uma dada região *foreground* S , devemos calcular a distância até à região *background* \bar{S} , para todos os pixels em S . O resultado desta transformação designa-se por transformada da distância. Para calcular a transformada da distância pode ser utilizado um algoritmo paralelo iterativo usando as seguintes equações

$$f^0[i, j] = f[i, j] \quad (2.22)$$

$$f^m[i, j] = f^0[i, j] + \min(f^{m-1}[u, v]) \quad (2.23)$$

onde m é o número de iteração para todos os pixels $[u, v]$, tal que $d([u, v], [i, j]) = 1$. De notar que este método apenas usa uma vizinhança-4 em $[i, j]$.

Eixo Médio

A transformada do eixo médio (TEM) é uma representação de imagem proposta por Blum [11]. Para uma imagem binária digital, o problema da TEM é definido como encontrar os círculos máximos dentro de um objecto, constituído apenas por pixels com valor 1.

O esqueleto e o eixo médio permitem representar de uma forma muito compacta os objectos, mesmo no caso de formas complexas e topológicas. Estas representações descrevem muito bem as formas, sendo uma ponte na representação dos objectos entre o baixo e o alto nível, pois permitem resumir, sintetizar e auxiliar a compreensão da forma do objecto e a sua topologia.

Dizemos que a distância $d([i, j], \bar{S})$ desde o pixel $[i, j]$ em S até \bar{S} é um máximo local se

$$d([i, j], \bar{S}) \geq d([u, v], \bar{S}) \quad (2.24)$$

para todos os pixels $[u, v]$ na vizinhança de $[i, j]$. Ao conjunto dos pixels em S com distâncias desde \bar{S} , que são localmente máximos são chamados de esqueleto, eixo simétrico ou eixo médio de S e é normalmente representado por S^* .

O conjunto original S pode ser reconstruído a partir de S^* e das distâncias de cada pixel de S^* desde \bar{S} . S^* é usado para representar a forma de uma região. Apagando os pixels de S^* cujas distâncias de \bar{S} são pequenas, podemos criar uma versão simplificada de S^* .

Thinning

O *thinning* é uma operação de processamento de imagem na qual as regiões binárias da imagem são reduzidas a linhas que aproximam as suas linhas centrais, também chama-

das de esqueleto ou linhas do núcleo. Esta operação é importante não apenas por reduzir a quantidade de dados armazenados, mas também porque facilita a extracção de padrões característicos. Embora a operação de *thinning* possa ser aplicada a imagens binárias contendo regiões com qualquer forma, é principalmente útil para formas alongadas e menos para convexas. O *thinning* é usado normalmente como uma etapa de pré-processamento nas aplicações de análise de documentos para representar linhas nos desenhos ou partes de letras numa imagem com texto.

Vários algoritmos têm sido propostos para realizar o *thinning* de padrões digitais. A maioria deles realiza uma transformação iterativa do padrão digital [141]. Em cada iteração, parte dos contornos são apagados até não ser possível eliminar mais pontos do objecto.

Expansão e Shrinking

O componente de uma imagem pode ser sistematicamente expandido ou contraído. Quando um componente é alterado de modo a que alguns pixels do *background* são convertidos para 1, esta-se perante uma operação designada de *expansão*. Se os pixels do objecto são sistematicamente apagados ou convertidos para 0, então está-se perante uma operação designada de *shrinking*. Deste modo, uma implementação simples de expansão ou *shrinking* pode ser:

Expansão: Troca um pixel de 0 para 1 se qualquer vizinho do pixel é 1.

Shrinking: Troca um pixel de 1 para 0 se qualquer vizinho do pixel é 0.

O *shrinking* também pode ser considerado como a expansão do *background*.

É interessante observar que as operações simples como a expansão e o *shrinking* podem ser utilizadas para realizar operações aparentemente complexas em imagens. A expansão

seguida de *shrinking* pode ser utilizada para preencher buracos indesejáveis, e o *shrinking* seguido da expansão pode ser usada para remover pixels de ruído.

2.4.5 Operadores Morfológicos

A morfologia matemática obtém o seu nome do estudo da forma. Esta abordagem é bastante utilizada na área da visão computacional, devido a ser natural e fácil pensar no processamento de imagens utilizando operações baseadas na forma dos elementos.

A *intersecção* de quaisquer duas imagens binárias A e B , escreve-se $A \cap B$, é uma imagem binária com valor igual a 1 em todos os pixels p onde as imagens A e B têm ambas o valor 1. Assim,

$$A \cap B = \{p | p \in A \text{ e } p \in B\}. \quad (2.25)$$

A *união* de A e B , escreve-se $A \cup B$, é uma imagem binária com valor igual a 1 em todos os pixels p onde A tem o valor 1 ou B tem o valor 1 (ou em ambas as imagens). Simbolicamente,

$$A \cup B = \{p | p \in A \text{ ou } p \in B\}. \quad (2.26)$$

Seja Ω uma imagem binária universal (apenas com valores iguais a 1) e A uma imagem binária. O *complemento* de A é uma imagem binária que altera os valores 1 e 0 na imagem A . Assim,

$$\bar{A} = \{p | p \in \Omega \text{ e } p \notin A\}. \quad (2.27)$$

O *vector soma* de dois pixels p e q com índices $[i, j]$ e $[k, l]$ é o pixel $p + q$ com índices $[i + k, j + l]$. O *vector diferença* $p - q$ é o pixel com índices $[i - k, j - l]$.

Se A é uma imagem binária e p é um pixel, então a *translação* de A por B é uma imagem dada por

$$A_p = \{a + p | a \in A\}. \quad (2.28)$$

Dilatação

A translação de uma imagem A pelo pixel p desloca a origem de A até p . Se $A_{b_1}, A_{b_2}, \dots, A_{b_n}$ são deslocados de uma imagem binária A por pixels 1 de uma imagem binária $B = \{b_1, b_2, \dots, b_n\}$, então a união das translações de A pelos pixels 1 de B é designada de *dilatação* de A por B e é dada por

$$A \oplus B = \cup_{b_i \in B} A_{b_i}. \quad (2.29)$$

A dilatação possui as propriedades da associatividade e da comutação. Assim, numa sequência de passos de dilatação, a ordem de realização das operações não é importante. Este facto permite partir uma forma complexa em formas mais simples que podem ser recombinadas como uma sequência de dilatações.

Erosão

O oposto da dilatação é a *erosão*. A erosão de uma imagem binária A por uma imagem binária B é 1 no pixel p se e apenas se todos os pixels 1 na translação de B até p são também 1 em A . A erosão é dada por

$$A \ominus B = \{p | B_p \subseteq A\}. \quad (2.30)$$

Algumas vezes a imagem binária B é uma forma regular que é usada como uma prova sobre a imagem A e é referida como um *elemento estruturante*. A erosão aplica-se em muitas situações, por exemplo a erosão da imagem usando os elementos estruturantes resulta numa imagem que dá todas as localizações onde o elemento estruturante está contido na imagem.

2.5 Reconhecimento de Imagem

Tradicionalmente, a análise de imagem em documentos tem sido por assim dizer uma actividade de *reverse engineering*, iniciando-se com o *scanning* de um documento em formato papel (a imagem do documento) e tendo como objectivo obter os caracteres originais, palavras, frases ou as estruturas dos documentos. Por outro lado, em documentos ricos em desenhos, o trabalho foi levado para a conversão de *raster* para *vector*, também chamado de vectorização.

O reconhecimento de gráficos é uma tarefa importante em aplicações da vida real. Este processo inclui ambos os reconhecimentos de baixo e de alto nível. O objectivo é reconhecer objectos gráficos, incluindo linhas rectas, arcos/círculos e curvas, os quais são parte fundamental em todo o processo. O último envolve uma análise estrutural pelo domínio de conhecimento. O reconhecimento gráfico presente neste trabalho é um reconhecimento de baixo nível. A investigação no reconhecimento de imagem foi conduzida nos últimos pares de anos, utilizando diversos métodos, tais como: a segmentação (texto/gráficos e linhas grossas/finas), vectorização, reconhecimento de gráficos característicos (tracejados, setas,

círculos e arcos) e reconhecimento de símbolos.

2.5.1 Segmentação Texto/Gráficos

Na análise da imagem de documentos, o processo de separação texto/gráficos visa segmentar o documento em duas camadas: uma camada contendo texto com caracteres e anotações e outra camada contendo objectos gráficos. Como as tarefas a serem efectuadas em cada uma das duas camadas são bastante diferentes, a maioria dos autores realiza esta separação muito cedo na cadeia de análise do documento, o que significa que o reconhecimento é realizado através das ferramentas de processamento de imagem, com um conhecimento limitado sobre a presença de objectos de alto nível.

Por exemplo, no caso especial das imagens com grelhas de preenchimento, o texto toca por vezes nos gráficos, mas estes últimos apresentam linhas horizontais e verticais, permitindo a possibilidade de procurar explicitamente por esse tipo de linhas, utilizando técnicas tais como a transformada de Hough [42].

Mas no caso geral de documentos gráficos, as linhas são mais complexas e todas essas abordagens não são muito eficientes. Neste caso, consideram-se três famílias básicas de métodos para a separação de texto e gráficos.

- Alguns autores realizam uma filtragem morfológica direccional para localizar todas as formas lineares e assim, separar essas de outras formas, as quais são consideradas como texto. Isto funciona particularmente bem para mapas simples [79], embora seja necessário observar a robustez da abordagem quando a complexidade dos desenhos aumenta.
- Da mesma maneira, outros autores procuram por linhas, usando a transformada da

distância [55], ou vectorizando a imagem do documento [32].

- Uma terceira abordagem muito usada é a análise baseada nos componentes conexos, os quais são filtrados através de um conjunto de regras para determinar a que camada pertencem. Um dos algoritmos melhor conhecido para realizar esta tarefa foi proposto por Fletcher e Kasturi [36]. Este método tem provado uma grande eficiência em documentos complexos, embora não consiga separar directamente o texto que toca nos gráficos.

Tombre et al. [126] apresentaram uma consolidação ao método proposto por Fletcher e Kasturi, com algumas melhorias, de modo a adapta-lo a documentos ricos em gráficos, nomeadamente discutindo sobre a escolha de *thresholds* e propondo um passo posterior de processamento para obter os componentes de texto que tocam nos gráficos.

2.5.2 Reconhecimento de Objectos Gráficos (Vectorização)

O reconhecimento de gráficos está centrado na análise de documentos gráficos intensivos, tais como desenhos técnicos, mapas e esquemas. Os métodos de reconhecimento de objectos gráficos existentes podem ser classificados em métodos de reconhecimento directo e métodos baseado na vectorização. A primeira classe reconhece os objectos gráficos directamente na imagem raster, enquanto a última classe converte primeiro a imagem raster numa representação vectorial de baixo nível, e então aplica alguns algoritmos de reconhecimento baseado em vectores para produzir o resultado final. Ambas as classes argumentam sobre as dificuldades da outra classe [124, 29, 115]; contudo, também sofrem das suas próprias dificuldades.

Realmente, ambas as classes têm feito um progresso significativo nos últimos anos; contudo o caminho mais promissor é talvez explorar um novo paradigma que integre as vantagens de ambos e evite as suas fraquezas [110].

Métodos baseados na vectorização

A vectorização, i.e. a conversão de raster para vectores, é certamente uma parte central do problema de reconhecimento de gráficos, uma vez que o seu propósito é converter uma imagem digitalizada para uma forma de vectores que posteriormente possam ser analisada. A maioria dos métodos de vectorização divide o processo de conversão em dois passos: vectorização pura e pós-processamento. Durante o passo de vectorização, extraem-se tantos segmentos de linha sem junções quanto possível, partindo da imagem raster, de seguida no passo de pós-processamento estendem-se e combinam-se os segmentos de linha em entidades gráficas exactas, tais como linhas rectas, arcos e curvas.

A principal razão para esta divisão em dois passos é que estes métodos de vectorização não conseguem extrair entidades gráficas num só passo, se existirem junções complicadas. Exemplos típicos da vectorização em dois passos são os métodos baseados na *skeletonization*, tais como algoritmos baseados no *thinning*, algoritmos baseados no contorno, algoritmos baseados na estrutura de grafos e métodos baseados no seguimento de *sparse pixels* (linhas finas e pouco densas).

Os métodos baseados no *thinning* [118, 22, 142, 87] normalmente aplicam um processo de erosão iterativa aos limites de modo a remover os pixels exteriores, até que exista apenas o esqueleto com largura de um pixel. Então ligam-se os pixels sobre o esqueleto à cadeia de pixels, usando um processo de seguimento de linha. Finalmente, converte-se a cadeia de

pixeis em vectores usando um procedimento de *line fitting*. Este método é conhecido como tendo como principais desvantagens a ineficiência de tempo e a existência de distorções nas junções.

O método baseado no contorno [44, 16, 9] extrai primeiro os contornos da imagem e então encontra o eixo médio entre os contornos. Este método por vezes tem o inconveniente de obter buracos que partem os vectores, devido à falta de pares de contornos nas junções.

O método baseado na estrutura dos grafos [91, 84] constrói um grafo de adjacência em linha depois de efectuar a codificação do comprimento dos segmentos na imagem. O procedimento de aproximação poligonal é aplicado aos pontos médios dos segmentos. Este método é susceptível ao ruído e pode causar distorções nas junções.

O método de seguimento de *sparse pixeis* [27, 29] pretende melhorar a eficiência e precisão pela verificação dos pixeis espalhados. O resultado é um esqueleto difuso e são necessárias operações de aproximações polinomiais para encontrar a melhor aproximação da linha. Este método pode apenas evitar parcialmente a quebra de linhas.

Como os vectores que resultam do primeiro passo não satisfazem as exigências, o segundo passo de pós-processamento, i.e. o refinamento dos vectores, deve considerar os problemas típicos, incluindo a reconstrução de intersecções, a localização dos extremos, a fragmentação de conexões e segmentação texto/gráficos, para converter o resultado dos vectores de baixo nível num gráfico final aceitável.

Métodos de Reconhecimento Directo

Os métodos de reconhecimento directo apenas funcionam sobre o nível pixel. O seguimento de pixeis, a transformada de Hough e a detecção de regiões são importantes

exemplos destes métodos.

Kovalevsky [67] propôs um algoritmo de seguimento de pixeis para linhas rectas e arcos. Para uma linha recta, o seguimento começa com dois pixeis vizinhos num padrão e constrói-se uma pequena linha protótipo com os primeiros pixeis. Então continua o seguimento desta linha até não encontrar mais pixeis com conexão-4 na pequena linha. Em cada passo do seguimento, é calculado o vector produto dos pixeis candidatos e os limites da pequena linha para verificar as condições internas de linha. Para detectar um arco, são encontrados pelo menos os dois primeiros segmentos de linha recta, os quais dão a primeira aproximação à curvatura do arco.

A transformada de Hough é também usada para detectar directamente na imagem raster linhas rectas, círculos e elipses. Uma vez que a transformação de todos os pixeis tem custos demasiado altos em tempo e memória, estes métodos seleccionam alguns pixeis como pontos característicos, contudo não se utilizam os pontos do esqueleto, para realizar a transformada, mas por exemplo *edge points* [119] ou pontos conexos [69].

Chiang et al. [18] propuseram um método baseado em regiões para reconhecer directamente linhas rectas a partir de imagens *raster*. Este método usa um círculo máximo inscrito (CMI) com padrões, para detectar as características de uma linha recta.

O método de vectorização de linha global [111, 115, 114], pode reconhecer directamente linhas rectas (sólidas e a tracejado), arcos e círculos completos a partir de imagens *raster*. A ideia principal deste método é primeiro detectar as características da forma pretendida, usando esta característica para guiar o seguimento de pixel de modo a obter o objecto gráfico completo. No final, os pixeis pretos que correspondem unicamente ao objecto gráfico detectado são apagados para simplificar os dados na imagens *raster*.

2.5.3 Reconhecimento de Gráficos Característicos

Existem algoritmos que reconhecem vários gráficos característicos e que por vezes são usados em desenhos de engenharia. Alguns desses algoritmos já se encontram no estado maduro. Por exemplo, existem boas ideias e caminhos validados para reconhecer linhas a tracejado [2, 28], setas [95], círculos e arcos [31, 100]. Outros métodos são ainda experimentais, tais como o reconhecimento de entradas (portas) e estão ainda em fase de maturação.

Contudo, alguns problemas continuam mais ou menos em aberto. Por exemplo, é difícil desenhar um algoritmo que possa reconhecer simultaneamente arcos e segmentos e conecta-los numa forma correcta, sem ambiguidades.

2.5.4 Reconhecimento de Símbolos

A detecção e classificação de símbolos é um dos aspectos mais relevantes da análise de muitos tipos de documentos. Cordella et al. [19] descrevem as várias técnicas propostas durante a última década. Os autores fazem referências aos métodos mais utilizados na representação, descrição e classificação de símbolos.

O conceito de símbolo é aqui entendido como um conjunto de sinais ou formas gráficas, distribuídos sobre um documento, que têm como função representar sinteticamente alguma informação. Eles são geralmente caracterizados pela atribuição de uma semântica e por um conjunto de regras. De facto, os símbolos são por vezes formas restritas. Eles são formados por regras e conhecendo-se essas regras pode simplificar-se o seu reconhecimento. A possibilidade de aparecerem conectados a outros componentes gráficos, sem isto ser obrigatório, torna difícil a generalização do seu procedimento de procura.

Normalmente é necessário um conhecimento prévio sobre a natureza do documento para localizar os símbolos candidatos ou regiões de interesse, as quais necessitam de técnicas específicas de reconhecimento.

Também foram apresentados um bom conjunto de artigos [125, 122, 57] dedicados parcialmente ou totalmente ao reconhecimento de símbolos.

Na generalidade, o processo de interpretação de uma imagem envolve três principais fases. Durante a primeira, a qual referimos como a fase de representação, o objectivo é colocar a imagem numa forma mais agradável, de modo a localizar as entidades de interesse para o subsequente processamento. Na segunda fase são descritas as entidades localizadas, enquanto na terceira é realizada a sua classificação.

Uma estratégia simples [19, 1], usada por muitos investigadores em várias aplicações, é a seguinte:

- Inicialmente o documento é pré-processado para obter uma representação mais conveniente. O tipo de pré-processamento depende parcialmente do tipo do documento.
- De seguida, os símbolos são procurados na representação, na base de características gerais partilhadas por todos os símbolos: este processo produz um número de possíveis candidatos.
- Seguidamente, cada símbolo candidato é descrito em termos de características com um maior detalhe para obter um vector característico ou uma descrição estrutural.
- Finalmente a descrição é preenchida na etapa de classificação, a qual atribui ao símbolo uma classe (i.e., reconhece a descrição de um símbolo específico tal como foi previamente aprendido ou como existe numa base de dados de *templates*).

Os principais objectivos da fase de representação é reduzir o ruído e a quantidade de dados, para descrever ou resumir os componentes da imagem e representa-los numa forma tal, que diferenças insignificantes das várias instâncias sejam esbatidas. Como referência a documentos contendo símbolos, existe uma grande variedade de técnicas nesta fase, por exemplo a binarização, *thinning*, aproximação poligonal, *run-length coding*, *curve fitting*, *labeling* dos componentes conexos, operações morfológicas matemáticas, transformadas de Hough, etc.

Um passo inicial muito comum no processamento de desenhos técnicos é o *labeling* dos componentes conexos, uma vez que existem muitas famílias de símbolos presentes nestas imagens e é necessário classificar cada símbolo. Os símbolos apresentam formas geométricas simples ou complexas e é necessário obter as características geométricas ou topológicas, de modo a classificar cada símbolo.

Em certos tipos de documentos, é comum que os símbolos toquem ou sobreponham os gráficos. É então conveniente seleccionar as características que caracterizam a representação de um símbolo a um nível tal que seja possível identificar univocamente cada símbolo das outras classes existentes.

Em princípio, a classificação implica o treino para aprendizagem do protótipo, uma base de conhecimento e a capacidade de usar essa informação junto à informação contextual.

2.5.5 Reconhecimento de Caracteres

O interesse nos sistemas de OCR para a digitalização automática de documentos de texto, cresceu exponencialmente nas últimas duas décadas. Este interesse não é surpreendente, considerando o volume de informação impressa e a necessidade dessa informação

estar disponível para os leitores no formato digital, e cada vez mais para os sistemas de processamento de informação.

Os métodos de reconhecimento de caracteres isolados, têm sido bem estudados na literatura. Os sistemas clássicos de OCR são constituídos por procedimentos de segmentação de caracteres, extracção de *features* e de classificação. Na maioria dos sistemas de OCR, a segmentação de caracteres conexos é um pré-processamento necessário ao reconhecimento, uma vez que cada caracter é isoladamente reconhecido. Os erros de leitura nos sistemas de OCR sucedem principalmente no processo de segmentação dos caracteres. O reconhecimento de caracteres isolados conduz ao problema do reconhecimento de padrões, contudo actualmente para estes casos já existem soluções maduras. As actuais direcções para melhorar a performance do OCR incluem a realização de melhores classificadores, a integração dos classificadores existentes para utilizar vários tipos de *features* e a extracção de *features* mais robustas.

Segmentação de Caracteres

A segmentação de caracteres é uma operação que decompõe uma imagem em sub-imagens de uma sequência de caracteres individuais. A segmentação dos caracteres tem sido uma área crítica no processamento do OCR. A elevada taxa de reconhecimento dos caracteres isolados em contraste com os resultados obtidos em palavras com caracteres conexos ilustra bem a importância do processo da segmentação.

Casey et al. [15] descreveram os métodos e estratégias da segmentação de caracteres, fazendo uma apreciação à variedade de técnicas desenvolvidas. Os autores resumem os avanços em quatro principais tipos de segmentação.

O primeiro tipo de segmentação também considerada como a abordagem clássica, consiste na utilização de métodos que dividem a imagem de entrada em sub-imagens, que são posteriormente classificadas. A operação de decompor a imagem em unidades classificáveis é chamada "dissecação". A maneira mais simples e fácil é analisar a projecção vertical de uma região *foreground*. Os métodos representativos [78, 73], detectam os vales na projecção para encontrar as zonas de corte.

O segundo tipo evita a dissecação e classifica ou explicitamente, pela classificação de janelas específicas, ou implicitamente pela classificação de sub-conjuntos de *features* espaciais obtidas da imagem como um todo. A abordagem de Casey e Nagy [14] reconhece a sub-imagem numa janela contraída, cuja largura diminui do lado esquerdo. Quando existe um *match* com o caracter, o bloco correspondente é cortado e o restante é recursivamente processado. Esta estratégia é mais eficiente do que a utilizada nos métodos baseados na projecção, uma vez que selecciona de forma dinâmica zonas de corte, porém não consegue tratar os tipos de corte não lineares.

A terceira estratégia, é uma solução híbrida [38] dos dois primeiros métodos aplicando a dissecação com regras de recombinação, i.e., definindo potenciais segmentos, classificando as sub-imagens e seleccionando um intervalo de possíveis segmentações.

Finalmente, a última abordagem evita a segmentação, reconhecendo as palavras inteiras como unidades descritivas. Os métodos sem segmentação reconhecem os caracteres directamente de uma imagem com texto. A maioria destes métodos está baseada no léxico e consideram uma palavra como uma entidade indivisível, fazendo o reconhecimento usando *features* da palavra como um todo. Assim, estes são normalmente chamados *métodos holísticos*. Madhvanath e Govindaraju [81] escreveram um *survey* sobre métodos

holísticos e afirmam que o tratamento de palavras com léxico em classes de padrões distintas limita a aplicação apenas para cenários com um léxico pequeno e estático. Rocha e Pavlidis [98] propõem um método sem segmentação, que converte uma imagem texto em grafos com *features* e então são procurados os *matches* dos sub-grafos de *features* com os protótipos dos caracteres predefinidos. Uma procura final pelo caminho óptimo na rede dá a melhor interpretação do texto da imagem.

Recentemente, Song et al. [109] propuseram uma solução para o problema, baseando a sua análise nos três tipos de relações de conexão entre caracteres, nomeadamente "linear", "não-linear" e "sobreposto". Considerando que na segmentação de uma palavra os caracteres do lado esquerdo mantêm sempre a forma original, é então encontrado um grupo de candidatos, analisando o lado esquerdo dos caracteres, verificando os candidatos pelo reconhecimento de caracteres e cortando a sub-imagem sem danificar os caracteres seguintes. O método utiliza a informação obtida no lado esquerdo do carácter conexo para prever candidatos do lado direito e então o caminho de corte segue a forma correcta do carácter reconhecido de modo a preservar a forma do carácter do lado direito.

Extracção de Features

Os tradicionais sistemas de reconhecimento de padrões utilizam um descritor de *features* e um procedimento de classificação particular para determinar a verdadeira classe de um dado padrão. Para problemas envolvendo um elevado número de classes e entradas com ruído, é difícil obter uma correcta classificação.

Devijver e Kittler [24] definem a extracção de *features* como o problema de extrair dos dados brutos, a informação que é mais relevante para o objectivo da classificação, no

sentido de minimizar a variabilidade das classes padrão existentes, enquanto evidência a variabilidade entre as classes.

Mori et al. [85] escreveram um resumo histórico dos avanços teóricos do reconhecimento de caracteres, bem como dos avanços dos sistemas de OCR comerciais. Os autores dividem os sistemas em duas principais classes de análise, os métodos de *template-matching* e os métodos de análise da estrutura de caracteres. Para cada classe são descritos vários métodos utilizados desde o início do OCR.

Trier et al. [128] descrevem vários tipos de métodos para a extracção de features no reconhecimento de caracteres isolados. A selecção do método de extracção de *features* no reconhecimento *off-line* é provavelmente o factor mais importante para obter uma elevada taxa de reconhecimento nos sistemas de reconhecimento de caracteres. Os diferentes métodos de extracção de *features* foram realizados para diferentes representações de caracteres, por exemplo, caracteres binários sólidos, contorno dos caracteres, *skeletons* (caracteres com *thinning*) e sub-imagens a *gray level* de cada caracter individual. Para cada representação são apresentados vários métodos de análise, nomeadamente *template matching*, *templates* deformáveis, transformada unitária, descritor de grafos, projecção de histogramas, análise do contorno, *features* discretos, *zoning*, momentos geométricos, momentos de Zernike e momentos de Fourier. Estes métodos de extracção de *features* são discutidos em termos de propriedades invariantes, reconstrutividade, distorções esperadas e variabilidade dos caracteres.

Classificação

O problema da classificação das formas dos caracteres é definido como a minimização entre as possíveis transformações que mapeiam a forma de entrada às formas protótipo [98]. Para classificar padrões é necessário treinar o sistema para realizar a aprendizagem, uma base de conhecimento e a capacidade de usar esta informação juntamente com a informação contextual [19].

A taxa de reconhecimento também depende do tipo de classificadores utilizados. Os classificadores são procedimentos estatísticos que colocam itens individuais em grupos, baseados na informação quantitativa sobre uma ou mais características inerentes aos itens e baseados no treino de classificação. Formalmente, o problema pode ser colocado como: para os dados de treino $\{(x_1, y), \dots, (x_n, y)\}$ produz-se um classificador $h : X \rightarrow Y$, o qual mapeia um objecto $x \in X$ à sua label de classificação $y \in Y$.

Depois de obtidos os vectores de *features* no final do processo de descrição, estes são usados para realizar a aprendizagem do sistema. A aprendizagem é a função básica do reconhecimento de padrões e são usados métodos bem estudados da literatura, entre os quais redes neuronais, algoritmos genéticos, sistemas difusos, *expert systems*, *k-NN clustering*, etc. Depois do treino do sistema utilizando os dados de entrada de treino, estes métodos estatísticos são usados para realizar a classificação no processo de reconhecimento de padrões.

2.5.6 Avaliação de Performance

O problema da interpretação automática de imagens em desenhos de engenharia ou reconhecimento de gráficos, tem sido estudado ao longo do tempo. Recentemente novos

métodos de vectorização foram propostos e muitos sistemas comerciais de vectorização já foram desenvolvidos. No entanto, existe frequentemente a necessidade de uma comparação de avaliação neste campo, para medir o progresso realizado no tempo. Sem concursos de avaliação de performance ou *benchmarks*, o futuro progresso do campo não é de todo possível, ou pelo menos não é mensurável.

A avaliação da performance no reconhecimento de gráficos é um campo relativamente novo; alguns métodos objectivos e quantitativos para o reconhecimento de gráficos foram recentemente propostos [66],[50],[136]. Kong et al. [66] propuseram um método quantitativo para avaliar o reconhecimento de linhas a tracejado. Hori e Doermann [50] propuseram uma metodologia de medição quantitativa de performance para tarefas específicas de conversão de *raster* para vectores. Wenyin e Dori [136] apresentaram um protocolo de avaliação no reconhecimento de linhas rectas e círculos. Todos estes métodos são limitados na sua aplicação.

Kong et al. [66] usam o ângulo, a distância, a relativa sobreposição e o deslocamento entre os segmentos de linha, para avaliar os encontros das linhas e a detecção dos estilos das linhas. Eles utilizam vários *thresholds* arbitrários e rígidos, contudo também não permitem a fragmentação de linhas detectadas.

Hori e Doermann [50] utilizam e estendem o trabalho de Haralick [45] para a caracterização da performance na análise de imagem, numa maneira dependente da aplicação, medindo a performance dos algoritmos de conversão *raster* para *vector*. As aplicações consideradas neste trabalho são o *thinning*, a procura de linha média, e o *line fitting*, todas elas são técnicas de baixo nível que não constituem completamente a vectorização. É difícil estender o trabalho para avaliar um sistema completo de vectorização. O protocolo de Hori e

Doermann não distingue entre a taxa de reconhecimento e a taxa de falsos alarmes. Não inclui uma métrica de avaliação global, nem permite a fragmentação de linhas detectadas.

Wenyin e Dori [136] propuseram índices de avaliação de performance para a detecção de linhas rectas e circulares. As taxas de detecção e dos falsos alarmes são definidos para os níveis pixel e vectorial. A utilização de índices de performance a nível pixel (medidas de preservação da forma) não é apropriada para imagens reais que contenham muitas distorções introduzidas no processo de digitalização. Em tais imagens, a tentativa de obter índices de pixel com alta precisão, iria inutilmente necessitar de que os vectores detectados fossem verdadeiros, i.e. sem formas distorcidas nas linhas da imagem e sem fragmentação das linhas detectadas. Para tais imagens, o índice de descoberta do pixel necessita de ser associado a um peso menor do que o índice dos vectores. Contudo, não existe uma maneira de predeterminar o peso relativo para os índices de descoberta dos pixeis e dos vectores.

Phillips e Chhabra [94], apresentaram um método empírico para avaliar sobre a performance dos sistemas de reconhecimento. A metodologia é realizada de modo a trabalhar com imagens que contenham linhas rectas sólidas e a tracejado, círculos, e arcos circulares, e tão bem como zonas com texto. As entidades gráficas são limitadas aos tipos acima enunciados. A metodologia de avaliação é bastante útil, uma vez que os desenhos de engenharia são uma combinação desses elementos geométricos. Esta metodologia difere dos trabalhos anteriores em duas formas distintas. Primeiro, pode avaliar completamente um sistema de *raster* para *vector* e não apenas um simples módulo, tal como acontecia na detecção de linhas a tracejado. Segundo, as métricas de performance são orientadas a objectivos, i.e. são motivadas pelo último objectivo de conversão para um dado desenho, o que envolve algum trabalho manual para corrigir as falhas do sistema. A metodologia inclui um conjunto de

critérios de *match* para pares de entidades gráficas, um conjunto de métricas para avaliação de performance e um *benchmark* para a avaliação do sistema de reconhecimento. A metodologia disponibiliza uma comparação empírica de pacotes de *software* para vectorização. Utiliza métodos de avaliação de performance práticos que podem ser aplicados a sistemas de vectorização completos.

2.6 Sumário

A análise de imagem em documentos apresenta uma ligação com outras áreas do conhecimento, nomeadamente o processamento digital de imagem e a análise de padrões. Primeiramente foi apresentada uma revisão sobre os principais métodos utilizados em análise de documentos, partindo dos algoritmos binários básicos até aos métodos de reconhecimento de imagem em documentos.

De seguida, foram apresentados os modelos de degradação de documentos para simular as possíveis perturbações desde a fotocópia até ao *scanning* e assim estudar o seu efeito nos algoritmos de reconhecimento de imagem. Depois foram apresentados os algoritmos de processamento de imagem binária, que se encontram muito bem estudados na literatura e que são a base de qualquer trabalho na área de reconhecimento de imagem. Seguidamente foram revistos vários métodos e algoritmos de reconhecimento de imagem, que permitem segmentar e reconhecer elementos característicos como por exemplo linhas rectas, círculos, arcos, símbolos, caracteres, etc. Finalmente foram apresentados alguns algoritmos para a avaliação de performance tornando mensurável o reconhecimento de imagem de cada método.

Capítulo 3

Trabalho Relacionado

3.1 Desenhos Técnicos

Em variados domínios técnicos, tais como na engenharia e na arquitectura é muitas vezes necessário apresentar a informação na forma gráfica, uma vez que deste modo torna mais rápida e fácil a sua interpretação. Os desenhos técnicos contêm informação gráfica de um dado domínio técnico, apresentando uma semântica específica, com várias regras, considerando uma escala de medição e são elaborados com uma elevada precisão. São exemplos, os desenhos de arquitectura [33, 34, 127], os desenhos de mecânica [135], os mapas de telefone [5], os mapas cadastrais [104, 53], os mapas topográficos [6, 117], os mapas cartográficos [137, 70] ou os mapas hidrográficos [74, 129].

3.1.1 Mapas Cadastrais Portugueses

Em Portugal o único cadastro em vigor é o Cadastro Geométrico da Propriedade Rústica, adiante designado CGPR, que está representado em mapas mais precisamente em secções

cadastrais às escalas 1:500, 1:1000, 1:2000, 1:2500 e 1:5000. Uma secção cadastral é uma planta topográfico-cadastral que representa, a uma determinada escala, uma projecção do território abrangendo os prédios rústicos. Este conjunto de prédios está desenhado num suporte inextensível de área útil 80cmX60cm e segundo normas e especificações determinadas.

Estes mapas foram desenhados entre 1940 e 1999, para representar a informação cadastral em Portugal. No entanto, o levantamento cadastral apenas foi realizado a sul do rio Tejo, devido à intensa densidade florestal no norte do país. A maioria dos mapas cadastrais apresentam um baixo nível de conservação, no entanto para evitar uma crescente degradação dos mesmos, o IGP ¹ procedeu a uma acelerada informatização dos mesmos. Segundo o IGP, todos os mapas já se encontram digitalizados e georeferenciados, estando prontos a ser trabalhados. Infelizmente, a digitalização é apenas o primeiro passo de um moroso processo, que envolve a vectorização, interpretação e validação da informação cadastral, para uma posterior integração num SIC ² central.

Semântica da Entidade Cadastral

Estando o processo de informatização do CGPR sob a alçada do IGP, é realizada seguindo as especificações técnicas para o Cadastro Geométrico da Propriedade Rústica [89]. Este documento elaborado pelo IGP, enumera as especificações técnicas da informatização e descreve sumariamente a semântica dos elementos constituintes das secções cadastrais portuguesas. Nele estão também especificados os métodos de rasterização, georeferenciação e vectorização.

¹Instituto Geográfico Português

²Sistema de Informação Cadastral

Na figura 3.1 podemos observar a imagem de uma secção cadastral digitalizada. No cabeçalho das secções cadastrais, existe um conjunto de informação alfanumérica onde pode estar indicado o número de prédios, a nomenclatura da secção, a escala, a freguesia, o concelho, o nome do Instituto Geográfico Português ou organismos que o antecederam, ano de levantamento cadastral e área total da secção. Dentro dos limites da quadricula da folha do desenho do mapa cadastral encontram-se legendas que indicam os prédios deslocados, os prédios suprimidos resultantes de processo de reclamação administrativa, os números de prédio não aplicados na secção, a localização geográfica e a escala do mapa. Cada secção cadastral apresenta cruces georeferenciadas, de modo a associar as coordenadas do mapa às coordenadas geográficas reais, permitindo assim a exportação dos dados directamente para um SIG.

No lado esquerdo da figura 3.1, podemos observar mais pormenorizadamente, os elementos gráficos que constituem o mapa cadastral rústico. O *prédio* é o elemento básico de um mapa cadastral e é identificado através de um número. O prédio é descrito por segmentos de recta contínuos, designados por extremas, que formam um polígono fechado, identificando o limite da propriedade. Sobre o limite do prédio existem marcos, que são pequenos círculos completamente preenchidos a preto e que representam os marcos existentes no terreno. Cada prédio tem no seu interior um círculo com seu respectivo número, representando uma entidade cadastral. Quando o prédio é apenas composto por um tipo de parcela, no seu interior é descrito o tipo de terreno através duma sigla (como por exemplo CA, Mt, Ar).

Dentro de cada prédio podem existir várias *parcelas* dependendo da existência de diferentes tipos de terreno. O limite de uma parcela é representado por linhas a tracejado,

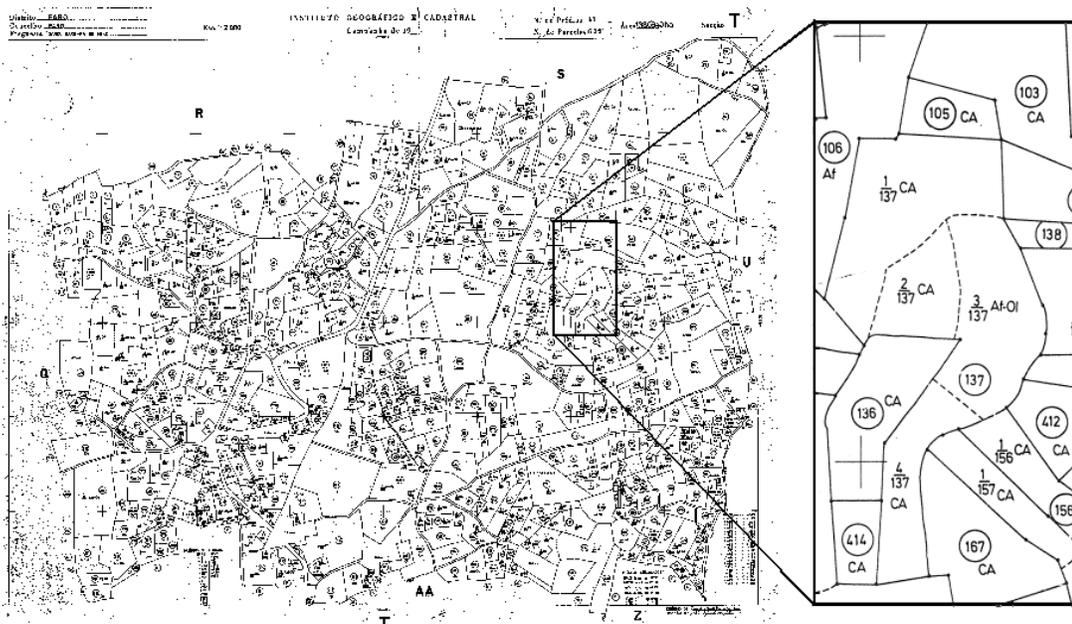


Figura 3.1: Visão geral de uma secção cadastral digitalizada. No lado esquerdo podemos observar um mapa cadastral rústico digitalizado com tamanho de 10784x7853 pixeis e resolução 300 dpi. No lado direito podemos visualizar os elementos que constituem um mapa cadastral, nomeadamente círculos, arcos, cruces, linhas contínuas, linhas a tracejado e texto. A cada elemento está associada uma semântica.

contendo cada uma delas um número de parcela e um tipo de terreno. O número da parcela apresenta a forma de uma fracção, onde o numerador representa o número da parcela e o denominador o número do prédio.

Além dos elementos acima descritos existem outras entidades cadastrais, tais como limites administrativos, marcos de concelho/freguesia e construções, contudo pela sua complexidade e fraca ocorrência não foram considerados neste trabalho, não se justificando desenvolver um algoritmo para a sua detecção.

3.2 Investigação na Área

Nos últimos 30 anos surgiram numerosas contribuições de investigadores na área de reconhecimento de imagem [120] e análise de documentos [86]. O desenvolvimento de novas técnicas de análise e o estudo de casos particulares permitiu ultrapassar variados problemas, no entanto ainda existem bastantes desafios [121].

Como aconteceram consideráveis avanços nesta área, é extremamente difícil enumerá-los pela sua importância, contudo é importante referenciar alguns trabalhos que permitiram orientar e qualificar novas pesquisas na área. Kasturi et al. [61, 63, 62] estudaram métodos de extracção de informação de desenhos de engenharia, desenvolvendo e aperfeiçoando métodos de reconhecimento de imagem. Dori and Wenyin [29] desenvolveram um sistema de vectorização para desenhos mecânicos, fazendo primeiramente uma descrição baseada em segmentos de recta e detectando posteriormente outras formas mais elaboradas, tais como círculos ou arcos. Dosch et al. [34] estudaram um sistema de reconhecimento de desenhos de arquitectura, também utilizando segmentos de recta provenientes do processo de vectorização. Por outro lado, Song et al. [112] estudaram o processo de reconhecimento de desenhos de engenharia usando uma abordagem de reconhecimento directamente a partir da imagem (nível pixel).

O problema do reconhecimento de imagem pode ser visto de duas perspectivas diferentes, através do reconhecimento directo na imagem ou baseando o reconhecimento, na análise dos segmentos de recta provenientes do processo de vectorização. Song et al. [110] descreveram os principais métodos de reconhecimento de imagem para cada abordagem, identificando as vantagens e desvantagens em cada situação. Infelizmente, ambas podem reconhecer primitivas gráficas numa imagem, mas cada uma delas apresenta problemas.

Song et al. [107] estudaram o reconhecimento de imagens binárias usando cada uma das abordagens e sugeriram utilizar ao mesmo tempo a combinação das duas, pois nenhuma delas de forma isolada é perfeitamente capaz de ultrapassar todas as dificuldades.

O reconhecimento de imagem ao nível pixel deve ser implementado de um ponto de vista hierárquico, reconhecendo desde as primitivas gráficas mais simples até às primitivas mais complexas. Song et al. [114] criaram um modelo conceptual para o reconhecimento de imagem, considerando que um desenho de engenharia pode ser visto como uma colecção de linhas, arcos, curvas, símbolos e palavras. Em cada processo de reconhecimento, os elementos detectados são eliminados de modo a simplificar as seguintes detecções. Desta forma, considera o problema de reconhecimento de um ponto de vista global, e não numa análise método a método como até então. Wenying [135] e Dosch [33] reconheceram por camadas, encontrando círculos e arcos a partir dos segmentos de recta, e implementado o reconhecimento subsequente usando a semântica do problema. Quaisquer uma das duas abordagens apresenta uma grande limitação, necessita de conhecer a semântica do problema, para aumentar a taxa de reconhecimento e validar a informação reconhecida.

A digitalização automática também já foi aplicada noutras áreas. Levachkine [70] estudou a conversão de *raster* para *vector* de mapas cartográficos com cor, implementando algoritmos de reconhecimento de padrões cartográficos em mapas *raster*. Lu et al. [127] estudaram o reconhecimento e interpretação de desenhos de arquitectura para o reconhecimento das dimensões, sistema de coordenadas e componentes estruturais. Arias et al. [5] estudaram a extracção automática de informação nos mapas de linhas de telefones.

Relativamente aos mapas cadastrais, é na França, de onde surgem os maiores contributos, nomeadamente Shahbazkia [104] estudou a extracção de informação do mapa cadastral

francês e Viglino et al. [53] ainda estão a pesquisar algoritmos para a extração do contorno. Embora os cadastros assentem nos mesmos princípios os mapas cadastrais franceses apresentam características e problemas distintos dos portugueses.

3.3 Estratégias Fundamentais

O reconhecimento de imagem de documentos técnicos é um processo complexo e por esse motivo é necessário considerar estratégias simplificadoras [114, 123, 122]. No problema de análise de imagem existe sempre um elevado número de possíveis resoluções, no entanto, muitas dessas estratégias são limitativas e pouco flexíveis à partida, porque cada passo melhora a imagem num determinado sentido, mas piora noutra, assim o problema tem de ser analisado de um ponto de vista global [110].

Neste caso concreto, podemos imaginar o problema da escolha dos métodos de reconhecimento, como um problema de procura do caminho mais próximo, minimizando a distância entre uma imagem e o seu completo reconhecimento, em que o caminho é percorrido através da utilização de métodos de análise específicos. Existem muitos caminhos que podem levar ao reconhecimento de uma imagem, mas sem dúvida que existem alguns que se aproximam mais rapidamente e com maior flexibilidade do que outros. Nesta escolha, é fundamental a experiência na área e o estudo dos métodos de análise [123], de modo a encontrar métodos de análise robustos [122].

3.3.1 Processamento em Imagem Binária

Inicialmente é digitalizada uma folha contendo um mapa cadastral, sendo obtida uma imagem *raster* pronta a ser processada. Nesta digitalização e dependendo do *scanner*, pode ser obtida uma imagem binária, uma imagem em escala cinza ou uma imagem colorida. Na literatura existem alguns trabalhos [129, 130] que consideram o processamento a partir de uma imagem em escala cinza, porque nesta análise existe mais informação que potencialmente poderá facilitar a resolução de problemas, como por exemplo na sobreposição de informação. Na imagem colorida é mais fácil a segmentação de informação baseada na cor dos elementos, devido a que cada elemento apresenta um cor distinta, representando cada cor uma camada de informação.

No caso de considerarmos uma imagem, com uma palavra com letras conexas, à partida, a análise em escala cinza apresenta vantagens, por existir mais informação para decidir onde segmentar cada letra. Por vezes, mesmo para especialistas humanos não é possível identificar um símbolo, mas no senso comum é aceite que é mais provável identificar um símbolo correctamente numa imagem cinza do que numa imagem binária, por existirem mais pistas.

A análise em escala cinza implica mais informação, logo mais memória e menor velocidade de processamento, contudo mais importante do que tudo isso é considerar o tipo de documento em estudo. A análise em escala cinza pode ser mais vantajosa quanto mais problemas de segmentação existirem, devido à sobreposição de informação. Na literatura é utilizada na maioria das vezes a análise binária, porque são poucos os casos onde a diferença entre a taxa de reconhecimento nos dois tipos de análise é significativa.

Para simplificar o processo de reconhecimento, a imagem em escala cinza é trans-

formada em imagem binária através da aplicação do processo de *threshold*. Na literatura existem vários métodos de *threshold* (ou binarização), nomeadamente métodos locais [131, 132], globais [39] ou adaptativos [102, 41]. O objectivo destes métodos é transformar uma imagem cinza numa imagem binária sem perder informação significativa.

Os mapas cadastrais foram digitalizados com uma resolução de 200 dpi, permitindo haver bastante detalhe, e através da aplicação do *threshold* de *niblack* [88] não é perdida qualquer informação. Toda a análise de imagem efectuada nesta tese é realizada a nível binário.

3.3.2 Extracção de Informação Sem Pré-Processamento

Usualmente, as imagens são processadas para evidenciar pequenos detalhes, através da aplicação de filtros ou de outras ferramentas de processamento de imagem. A aplicação destes métodos pode melhorar significativamente uma imagem, no entanto também pode introduzir novas características à imagem. Estes métodos são aplicados utilizando parâmetros escolhidos pelo utilizador e pretende-se um sistema sem parâmetros e de reconhecimento automático.

Neste caso, a estratégia mais flexível é não realizar pré-processamento [110, 122], de modo, a que a imagem apresente sempre características normalizadas e não facilitar o reconhecimento do sistema, aplicando métodos apenas ajustáveis pelo utilizador. Ao invés de tentar ultrapassar os problemas iniciais de reconhecimento realizando um pré-processamento, é aconselhável não alterar a imagem, para que os métodos resolvam por si mesmos os problemas inerentes ao reconhecimento.

Por exemplo, na extracção do contorno existem problemas devido às descontinuidades

do contorno e até seria apelativo, realizar um pré-processamento à imagem de modo a tapar essas descontinuidades e assim poder aplicar um algoritmo de extracção simples. Contudo, a robustez do reconhecimento assenta na escolha de métodos que considerem esses problemas. Assim, de modo a obter robustez na análise da imagem é feito o reconhecimento sem realizar pré-processamento.

3.3.3 Análise Hierárquica

O reconhecimento de imagens é normalmente realizado directamente na imagem ou então baseado no processo de vectorização [107]. Wenyin [135], Dosch [33] e Hilaire [46] estudaram processos de reconhecimento de imagem realizando dois passos distintos, primeiramente realizando a vectorização da imagem e depois fazendo um pós-processamento para detectar formas geométricas mais complexas, de modo a reconhecer as primitivas básicas da imagem. Esta abordagem apresenta problemas quando existe conexão de linhas ou a informação está sobreposta, ocorrendo deformações nos vectores [123].

Como os mapas cadastrais portugueses têm elementos conexos ao contorno, uma vectorização seguida de um reconhecimento de formas, introduziria várias más detecções. Assim, foi escolhida a análise da imagem a nível pixel que embora mais lenta torna o processo menos complexo [123]. Além disso, a abordagem a nível pixel não necessita de um vectorização inicial e o reconhecimento é realizado directamente na imagem, observando-se conjuntos de pixels. Neste caso, o reconhecimento é realizado de uma forma hierárquica, partindo dos objectos mais simples para os mais complexos. Os elementos reconhecidos são apagados da imagem de modo a não interferir em futuras detecções. O modelo conceptual utilizado por Song et al. [110] é representativo deste tipo de análise.

Os mapas cadastrais portugueses são constituídos na sua grande maioria por segmentos de rectas, círculos e caracteres. Na vectorização de caracteres acontecem deformações devido às junções (T,X,L como descrito em [123]), uma vez que nos pontos de junção acontecem deformações. Aplicar directamente a vectorização sobre as linhas contínuas do contorno é possível, mas não é um método robusto, por existirem caracteres conexos ao contorno. Também é necessário realizar a validação do contorno, uma vez que o contorno só é válido se for fechado. Deste modo, a validação da análise vectorial é consideravelmente mais complexa, dado que é necessário agrupar e seguir os segmentos extraídos. Assim, seria necessário outro processamento de modo a encontrar zonas fechadas, partindo de uma lista de segmentos dispersos. Utilizando a análise a nível pixel, o processo é consideravelmente mais simples porque é vectorizado prédio a prédio.

Numa visão sintética, a análise a nível pixel e hierárquica é aconselhável, mas tem o inconveniente de considerar a semântica do problema. De notar que este inconveniente existe sempre para qualquer tipo de reconhecimento complexo, quer a análise seja efectuada directamente a partir da imagem ou partindo do processo de vectorização, necessitando sempre de encontrar-se uma estratégia para cada caso de estudo.

3.3.4 Dados de Treino

No treino de sistemas de aprendizagem [101], tal como acontece em redes neuronais, algoritmos genéticos ou sistemas difusos, o sistema realiza a aprendizagem através do treino, utilizando para tal dados de entrada que designam-se como dados de treino. Quando não existe uma quantidade de dados suficiente para realizar este treino, são sintetizados novos dados, introduzindo variações ao nível do ruído e da deformação, de modo a considerar as

principais condições adversas ao reconhecimento. O objectivo é generalizar o conhecimento adquirido na utilização dos dados de treino aos outros dados existentes.

Kanungo et al. [58] estudaram o modelo de degradação de documentos, introduzindo um modelo paramétrico que descreve as principais transformações durante o processo da cópia. Deste modo, é possível simular a degradação de documentos introduzindo ruído e distorção, de maneira a testar os processos de reconhecimento de imagem, considerando estes dois factores indesejáveis à detecção.

Geralmente, para avaliar a performance dos métodos de reconhecimento, são inicialmente sintetizadas imagens sem ruído ou qualquer outras deformações, no passo seguinte é adicionado ruído e distorção utilizando um modelo de degradação e, finalmente são aplicados os métodos de reconhecimento, testando a sua performance. As taxas de ruído e distorção são aumentadas gradualmente, de maneira a estudar o nível de tolerância dos métodos de análise relativamente à degradação da imagem. Esta metodologia é bastante utilizada em sistemas de reconhecimento de caracteres [1] ou no estudo da performance dos métodos de análise às variações do ruído [94], no entanto é considerado mais promissor estudar os métodos de análise usando a maior quantidade de dados reais possível, ao invés de tentar abordar o problema usando mapas sintéticos. Esta última abordagem só deverá ser considerada caso exista pouca quantidade ou pouca variedade de dados de treino.

O sistema de reconhecimento de imagem descrito neste trabalho foi implementado utilizando de 15 mapas cadastrais reais, permitindo treinar o sistema e compreender os problemas característicos da análise, de modo, a generalizar os métodos de análise à variabilidade dos mapas cadastrais existentes. Desta forma, os problemas abordados são efectivamente

reais e a análise dos problemas terá sempre aplicabilidade.

3.4 Sumário

Neste capítulo foram introduzidas as secções cadastrais, explicando-se as especificações e as normas do cadastro. Para este domínio de análise foram evidenciadas as estratégias fundamentais de análise, considerando a experiência dos trabalhos descritos.

Capítulo 4

Extracção de *Features*

Um mapa cadastral é um documento complexo, composto por uma enorme quantidade de entidades e objectos cadastrais, nomeadamente prédios e parcelas. Nalguns casos, apenas num mapa cadastral existem mais de 400 prédios e dentro de cada prédio podem existir várias parcelas com uma vasta informação no seu interior. Estes mapas foram desenhados à mão, usando primitivas básicas de desenho, tais como segmentos de recta, círculos, arcos, curvas e linhas a tracejado. Neste sentido, é importante evidenciar a enorme variedade e quantidade de informação existente em cada secção cadastral, o que aumenta consideravelmente a complexidade do reconhecimento dos mapas cadastrais.

No processo de digitalização, o *scanner*¹ introduz deformações na imagem do documento, tais como ruído e distorção. Como o problema do ruído é sistemático, todos os métodos de análise realizam uma análise probabilística à imagem, de modo a classificar os objectos cadastrais em classes distintas. Através da análise probabilística todos os elementos são classificados na classe mais provável.

¹Dispositivo de digitalização de documentos.

Uma vez que estamos a trabalhar no domínio específico dos mapas cadastrais portugueses, é utilizado o conhecimento prévio da semântica cadastral, de modo a simplificar todo o processo de análise. Certamente que os métodos que analisam a imagem directamente a nível pixel são mais úteis para este caso, em vez de realizar uma primeira vectorização, tal como no trabalho efectuado por Dosch et al. [34]. Através da utilização desta abordagem não só elevada-se a flexibilidade do processo de reconhecimento, como também o conhecimento prévio torna-se mais fácil de implementar.

Existem duas maneiras distintas que descrevem como uma máquina pode reconhecer um documento, descrevendo primeiro as primitivas básicas e depois interpretando-as (partindo do particular para o global), ou de um outro modo, considerando a imagem como um todo, sem atender às primitivas básicas pelas quais uma imagem é constituída (partindo do global para o particular). Como neste caso, o reconhecimento está orientado a objectivos semânticos, nomeadamente prédios e parcelas, assim considerar na extracção do contorno, por exemplo, a detecção de segmentos de recta e a intersecção de linhas, teria como consequência aumentar a complexidade do problema. Deste modo é considerado o reconhecimento das entidade semânticas, numa perspectiva do global para o particular.

Em cada imagem dos mapas cadastrais existem elementos gráficos que se repetem, apresentando-se como padrões visuais. Esses elementos apresentam simultaneamente características semelhantes e distintas entre si, e é pretendido considerar estes factores bivalentes, de maneira a agrupar os objectos gráficos em classes iguais, tornando as classes distintas entre si. Nas secções seguintes são estudados os métodos de análise na detecção dos padrões existentes nos mapas cadastrais portugueses.

4.1 Detecção de Cruzes

A cruz é o objecto cadastral mais simples que podemos encontrar nos mapas cadastrais Portugueses e é definida como tendo dois segmentos de recta perpendiculares e com o mesmo comprimento. Felizmente, a orientação da cruz e o diferencial do comprimento entre ambos os segmentos de recta, são parâmetros constantes em todas as secções cadastrais, desta forma, a posição da cruz é o único parâmetro na sua detecção.

Uma vez que este objecto gráfico tem uma forma estática e bem definida, o método de detecção é simples e pode ser implementado de diferentes maneiras, nomeadamente através de métodos de *template matching* ou de extracção de características. Primeiramente foi implementada a abordagem utilizando o método de *template matching* [54], mas este não é utilizado porque é um método demasiado estático, apenas considera os pixels a preto e introduz problemas de decisão em imagens que apresentam um elevado número de elementos conexos.

Considerando a abordagem de *extracção de características* [23], a cruz é descrita tendo como principais características um segmento de recta vertical e outro horizontal, existindo uma orientação perpendicular entre ambos os segmentos. É também importante notar que utilizando esta abordagem, ambos os pixels a branco e a preto são considerados na análise. Na figura 4.1, os pixels a branco são representados com segmentos de recta a tracejado, enquanto os pixels a preto estão representados com segmentos de recta contínuos, de modo a distinguir ambas as análises.

Os pixels a branco são considerados, de modo a distinguir uma cruz do *background*, contudo existem várias maneiras de realizar esta distinção. Por exemplo, usando quatro segmentos de recta laterais ao contorno da cruz (figura 4.1-a), considerando dois segmentos

de recta perpendiculares com orientação $\pi/2$ e $3\pi/2$ (figura 4.1-b) ou usando dois segmentos de recta perpendiculares com origem numa extremidade da cruz (figura 4.1-c). Esta escolha está dependente da distinção entre a cruz e o *background*, por exemplo, se existirem muitas zonas com pixels a preto, então o método a) é utilizado ou se o *background* for normalmente branco então é suficiente utilizar o método c). Neste caso, o método c) foi o escolhido porque apresenta uma mais rápida performance computacional. Adicionalmente, são definidos quatro pequenos segmentos de recta nas extremidades da cruz (ver figura 4.1) para garantir a simetria deste elemento, i.e., um comprimento igual em todos os segmentos e a existência de um ponto central no centro da cruz.

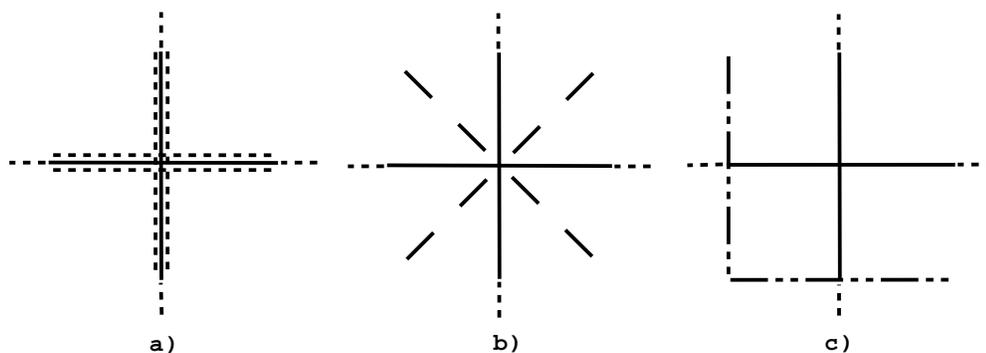


Figura 4.1: É possível distinguir uma cruz do *background* de diferentes maneiras, usando: a) quatro segmentos de recta, b) dois segmentos de rectas oblíquos ou c) dois segmentos de recta perpendiculares. De notar também, que as linhas contínuas representam o *match* com os pixels a preto, enquanto as linhas a tracejado representam o *match* com os pixels a branco. São definidos quatro pequenos segmentos de recta nas extremidades da cruz para garantir a simetria deste elemento.

As características principais da cruz foram previamente descritas, no entanto precisavam de ser definidas de modo a implementar um método de reconhecimento preciso. A equação 4.1 calcula a probabilidade de ter quatro pequenos segmentos de recta, cada um numa direcção distinta, nos extremos na cruz.

$$P_{linhas_{finais}} = P_{final_{cima}} + P_{final_{baixo}} + P_{final_{esquerda}} + P_{final_{direita}} \quad (4.1)$$

Considerando o método c) da figura 4.1, definimos a equação 4.2 para calcular a probabilidade de ter dois segmentos de recta a preto à esquerda e por baixo relativamente ao centro da cruz.

$$P_{linhas_{background}} = P_{esquerda_{background}} + P_{baixo_{background}} \quad (4.2)$$

Os pixels a preto são também considerados, utilizando dois segmentos de recta contínuos e perpendiculares, como mostra a figura 4.1.

Considerando todas estas características, elabora-se a equação 4.3, onde p_{linha_h} e p_{linha_v} representam a taxa de pixels a preto nos eixos vertical e horizontal respectivamente, enquanto $p_{linhas_{finais}}$ e $p_{linhas_{background}}$ representam a taxa de pixels a preto nos extremos da cruz e no *background*. Desta forma, a probabilidade da cruz p_{cruz} , é calculada considerando todos os elementos previamente descritos, de modo a implementar o método da figura 4.1-c).

$$p_{cruz} = p_{linha_h} + p_{linha_v} - p_{linhas_{finais}} - p_{linhas_{background}} \quad (4.3)$$

Por outro lado, é também possível validar a detecção de todas as cruzes em conjunto e numa perspectiva global, uma vez que a distância entre as cruzes é constante em ambos os eixos (xx e yy). Adicionalmente e como regra, a distância entre duas cruzes deve ser maior do que o tamanho máximo de cada cruz. Quando esta situação não acontece, o objecto com menor probabilidade é apagado e não é considerado na detecção.

Como por vezes as cruzes estão conexas a outros elementos, são utilizadas máscaras para apagar as cruzes da imagem de detecção, tal como mostra a figura 4.2. As cruzes são

apagadas seguindo o eixo de cada segmento de recta que a constitui a cruz e eliminando-se a linha perpendicular ao segmento da cruz, caso não exista nenhum elementos conexos a essa parte da cruz.

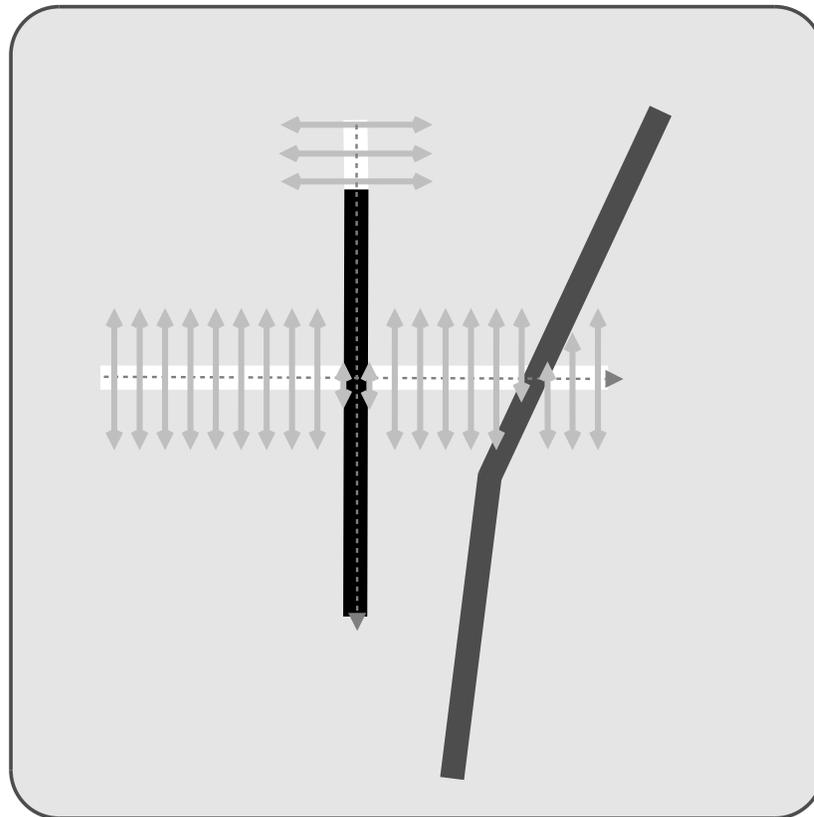


Figura 4.2: As cruzes são apagadas dos mapas cadastrais utilizando máscaras, isto é, seguindo o eixo de cada segmento de recta e apagando na direcção perpendicular ao segmento em análise, apenas se não existir qualquer elemento conexo a essa parte da cruz.

4.2 Detecção de Círculos

A detecção de círculos é *à priori* um problema simples e não apresenta muitas dificuldades, no entanto as imagens cadastrais apresentam deformação, distorção, ruído. E como os círculos apresentam tamanho heterogéneo, a detecção é ainda invariante ao tamanho.

Na literatura existem vários métodos que realizam esta detecção, tais como a Transformada de Hough [51] ou a Transformada de Radeon [133]. O método mais utilizado neste tipo de problema é a Transformada de Hough, mas como é um algoritmo bastante pesado computacionalmente, foram realizadas várias optimizações a esta abordagem [8, 83], de modo a reduzir o tempo despendido na sua aplicação. Adicionalmente, na análise realizada pela Transformada de Hough apenas são considerados os pixels a preto. Desta forma será necessário introduzir as características intrínsecas do círculo, de modo a considerar os pixels a branco e distinguir os círculos de outros elementos circulares.

Os pixels a branco são utilizados para considerar a forma perfeita do círculo de maneira a distinguir os círculos que têm elementos conexos dos que não têm. Assim, as características do círculo são consideradas desde o início, usando os pixels a preto e a branco, implementando um método baseado em *features* [23].

Tal como nas cruces, foi estudada uma definição do círculo para este domínio. E a primeira questão encontrada está relacionada com a forma que o círculo deve ter, nomeadamente se estamos à procura de um círculo ou de um anel. Um círculo é definido como sendo uma circunferência com uma largura de linha igual a um, enquanto que um anel é definido como sendo uma circunferência com largura de linha maior do que um. Devido à existência de ruído, um círculo pode apresentar uma forma elíptica. Considerando este facto, a procura necessita de ter intervalos de tolerância. Assim, com uma tolerância de cinco pixels, utiliza-se um anel com largura de linha igual a cinco, se estivermos a procurar um círculo com largura de linha igual a um. Contudo, se o objectivo for detectar anéis e não círculos perfeitos, é utilizada uma circunferência na sua detecção ao invés de um anel, uma vez que já existe uma tolerância na forma que se procura.

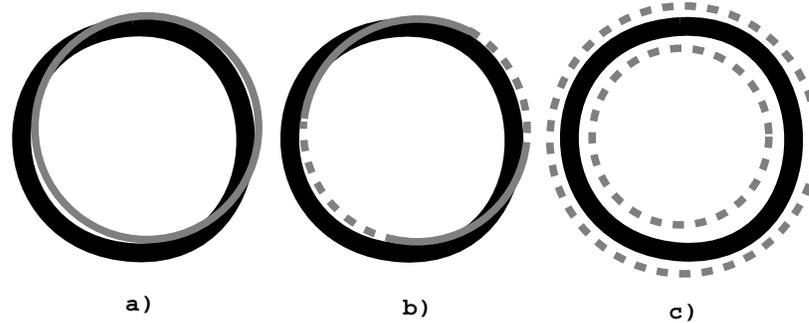


Figura 4.3: A detecção do círculo está baseada na extracção de características, nomeadamente a) a análise da percentagem de match (p_{match}), b) a análise da taxa de continuidade máxima (p_{cont}) e a taxa de descontinuidade máxima ($p_{descont}$) e c) a detecção de elementos dentro (p_{dentro}) e fora do círculo (p_{fora}). A combinação desta análise de características culmina na equação 4.5, que calcula a probabilidade de existência do círculo.

A segunda questão está relacionada com as características do círculo que devem ser consideradas na sua detecção. Um círculo pode ter múltiplas definições e esta abordagem não é uma definição genérica, mas é apenas um método genérico neste domínio, uma vez que considera todas as características intrínsecas dos círculos existentes neste tipo de mapas. Um círculo é definido como tendo uma forma circular, uma elevada taxa de continuidade e um espaço vazio dentro e fora da sua forma. Um círculo ($p_{circulo}$) tem uma forma circular tão grande quanto maior a taxa de matches (p_{match}) (ver figura 4.3-a). A continuidade é medida considerando a taxa de continuidade do maior segmento (p_{cont}) e a taxa de descontinuidade do maior segmento descontínuo ($p_{descont}$) (ver figura 4.3-b). Uma vez que não é possível detectar círculos sem considerar a distinção entre os círculos e o background, então é considerada uma circunferência vazia dentro (p_{dentro}) e fora (p_{fora}) do círculo (ver figura 4.3-c). Adicionalmente, é obtida uma heurística pelo facto da confiança de existir um círculo aumentar, se $p_{match} > 0.6$ e $p_{cont} > p_{descont}$ (ver equação 4.4).

$$p_{conf_c} = \begin{cases} 0.25, & \text{se } p_{match} > 0.6 \text{ e } p_{cont} > p_{descont} \\ 0, & \text{senão} \end{cases} \quad (4.4)$$

Depois de se encontrar algumas características e uma heurística sobre o círculo, é obtida uma relação genérica entre estes elementos na equação 4.5. Os pesos são unitários por simplicidade, mas podem ser modificados noutros casos para um melhor comportamento, no entanto neste caso não é necessário.

$$p_{circle} = p_{match} + p_{cont} - p_{descont} - p_{dentro} - p_{fora} + p_{conf} \quad (4.5)$$

A equação 4.5 calcula a probabilidade de ter um círculo numa posição genérica (x,y) , e devolve valores maiores do que 1 e menores do que 0, dado a probabilidade das características $(p_{caract}) \in [0, 1]$. Assim, os valores menores do que zero são considerados como zero e os valores maiores são considerados como 1, obtendo-se a probabilidade de existência de um círculo na posição (x,y) .

4.3 Detecção de Arcos

Os arcos são também primitivas básicas dos desenhos técnicos de engenharia e são definidos considerando a posição, o raio, o ponto inicial e o ponto final. Adicionalmente é também conhecido que um círculo é uma instância de um arco, onde o ponto inicial é coincidente com o ponto final.

Tal como acontece com outras primitivas gráficas, na literatura existem várias maneiras de realizar esta detecção, nomeadamente através dos segmentos de recta provenientes do processo de vectorização, a partir directamente da imagem ou então analisando o esqueleto

da imagem que se pretende detectar. No primeiro processo, Dosh et al. [93] melhoraram a detecção dos arcos, partindo do algoritmo implementado por Dori [29], que reconhecia arcos a partir dos segmentos de recta provenientes do processo de vectorização. Noutra abordagem, Song et al. [113] estudaram um algoritmo de detecção de arcos partindo directamente da imagem, através da procura de arcos potenciais e do seguimento desses elementos circulares, determinando os parâmetros do arco e realizando a sua validação. Por outro lado, Hilaire et al. [47] estudaram a detecção de arcos partindo do esqueleto da imagem, segmentando o esqueleto em linhas de continuidade assim, encontrando as primitivas que melhor se ajustam a essas curvas.

Todos estes métodos são muito específicos, complexos e apresentam várias dependências, alguns métodos necessitam primeiramente de uma vectorização, uma vez que os arcos são detectados a partir dos segmentos de recta, outros necessitam de calcular a imagem esqueleto, apresentando posteriormente problemas de análise em imagens com muitas junções, devido às deformações que aparecem do processo de esqueletização. A abordagem que parte directamente da imagem, parece ser a metodologia mais robusta devido ao facto de não ser introduzidas deformações, distorções ou ruído, no entanto esta metodologia de análise necessita de ser bastante consistente, uma vez que a detecção considera mais informação do que nos outros casos de análise.

A estratégia de detecção em fases distintas é usual na literatura e foi também utilizada por Song et al. [113], realizando a localização dos arcos e depois a sua validação. Esta abordagem é particularmente útil na detecção a nível pixel, uma vez que cada fase é independente e pode ser completamente separada das restantes.

Uma vez que a detecção dos círculos foi bem sucedida, também se utilizou o método de

extração de características para realizar a localização dos arcos. O processo de detecção dos arcos é realizado em três fases distintas: localizando os elementos com forma circular, estimando os parâmetros do arco candidato e realizando a sua validação. A estimação dos parâmetros do arco é realizada para encontrar o ponto inicial, o ponto final e a posição mais ajustada do centro, considerando os pontos extremos. Finalmente o arco é validado, considerando o *match* entre o arco estimado e o arco candidato.

Localização do Arco

Nesta detecção são primeiramente encontradas as zonas com elevada probabilidade de existência de arcos. Certamente que a equação da probabilidade de existência de um círculo (equação 4.5) não resolve o problema da detecção dos arcos e é necessário adicionar alguns elementos à equação para permitir uma correcta detecção. A probabilidade de *match* (p_{match}), a maior taxa de continuidade (p_{cont}) e a maior taxa de descontinuidade ($p_{descont}$) são mantidas, pois também são características do arco. Por vezes, o arco está conexo a linhas do contorno e a outros elementos, pelo que alguns pesos são diminuídos de modo a considerar os elementos dentro (p_{dentro}) e fora (p_{fora}) da circunferência. Como os arcos são círculos incompletos ou partidos, é necessário diminuir para metade estes pesos (p_{dentro} e p_{fora}). Adicionalmente, o factor de confiança também é alterado, uma vez que a taxa de *matches* e a taxa de continuidade são inferiores comparativamente ao caso dos círculos. Através de uma análise qualitativa, observamos que um arco tem sempre 30% de *matches* com um círculo perfeito e sempre mais do que 20% de continuidade relativamente ao perímetro da circunferência. Assim, surge a equação 4.6, que descreve o factor de confiança relativamente às características do arco.

$$p_{confarc} = \begin{cases} 0.25, & \text{se } p_{match} > 0.3 \text{ e } p_{cont} > 0.2 \\ 0, & \text{senão} \end{cases} \quad (4.6)$$

Uma vez que os arcos apresentam uma menor quantidade de informação comparativamente aos círculos, é adicionada uma nova característica, de modo a estudar a continuidade do arco, com tolerância a pequenas descontinuidades. A análise à continuidade com tolerância a pequenas descontinuidades ($p_{comprimento}$) é realizada procurando no contorno do círculo pelos pixels a preto, preenchendo as pequenas descontinuidades e determinando o maior arco na circunferência. O factor de probabilidade $p_{comprimento}$ é calculado, considerando a *razão* entre o comprimento do maior segmento contínuo e o comprimento do círculo ($p_{comprimento} = comprimento_{arco}/comprimento_{circulo}$). Considerando todos estes factores surge a equação 4.7.

$$p_{arco} = p_{match} + p_{cont} - p_{descont} - \frac{p_{dentro} + p_{fora}}{2} + p_{confarc} + p_{comprimento} \quad (4.7)$$

Considerando as características do arco e as suas relações (ver equação 4.7) é obtida uma primeira localização dos arcos candidatos, necessária às análises posteriores.

Determinação dos Parâmetros do Arco

Depois de localizar a posição dos arcos candidatos, são calculados numa fase seguinte os seus outros parâmetros, nomeadamente o ponto inicial (p_i), o ponto final (p_f) e também é ajustado o centro, considerando os dois pontos extremos. A análise nesta segunda etapa é realizada partindo dos parâmetros previamente detectados, designadamente da posição e do raio do arco candidato.

1. Localização dos pontos extremos do arco para arcos sem elementos conexos

Com o objectivo de determinar a trajectória do arco, i.e. a linha média com largura de pixel igual a um, que descreve o seu traçado, foi aplicado o algoritmo de *skeletonization* na zona de localização do arco candidato. Fazendo o seguimento da linha média do arco candidato são obtidos todos os pontos pertencentes à sua trajectória. E é verificado para cada ponto desta trajectória, se o ponto é uma junção ou um ponto terminal. Sendo este último o tipo de ponto onde se situam os pontos extremos do arco (p_i e p_f) (ver figura 4.4). Esta metodologia é bastante precisa, obtendo-se os pontos extremos directamente da trajectória do arco em estudo, no entanto apenas pode ser utilizada para a análise dos arcos com trajectórias bem definidas e sem elementos conexos.

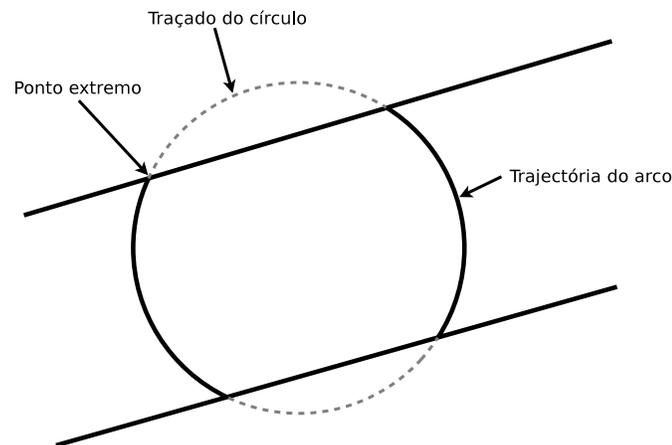


Figura 4.4: Esta figura mostra a sobreposição do traçado do círculo teórico sobre o arco detectado na fase anterior. Seguindo a trajectória do círculo são encontrados os pontos extremos do arco, nomeadamente p_i e p_f . Neste caso o arco apresenta-se com uma trajectória perfeita, não existindo descontinuidades no *match* entre o arco e o círculo.

2. Localização dos pontos extremos do arco para arcos com match parcial

Caso se verifique uma reduzida distância entre os pontos extremos pode esta-se pe-

rante dois casos: um arco com elementos conexos ou um arco não esteja completamente sobreposto ao traçado do círculo teórico. Neste último caso é realizada uma análise ao *match* entre os dois traçados. Mais detalhadamente, utiliza-se os parâmetros do círculo sobre o qual se encontra o arco candidato para realizar a sobreposição do traçado do círculo [76] ao traçado do arco em estudo, de modo a encontrar os seus pontos extremos. Como a imagem apresenta problemas de distorção e ruído, esta sobreposição apresenta pequenas descontinuidades na trajectória de *match* entre o traçado teórico e o arco em estudo. Assim, as pequenas descontinuidades da trajectória de *match* são eliminadas, ligando os dois pontos terminais vizinhos, de maneira a resolver as pequenas descontinuidades da sobreposição. Através da eliminação destas pequenas descontinuidades é obtido o maior arco possível e os seus pontos extremos (p_i e p_f) (ver figura 4.5). Mas infelizmente, devido a esta análise ser consideravelmente estática, não é possível ajustar dum forma dinâmica os pontos extremos e o centro do arco, sendo necessário realizar o ajuste do centro *a posteriori*.

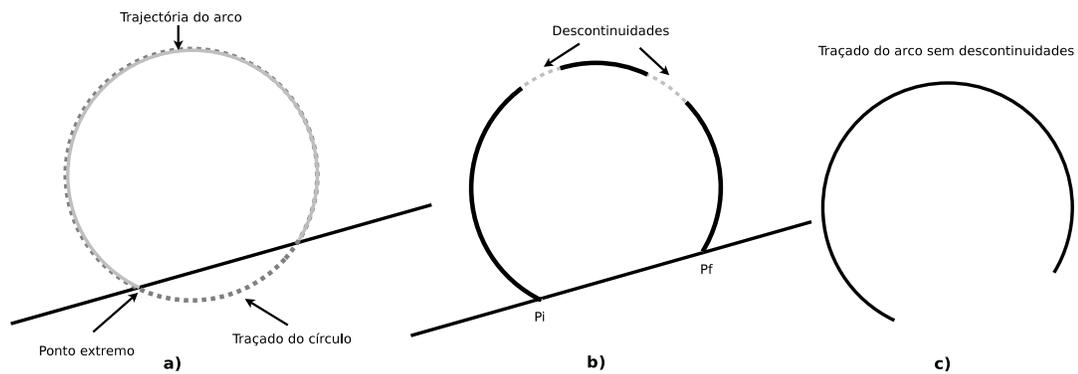


Figura 4.5: Esta figura mostra que a sobreposição das trajectórias do círculo e do arco não é perfeita e apresenta algumas descontinuidades. No entanto, as pequenas descontinuidades são eliminadas de modo a obter-se o maior arco possível e assim determinar-se os seus extremos.

3. Localização do centro

Dori [31] investigou sobre alguns métodos fundamentais da detecção de arcos em documentos, descrevendo um algoritmo de detecção do centro do arco a partir dos seus pontos extremos. Na figura 4.6 podemos observar o traçado do arco e os seus pontos extremos (A e B), a partir dos quais é encontrado um terceiro ponto (F) pertencente ao arco. Considerando C , o ponto médio do segmento AB , é traçado o segmento CD , perpendicular a AB e sobreposto ao arco. Seguindo a orientação do segmento CD é detectado o ponto D , que é o primeiro ponto com pixel a preto sobre o arco e E o último ponto na direcção do segmento do arco. Finalmente, é detectado o ponto médio do segmento DE , F , sendo o terceiro ponto pertencente ao arco.

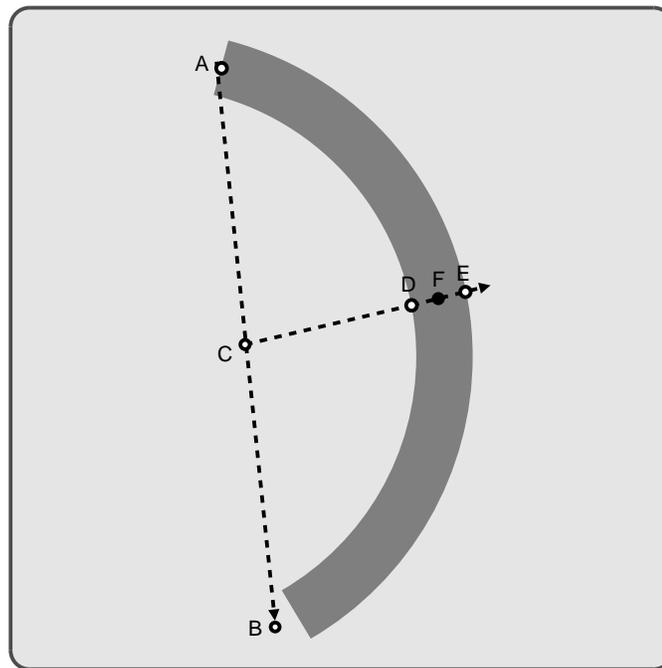


Figura 4.6: Na figura observamos que a partir dos pontos extremos do arco, A e B , encontramos um terceiro ponto pertencente ao arco (F); partindo de um segmento de recta CD que está perpendicular ao segmento AB e de seguida encontrando o ponto médio de DE situado no centro do traçado do arco.

A figura 4.7 mostra o traçado do arco e os três pontos do arco (A, B, C), é de notar que nesta figura o ponto C representa o ponto F da figura 4.6. Partindo dos três pontos do arco, dois pontos extremos e um ponto central, A, B e C respectivamente, é encontrado, através da intersecção das duas bissetrizes² perpendiculares AB e BC , o primeiro centro do arco O_0 . Considerando os pontos A, C e o centro O_0 é encontrado um novo ponto D sobre o arco. Do mesmo modo, usando C, B e O_0 é encontrado o ponto E .

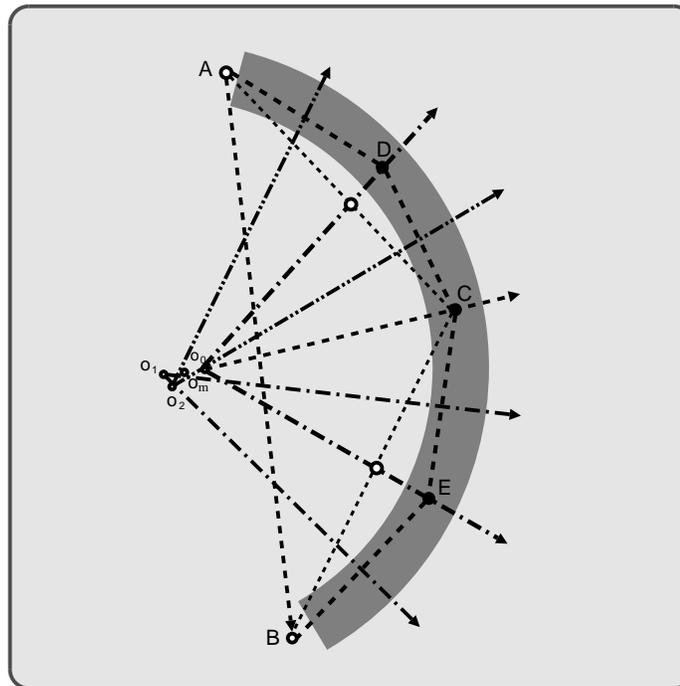


Figura 4.7: Esta figura mostra o processo iterativo para a determinação do centro do arco. Partindo dos três pontos iniciais do arco A, B e C são encontrados centros O_n para as bissetrizes de intersecção dos pontos do arco. Analisando a dispersão dos centros é finalmente encontrado o centro mais provável.

Os dois pontos triplos (A, C, D) e (B, C, E) são usados para encontrar as estimativas do centro O_1 e O_2 respectivamente. As diferenças de distância entre os centros O_0 ,

²A bissetriz é um segmento de recta que divide um ângulo em duas partes iguais.

O_1 e O_2 são verificadas e comparadas a um nível de dispersão de *threshold*. Se o centro estimado do potencial arco estiver muito afastado do centro de dispersão de *threshold*, então o arco candidato em estudo não representa um arco.

Validação do Arco

Após determinar os parâmetros óptimos do arco candidato, é realizada a validação para calcular a percentagem de *match* entre o arco estimado e o arco em estudo. Esta validação é implementada através do *match* entre os dois traçados, tal como o método descrito por Song et al. [113].

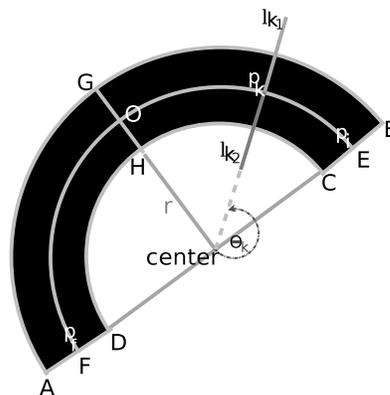


Figura 4.8: A figura mostra o traçado teórico do arco com os parâmetros previamente encontrados. Esta trajetória é utilizada para calcular a percentagem de *match* entre os dois arcos. Percorrendo o traçado do arco de p_i até p_f através de p_k e analisando o arco candidato sobre as retas l_{k1} e l_{k2} , são encontradas características do arco, nomeadamente a simetria central, interferência ao ruído e conexão com outros elementos. Finalmente, ocorre a decisão sobre a validade do arco candidato.

O algoritmo de Bresenham para os arcos é usado para encontrar todos os pontos da trajetória genérica do arco (x, y, r, p_i, p_f) , onde a variável p_k é um ponto na sua trajetória (ver figura 4.8). Para cada ponto p_k , pertencente à trajetória do arco, é calculado um segmento de recta l_k , com orientação θ_k e perpendicular à trajetória do arco nesse ponto.

Considerando que p_k é o ponto médio dessa recta, podemos encontrar dois segmentos de recta l_{k1} e l_{k2} , sobrepostos a l_k , em que d_k é a largura do arco sobre a recta com orientação θ_k . Em cada ponto do intervalo p_k , desde p_i e até p_f é encontrada a largura superior (d_{k1}) e inferior (d_{k2}) do arco, usando os segmentos de recta l_{k1} e l_{k2} . Este método foi inspirado na validação realizada por Song et al. [111] a segmentos de recta, considerando a sua linearidade e o seu preenchimento. Por outro lado, este método analisa algumas características do arco, tais como a simetria central, a interferência ao ruído e a conexão entre o arco e outros elementos. O arco é rejeitado se o módulo da diferença entre a largura superior e inferior, em qualquer parte do arco, for um valor maior do que o permitido, pois um arco não pode ter qualquer elemento conexo e deve apresentar simetria central.

4.4 Extracção de Linhas a Tracejado

As linhas a tracejado dividem um prédio em parcelas, separando as áreas do prédio consoante o tipo de terreno. Estas linhas não têm uma trajectória linear, tal como acontece com as linhas existentes no contorno dos prédios, porque são formadas simultaneamente por curvas e segmentos de recta. As duas principais dificuldades na extracção de linhas a tracejado são as discontinuidades periódicas e as junções, sendo estas últimas devido aos cruzamentos dos tracejados.

Para este tipo de detecção existem várias abordagens, por um lado, abordagem de análise a nível pixel, utilizando ferramentas de processamento de imagem, tais como os operadores morfológicos [2] ou por outro lado, a nível vectorial [28], depois de efectuar a conversão *raster-to-vector*. A implementação foi realizada directamente na imagem, através da utilização dos operadores morfológicos e de outras heurísticas, tal como su-

gerido por Dosch et al. [34].

Detecção do Tracejado Elementar

O tracejado elementar é definido como sendo o elemento básico das linhas a tracejado e apresenta uma forma rectangular completamente preenchida. Primeiramente, cada tracejado elementar é detectado; depois a sua posição é analisada, de modo a ligar todos os restantes e assim, finalmente encontrar as linhas a tracejado completas. A detecção através do seguimento da trajectória dos tracejados é um processo mais complexo do que apenas detectar os tracejados elementares das linhas a tracejado, ligar as extremidades de cada elemento e obter a linha completa. A sua complexidade reside no facto destas linhas apresentarem curvas e junções.

Os tracejados são elementos não conexos e são obtidos através do algoritmo de *component labelling* [37]. O primeiro problema a ser considerado é segmentar os tracejados dos outros elementos não conexos, tais como os caracteres. A segmentação é realizada considerando três condições particulares, designadamente o conhecimento de que, um tracejado tem um tamanho e uma forma característica, existe pelo menos um tracejado vizinho com características semelhantes à sua e um conjunto de tracejados segue uma linha virtual [34].

A primeira condição está baseada na análise ao tamanho e à forma de um tracejado elementar. Definitivamente, um tracejado tem a forma de um segmento de recta elementar com uma orientação arbitrária, deste modo, um tracejado é descrito usando um rectângulo com a mesma orientação. Contudo, para uma detecção invariante à rotação, é calculado o eixo de maior fluxo [65] de cada elemento, obtendo uma orientação ϕ , depois é aplicada a rotação $-\phi$ a cada elemento, de modo a que todos os tracejados tenham a orientação igual

ao seu eixo de fluxo máximo. Posteriormente, é aplicada uma dilatação isotrópica, para obter uma rotação normalizada.

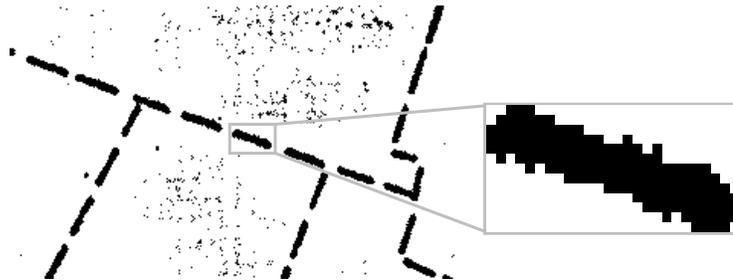


Figura 4.9: A detecção de linhas a tracejado está baseada em três factos, nomeadamente um tracejado tem um tamanho e forma característica, tem pelo menos um vizinho com uma orientação semelhante à sua e um conjunto de tracejados segue uma linha virtual.

Uma vez que o tracejado genérico tem uma forma rectangular completamente preenchida, os elementos não conexos são analisados considerando a sua densidade. Por exemplo, comparando o caracter '1' a um tracejado, observamos que ambos apresentam uma forma rectangular bastante parecida, considerando o cálculo da densidade do tracejado sobre um rectângulo (ver figura 4.10). No entanto, verificamos que o caracter apresenta uma forma rectangular não totalmente preenchido, enquanto que o tracejado apresenta-se completamente preenchido, desta forma é possível segmentar os elementos não conexos atendendo ao preenchimento da forma rectangular.

Segmentação de Tracejados Conexos ao Contorno

Uma vez que as linhas a tracejado estão definidas dentro do contorno dos prédios, é comum existirem tracejados sobrepostos a linhas contínuas. Para identificar duma forma completa as linhas a tracejado é necessário considerar todos os tracejados elementares, inclusive aqueles conexos às linhas contínuas.

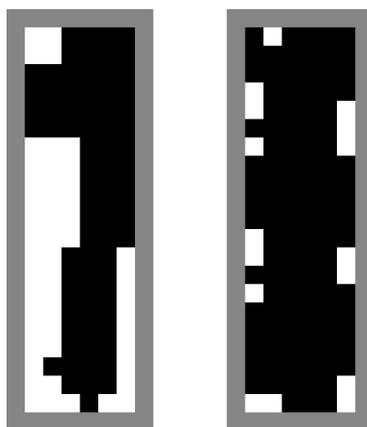


Figura 4.10: Esta figura mostra a distinção relativa à densidade entre o caracter '1' e um tracejado. Considerando que o tracejado apresenta um densidade próxima de um rectângulo completamente preenchido é assim possível detectar estes elementos de acordo com esta característica.

À partida, a procura por tracejados elementares sobre as linhas contínuas, pode ser realizado de duas formas distintas, por uma lado, extraíndo linhas a tracejado e procurando nos extremos dessas linhas por tracejados conexos ao contorno, como é normalmente realizado em problemas de segmentação de caracteres, ou por outro lado, realizando uma procura sobre o contorno numa perspectiva global, de modo a encontrar tracejados elementares.

A segunda abordagem parece ser a mais indicada nesta situação, porque no primeiro caso, para uma extracção exacta do tracejado conexo, seria sempre necessário extrair totalmente a linha a tracejado, que é a chave da procura. Como é recomendável implementar métodos de detecção com o mínimo de dependências possível, a escolha recai sobre o método da procura global no contorno por tracejados elementares.

As duas condições necessárias à procura por tracejados conexos são nomeadamente, utilizar a imagem esqueleto do contorno (com largura de linha igual a um) e partir de um ponto sobre o contorno. A procura por estes elementos é realizada seguindo as linhas do contorno numa perspectiva global e recursiva. Numa determinada posição do contorno,

um ponto pode ter o estado de ponto final ou de nó, dependendo da linha do contorno terminar ou ser um ponto de ramificação de outras linhas do contorno, respectivamente. O algoritmo de procura tem início num ponto da linha contínua de seguida segue todos os pontos do contorno até chegar aos pontos com apenas um vizinho (ponto final) ou com mais do que dois vizinhos (nó). Ao encontrar um nó, o algoritmo de seguimento do contorno vai guardar todos os pontos dos vizinhos numa fila de espera. À medida que vai seguindo o caminho, vai também marcando os pontos percorridos, de modo a não considerá-los numa segunda vez. Ao encontrar um ponto final é chegado o momento de análise à forma do tracejado elementar.

De uma forma simplificada, o algoritmo analisa os segmentos de linha, através dos seus extremos que podem ser nós ou pontos finais e detecta se esses segmentos são tracejados elementares. Por definição, um tracejado elementar dispõe duma dimensão reduzida comparativamente a outros segmentos de linha e tem pelo menos um tracejado elementar vizinho a uma distância menor do que o dobro do seu comprimento. É de considerar que o tracejado elementar isolado tem uma orientação similar ao tracejado elementar conexo.

Depois de detectar o tracejado elementar conexo, este é segmentado ao contorno e adicionado à lista de tracejados elementares isolados, de modo a ser considerado na ligação dos tracejados isolados.

Ligação entre Tracejados Isolados

O algoritmo de *range searching* [103] foi utilizado para implementar o método de procura do tracejado mais próximo, usando uma grelha de duas dimensões e colocando todos os tracejados elementares nessa estrutura de modo a otimizar a velocidade de procura. A

lista de tracejados isolados é percorrida, encontrando-se para cada tracejado elementar, uma lista ordenada dos tracejados vizinhos; este processo considera como critério de ordenação a distância euclidiana entre o tracejado elementar e o seu vizinho, de modo a obter a lista de tracejados vizinhos mais prováveis e as respectivas ligações.

Posteriormente, os tracejados são analisados de maneira a verificar se têm pelo menos um tracejado vizinho a uma distância elementar, menor do que o dobro do seu comprimento. Adicionalmente, a diferença entre a orientação do tracejado e do seu vizinho deve estar compreendida no intervalo $i: i \in [\phi - \theta, \phi + \theta]$, onde ϕ representa o eixo de maior fluxo do tracejado e θ o máximo ângulo de tolerância entre dois tracejados vizinhos ligados.

Tracejados Não-Lineares

Alguns tracejados elementares apresentam-se isolados e sem ligação (nas suas extremidades) a outros elementos de tracejado elementar. Ao analisar os elementos vizinhos mais prováveis para efectuar a ligação, são detectados elementos não conexos com duas ou mais extremidades representando tracejados com forma curva ou junções, respectivamente.

Embora possuam duas extremidades, os tracejados que apresentem uma forma curva e menos rectangular, não são detectados como tracejados elementares, não sendo assim ligados aos restantes elementos. As *junções* apresentam características não lineares e ocorrem quando várias linhas a tracejado se intersectam.

Em ambos os casos, as extremidades do tracejado não-linear são encontradas na imagem esqueleto do elemento, detectando os extremos das linhas que o constituem. Os extremos dos segmentos de recta que definem cada tracejado são alvos de ligação, tendo como objectivo encontrar as conexões mais prováveis entre os potenciais pontos de ligação.

Assim, ao detectar elementos não lineares entre dois elementos a tracejado vizinhos, em que o elemento não-linear apresenta extremidades alvo de ligação, são analisadas as ligações mais prováveis de modo a unir os elementos. Tal como acontece entre os elementos lineares, a ligação é validada atendendo à distância e orientação, depois de segmentada a linha de ligação da junção, entre a sua extremidade e o ponto central de ramificação do elemento (ver figura 4.11).

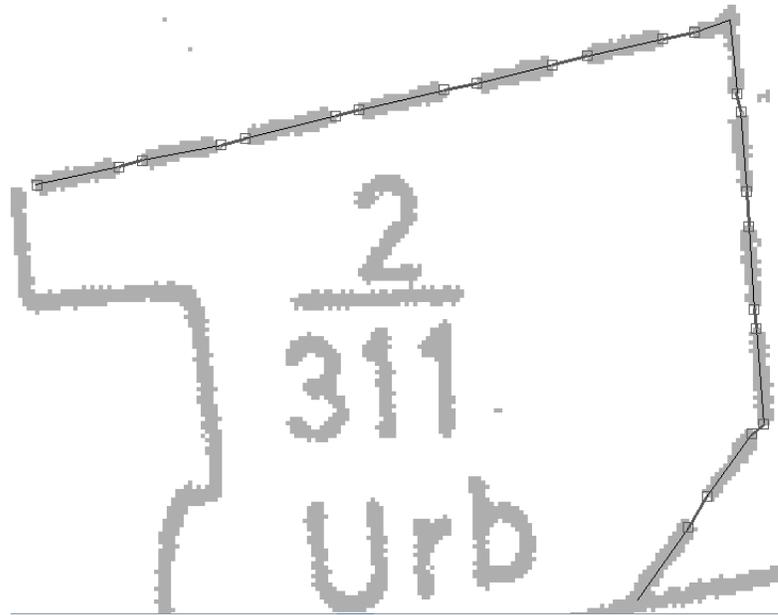


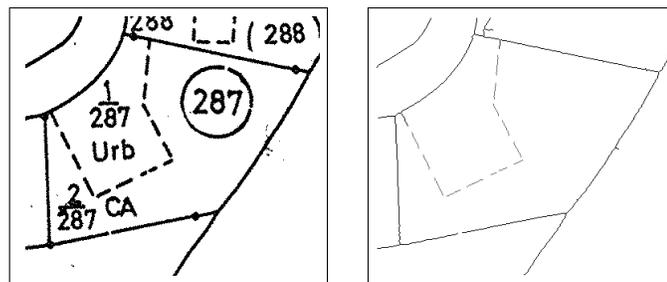
Figura 4.11: A ligação dos tracejados elementares às junções (tracejados elementares compostos) é realizada considerando a proximidade dos extremos dos tracejados elementares aos extremos das junções.

Validação e Vectorização das Linhas a Tracejado

Posteriormente à ligação dos tracejados elementares, é analisada a estrutura das linhas a tracejado virtuais. Nomeadamente, são verificadas as ligações entre os tracejados e se for encontrada a falta de um tracejado, a trajectória da linha entre dois potenciais vizinhos é se-

guida, procurando por elementos em posições intermédias e caso existam são adicionados à linha a tracejado. Os tracejados que previamente não são reconhecidos, como é o caso dos elementos que estão conexos a outros elementos ou dos elementos que não têm uma forma rectangular, são detectados como tracejados não-lineares, segmentados e ligados aos demais tracejados elementares detectados, considerando a proximidade da distância euclidiana e a diferença de orientação entre os dois elementos, tal como descrito anteriormente.

Após o seguimento da linha virtual, usando os tracejados elementares reconhecidos, é feita a ligação das extremidades entre vizinhos de modo a obter linhas contínuas (ver figura 4.12-b e 4.13-a). Finalmente, as linhas contínuas são vectorizadas através do método de Rosin-West [123, 124].



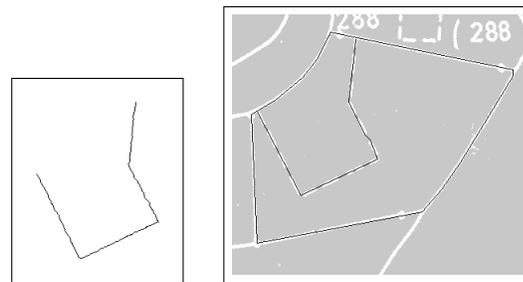
(a) Imagem inicial.

(b) Imagem emagrecida do contorno e tracejado.

Figura 4.12: Imagens do pré-processamento da extracção das linhas a tracejado.

4.5 Detecção dos Marcos

As extremas de prédio são em geral segmentos de recta contínuos que unem marcos de propriedade, exceptuando os casos em que estes seguem elementos naturais do terreno,



(a) Ligação das extremidades da linha a tracejado.
(b) Vectorização das linhas utilizando o método de Rosin-West.

Figura 4.13: Validação e vectorização das linhas a tracejado.

onde não seria viável a colocação de marcos, como seja o eixo de um rio. O conjunto das estremas e marcos definem o limite das propriedades. Nos mapas cadastrais os marcos de terreno estão representados por pequenos círculos completamente preenchidos posicionados sobre o contorno dos prédios. Os marcos de propriedade unem os segmentos de recta que constituem as estremas de prédio.

De um ponto de vista da percepção visual os marcos salientam-se dos limites da propriedade por aumentarem a largura do contorno de forma circular num determinado ponto. Em regra geral, os pequenos círculos sobre o contorno apresentam-se bem diferenciados e podem ser identificados de modo automático.

Na literatura é conhecido que o processo de convulsão pode evidenciar algumas características das imagens, tais como detectar as linhas horizontais, verticais ou oblíquas [134]. O processo de convulsão utiliza uma matriz, designada por *kernel* para calcular o resultado do processo num dado ponto da imagem. O *kernel* escolhido permite evidenciar determi-

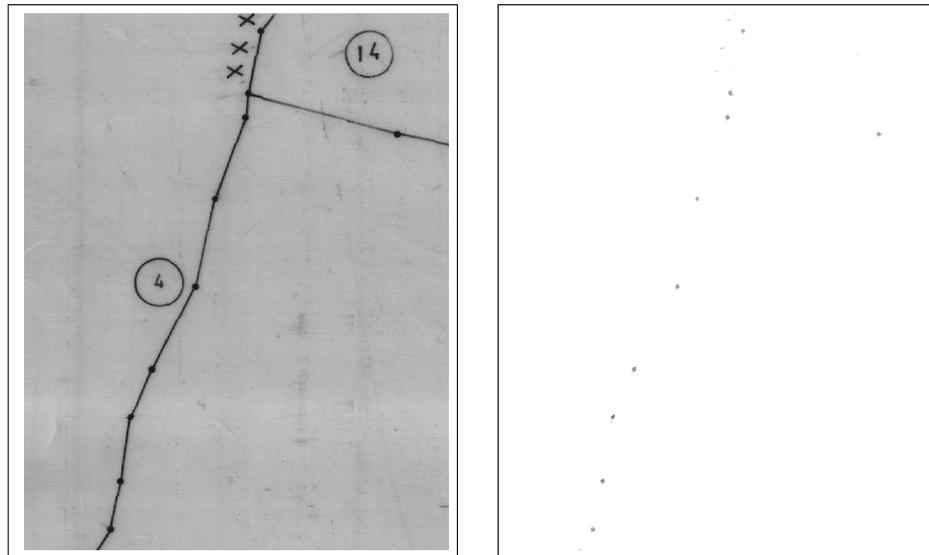
nadas características da imagem funcionando como uma filtro. Na literatura são bastantes conhecidos os filtros de *Sobel*, *Laplace*, *blur*, entre outros [43].

O *kernel* foi determinado experimentalmente considerando a noção de vizinhança, ou seja, a contribuição dos pixels vizinhos para o resultado final num determinado ponto. Considerando que os pequenos círculos estão preenchidos é possível segmenta-los das linhas contínuas de modo a detecta-los facilmente. O *kernel* definido pela matriz K , é utilizado para detectar os pequenos círculos sobre o contorno e apresenta a dimensão 5×5 .

$$K = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.8)$$

Através do processo de convulsão e utilizando o *kernel* atrás definido, o contorno é emagrecido evidenciando-se as zonas com uma largura de linha mais acentuada e consequentemente é detectado os marcos sobre o contorno (ver figura 4.14). De notar que as zonas com os marcos estão evidenciadas enquanto as restantes zonas são removidas.

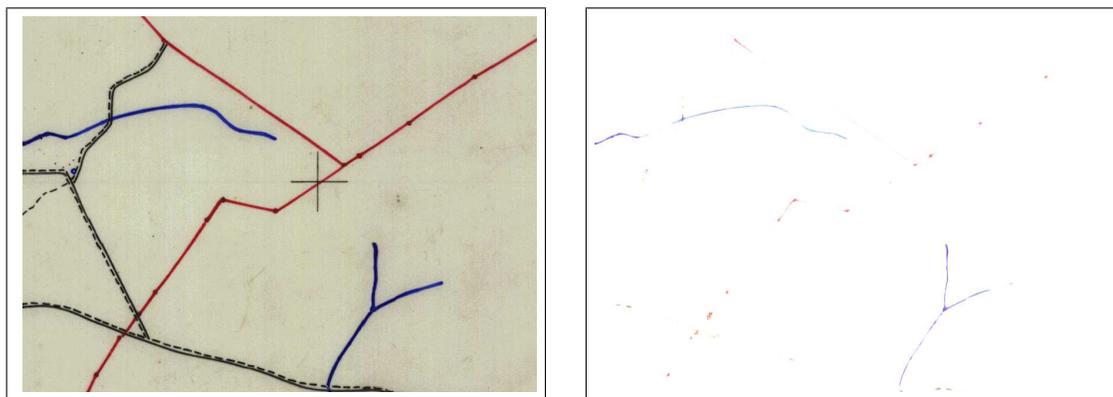
Considerando também os mapas coloridos (ver figura 4.15), pode observar-se que o resultado da convulsão com o *kernel* K evidencia do mesmo modo os marcos sobre o contorno, permitindo encontrar as zonas com elevada probabilidade de existência de marcos. Depois da convulsão, os picos de cor vermelha são encontrados e determinados os marcos sobre o contorno.



(a) Imagem inicial contendo os marcos sobre o contorno.

(b) Imagem do resultado da convulsão que implementa a detecção dos marcos sobre o contorno.

Figura 4.14: A detecção dos marcos em imagens a cinza é implementada através do processo da convulsão utilizando o *kernel K*.



(a) Imagem colorida contendo os marcos sobre o contorno com a cor vermelha.

(b) Imagem do resultado da convulsão que implementa a detecção dos marcos sobre o contorno.

Figura 4.15: A detecção dos marcos em imagens coloridas é implementada através do processo da convulsão utilizando o *kernel K*.

4.6 Conclusões

Para realizar o reconhecimento de imagem foram implementados vários métodos de análise, que permitem extrair informação das secções cadastrais. A detecção dos objectos gráficos foi implementada através de métodos de análise baseados nas características intrínsecas desses elementos. À excepção dos marcos, esta implementação foi realizada ao nível pixel, procurando pelas formas características dos objectos gráficos e decidindo sobre a validação através de uma análise probabilística. Os métodos de análise encontrados são simples, estáveis e robustos, podendo ser utilizados para realizar a extração de informação neste tipo de imagem.

Capítulo 5

Reconhecimento de Caracteres

A informação cadastral é representada utilizando primitivas gráficas de desenho e caracteres. Contudo, estas duas camadas de informação apresentam-se sobrepostas e para reconhecer cada uma delas é necessário realizar previamente uma separação.

Estes mapas são constituídos por uma elevada quantidade de caracteres que representam palavras, no entanto nem toda essa informação é útil, como é o caso do cabeçalho e da legenda situada no limite da quadricula do desenho. Deste modo, o problema do reconhecimento de caracteres fica simplificado considerando apenas como informação necessária, os identificadores de prédio e do tipo da parcela, caracteres estes que representam mais de 90% de todo o seu conjunto.

O reconhecimento de caracteres implementado detecta os elementos não conexos, classificando isoladamente cada caracter, usando para tal um método de reconhecimento que agrupa conjuntos de caracteres vizinhos em palavras e valida a semântica da palavra obtida com a ajuda de um dicionário. Seguidamente, são apresentados pormenorizadamente os métodos de detecção e reconhecimento de caracteres utilizados nos mapas cadastrais

Portugueses.

5.1 Segmentação Texto/Gráficos

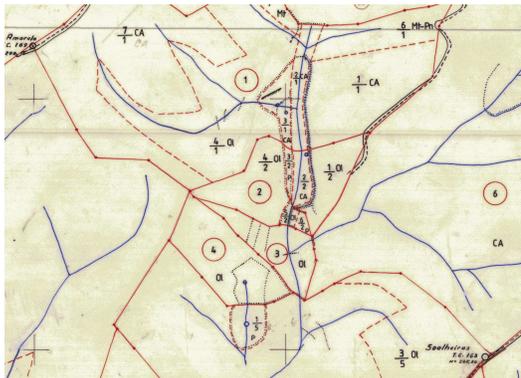
Os mapas cadastrais são documentos que apresentam informação composta por variados elementos, considerando os tamanhos, formas, áreas e a conectividade com outros elementos. Os mapas cadastrais são na sua maioria monocromáticos, contudo também existe uma quantidade considerável a cor.

5.1.1 Mapas Coloridos

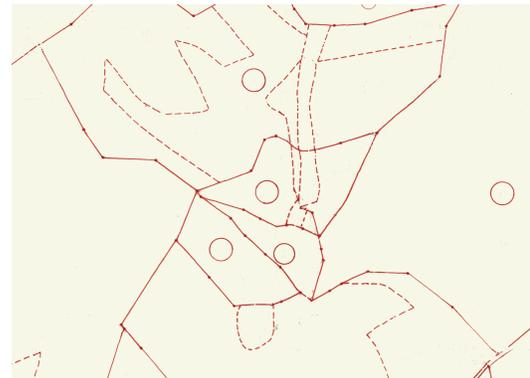
Quando os mapas apresentam cor, os elementos gráficos são facilmente segmentados considerando o intervalo de cor pelo quais são formados, realizando uma segmentação baseada na cor dos elementos [68, 96, 75], tal como é realizado para outro tipo de mapas, como por exemplo, os mapas cartográficos [70] ou os mapas topográficos [117].

A figura 5.1 mostra a segmentação baseada na cor dos elementos gráficos, nomeadamente dos elementos cadastrais a vermelho, dos elementos alfanuméricos a preto e das linhas de água a azul. Partindo da figura 5.1(a), os elementos a vermelho são segmentados considerando todos os elementos com uma tonalidade próxima da cor *RGB* vermelho padrão¹ (ver figura 5.1(b)). A distância entre as cores é definida pela soma total da diferença de cada componentes *RGB*, relativamente à cor padrão segmentada. Do mesmo modo são obtidos os elementos a preto (ver figura 5.1(b)) e a azul (ver figura 5.1(b)).

¹A cor *RGB* do vermelho é definida pelo triplo (255,0,0)



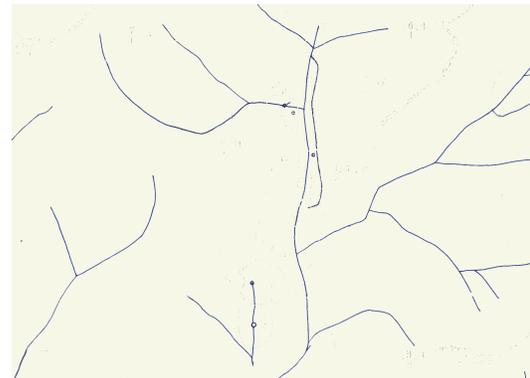
(a) Folha cadastral (0511030.au) com os elementos gráficos de várias cores.



(b) Segmentação dos elementos cadastrais com a cor vermelha.



(c) Segmentação dos elementos alfanuméricos com a cor preta.



(d) Segmentação das linhas de água com a cor azul.

Figura 5.1: Imagens representativas da segmentação dos elementos gráficos baseado na cor.

5.1.2 Mapas Monocromáticos

No caso dos documentos monocromáticos, a segmentação texto/gráfico considera as características morfológicas dos elementos gráficos, tal como sugerido por Tombre et al. [126].

Para realizar a segmentação baseada na morfologia é necessário considerar os vários elementos que constituem cada camada, de modo a realizar algoritmos de segmentação baseados nas características específicas desses elementos. Por um lado, a camada que

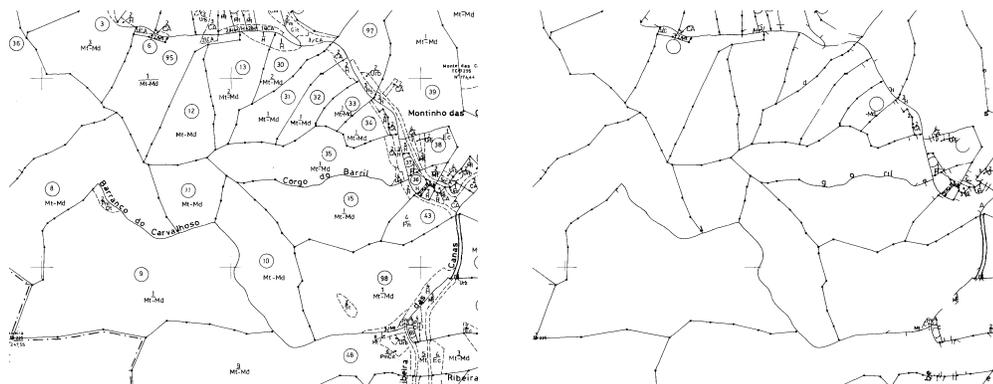
contém a informação gráfica é constituída por cruces, círculos, arcos, linhas contínuas e linhas a tracejado. Por outro lado, a camada com a informação alfanumérica apresenta conjuntos de caracteres devidamente alinhados, que representam palavras, tais como escala, coordenadas, referências, número de prédios que sofreram modificações, identificação do número de prédio/parcela e do tipo de parcela.

Os elementos gráficos, nomeadamente as cruces, os arcos e as linhas contínuas apresentam-se conexos entre si, formando conjuntos de pixeis ligados. Porém, os caracteres, os círculos e os elementos das linhas a tracejado (tracejados elementares) não se encontram conexos, apresentando-se de forma isolada e bem definida. Deste modo, podemos separar os dois conjuntos de informação considerando o valor da área que estes elementos ocupam na imagem (ver figura 5.2). Esta segmentação é realizada usando um algoritmo de *component labeling* e posteriormente um filtro relativo à área que os elementos ocupam, utilizando os valores do rectângulo que define o seu limite espacial (*bounding box*). Como os outros elementos que poderiam pertencer à camada dos elementos não conexos já foram previamente segmentados na detecção de círculos e de tracejados, apenas fazem parte desta camada os caracteres não conexos (ver figura 5.2(c)).

5.2 Métodos de Análise

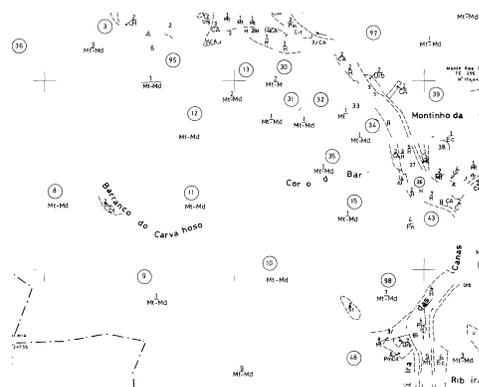
O reconhecimento de caracteres é um assunto bem estudado, existindo na literatura vários algoritmos com diferentes abordagens. Cordella et al. [19] apresentaram um artigo que revisita o conjunto de técnicas para o reconhecimento de símbolos em documentos.

Trier et al. [128, 129] estudaram o reconhecimento de caracteres isolados descrevendo vários métodos baseados na extracção de características, nomeadamente os métodos de



(a) Amostra de uma imagem monocromática.

(b) Segmentação dos elementos conexos.



(c) Segmentação dos elementos não conexos.

Figura 5.2: Imagens representativas da segmentação de uma imagem monocromática baseada na área e conexão dos elementos constituintes da imagem.

template matching, *templates* deformáveis, zoning, momentos geométricos invariantes, momentos de Zernik, descritores de Fourier, redes neuronais, etc. Cada um destes métodos apresenta características diferentes relativamente às invariâncias, tais como o tamanho, a rotação, a análise local, a análise global e o método de aprendizagem.

Os métodos de reconhecimento de caracteres podem realizar, análises de três tipos, baseadas no esqueleto, no contorno ou considerando todos os pontos do caracter. A análise baseada no esqueleto considera a forma e ignora os principais detalhes do caracter, transforma primeiramente o caracter no seu esqueleto e realiza posteriormente a análise baseada

na comparação das suas linhas. O método que realiza a análise baseada no contorno estuda a linha exterior que representa o caracter, usando assim como medida de comparação a forma do caracter. Enquanto que, a análise baseada em todos os pontos do caracter, estuda os seus detalhes e utiliza-os para reconhecer caracteres similares em maior detalhe.

5.2.1 Método de *Template-Matching*

A primeira abordagem ao problema do reconhecimento de caracteres nos mapas cadastrais foi a implementação do algoritmo dos *templates* deformáveis [40], uma vez que os caracteres existentes nestes mapas são muito semelhantes entre si e estão por vezes conexos. Este método foi implementado definindo os *templates* e os pontos de *match*, de modo a classificar a taxa de semelhança entre dois caracteres.

Os *templates* são modelos representados através de uma imagem que caracteriza univocamente cada caracter, sendo, os pontos do *template*, conjuntos de pontos representativos do caracter e são utilizados para calcular a taxa de *match* (ver figura 5.3). Os pontos de *match* podem ser definidos por dois modos distintos, por um conjunto de pixels interiores do caracter (esqueleto do caracter) ou por um conjunto de pixels exterior ao caracter (*outline* do caracter). A utilização destes dois tipos de pixels está relacionado com a contribuição positiva ou negativa dos pontos do *template* durante o *match* entre o *template* e o caracter em estudo. Especificamente se os pontos de *match* internos ou externos tiverem o mesmo valor (branco ou preto) então aumenta a taxa do *match*, caso contrário esse valor diminui. Os pontos de *match* internos são encontrados utilizando o esqueleto do caracter e mais a condição de que dois pontos vizinhos do esqueleto têm a distância mínima de três pixels. Os pontos de *match* externos são encontrados realizando primeiramente uma

dilatação à imagem do *template* e depois posteriormente uma subtração à imagem do *template* original na posição central, de modo a obter o outline do *template*, considerando a condição de que dois pontos vizinhos no outline têm a distância mínima de três pixels. Finalmente, os caracteres são comparados usando os pontos do *template*, com o objectivo de obter o valor de *match* entre ambos.

As figuras 5.3(a) e 5.3(b) mostram o *template* dos caracteres '1' e 't' respectivamente, mais os pontos de *match* sobre cada *template*. O método de *template matching* calcula a taxa de *match* entre o *template* e o caracter em estudo, comparando na posição dos pontos de *match*, os valores dos pixels. A pontuação de *match* aumenta caso o valor do pixel seja igual, e diminui em caso contrário. Deste modo, a taxa de *matching* final é calculada analisando a semelhança entre os valores dos pixels em cada ponto de *match*.

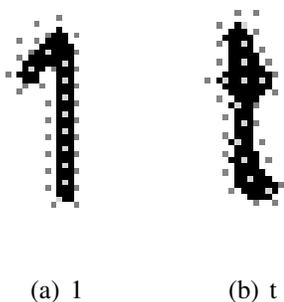


Figura 5.3: A determinação dos pontos de *match* no método de *template matching* considera os pontos de esqueleto do caracter e os pontos do *outline* do caracter, mais a condição de que dois pontos de *match* devem estar a uma distância mínima de três pixels.

Por forma a distinguir uma enorme variedade de caracteres considerando apenas os pontos do *template*, é muitas vezes necessário associar um peso a cada ponto, de modo a otimizar o método de reconhecimento. Apesar de considerar os pesos nos pontos de *match*, este processo revelou-se pouco satisfatório, dado ter pouca tolerância ao ruído.

5.2.2 Método de Zoning

O método de Zoning [26] foi implementado por ser um método robusto a altas frequências (filtro passa baixo) e por ter um bom nível de tolerância ao ruído e aos detalhes do carácter. Este método, em vez de analisar o carácter ao nível pixel, tal como acontece com o método de *template matching*, realiza uma análise nas zonas similares entre o *template* e o carácter em estudo.

Em cada zona é efectuada uma comparação contando o número de pixeis a preto e consoante a sua percentagem de pixeis a preto é atribuído a cada um um valor cinza. Na avaliação final são consideradas todas as zonas similares entre o carácter e o *template*. A análise é realizada primeiro numa perspectiva local, na mesma zona entre o *template* e o carácter em estudo e depois é obtida uma comparação global, considerando o conjunto das zonas comparadas.

Os *templates* estão numa base de dados que contêm todos os caracteres possíveis e alguns caracteres têm até mais do que um *template*. Como os caracteres desconhecidos são comparados com todos os modelos da base de dados para obter uma taxa de semelhança para cada zona, a taxa de *match* aumenta se ambas as zonas forem similares (i.e. zonas com um valor próximo de pixeis a preto). Um carácter é reconhecido se a maior percentagem de *match* for superior a 50%, caso contrário é classificado como desconhecido.

A figura 5.4 mostra um exemplo da análise realizada pelo método de Zoning. Mais uma vez reforça-se o facto da análise em zonas ter uma elevada tolerância aos detalhes do carácter e ao ruído, sendo estes os problemas principais problemas do método de *template matching*.

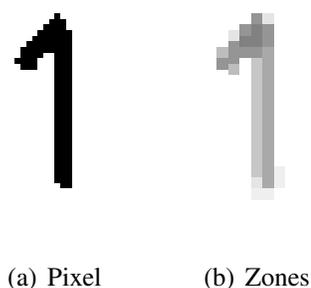


Figura 5.4: Exemplo da análise realizada pelo método de Zoning.

5.2.3 Métodos de Aprendizagem (K-NN)

Os caracteres são os elementos mais sensíveis no reconhecimento dos mapas cadastrais, dado existir uma elevada quantidade de mapas cadastrais elaborados em diferentes períodos de tempo. Ao longo de cinco décadas, os caracteres nos mapas foram sendo desenhados utilizando diferentes escantilhões e desta forma, foi aumentando significativamente a variedade da representação deste tipo de informação. Num mapa cadastral todos os caracteres são desenhados com a mesma régua, no entanto, de documento para documento surgem diferenças significativas relativas ao tamanho do carácter, morfologia, largura de traço, ruído, rotação, etc. Assim, a implementação de um método que realize o reconhecimento de caracteres de uma forma invariante e automática em todos os mapas, aparece como uma tarefa impraticável e não muito promissora.

Os caracteres apresentam diferenças morfológicas de secção para secção cadastral, no entanto no mesmo mapa os caracteres apresentam a mesma morfologia. Assim, é necessário seleccionar os *templates* representativos de cada classe nos diferentes mapas, de modo a estabelecer comparações de todos os caracteres existentes com os *templates*, com vista ao seu reconhecimento.

Nos sistemas de reconhecimento de caracteres é usual realizar-se o treino do sistema com o objectivo de construir funções de reconhecimento e assim, evitar comparações sucessivas com os *templates*. No entanto, como o treino demora algum tempo e sendo necessário realizar esta operação em cada mapa, surge assim a necessidade de utilizar-se métodos de aprendizagem.

Com o objectivo de treinar o sistema para reconhecer os caracteres existentes nos mapas cadastrais foi utilizado o software *Gamera* [80, 35, 49]. Este software têm vindo a ganhar popularidade nos últimos anos e actualmente apresenta-se como uma ferramenta estável e de grande utilidade para realizar o reconhecimento de caracteres. Esta ferramenta é constituída por um interface que permite treinar iterativamente o sistema de reconhecimento, utilizando variadas *features* para descrever cada caracter.

Inicialmente é aberta uma imagem e todos os elementos não conexos são carregados para uma tabela de modo a serem seguidamente classificados (ver figura 5.5).

De seguida são seleccionados e classificados alguns caracteres com o objectivo de construir-se uma base de dados de caracteres reconhecidos (ver figura 5.6), obtendo-se uma base de conhecimento. Assim, são classificados manualmente vários *templates* representativos de cada caracter, de modo a utilizar-se posteriormente este conhecimento para realizar o reconhecimento automático. Nota-se que a tabela aparece de cor verde, após a classificação manual dos caracteres e de cor branca, como mostra a figura 5.5 enquanto os símbolos não ficarem classificados.

Depois da classificação manual são obtidos vários *templates* com uma semântica conhecida, podendo esta informação ser utilizada para treinar o sistema e implementar funções de reconhecimento. Para realizar o treino é necessário seleccionar quais as *features* que

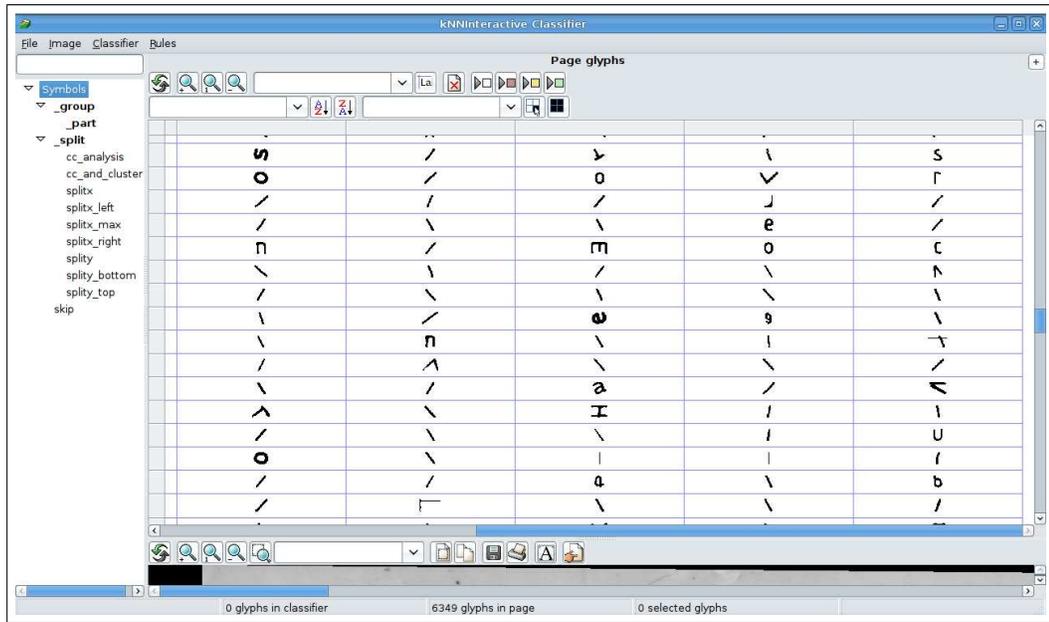


Figura 5.5: Primeiramente os elementos não conexos são identificados com vista à sua classificação.

Character	Template 1	Template 2	Template 3	Template 4	Template 5	Template 6	Template 7	Template 8
latin.small.letter.0	0	0	0	0				
latin.small.letter.1	1	1	1	1	1	1		
latin.small.letter.2	2	2	2	2	2	2	2	2
latin.small.letter.3	3	3	3	3	3			
latin.small.letter.4	4	4	4	4	4			
latin.small.letter.5	5	5	5					
latin.small.letter.6	6							
latin.small.letter.7	7	7	7					
latin.small.letter.8	8							
latin.small.letter.9	9	9	9					
latin.small.letter.a	a	a	a	a	a	a	a	a
latin.small.letter.b	b	b						
latin.small.letter.c	c	c						
latin.small.letter.d	d							
latin.small.letter.e	e							
latin.small.letter.l	l							

Figura 5.6: Os caracteres são classificados manualmente, construindo-se uma base de dados de *templates* reconhecidos. De notar que cada caracter pode apresentar vários *templates* de modo a obter-se maior robustez.

devem ser utilizadas (ver figura 5.7) de modo a distinguir as classes de elementos.

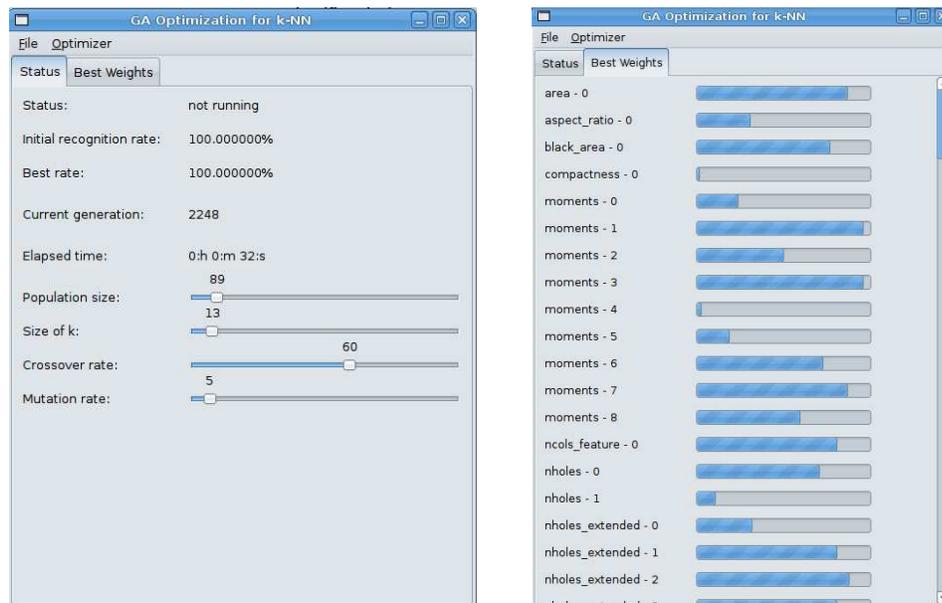


Figura 5.7: Esta imagem apresenta uma lista de features que são utilizadas para classificar os símbolos.

O sistema realiza a aprendizagem utilizando um classificador K-NN [21] em conjunto com um algoritmo genético [48] para otimizar os pesos das *features*. A classificação dos símbolos utilizando o algoritmo K-NN é um processo simples envolvendo a computação de uma descrição matemática da imagem, designada vector de *features*. Este processo calcula a distância entre os vectores de *features* de um símbolo desconhecido e as entradas na base de dados dos vectores de *features* conhecidos. A classificação do vector de *features* k mais próximo é dada ao vector de *features* desconhecido. A aprendizagem do sistema faz-se pela adição de entradas na base de dados dos vectores de *features*, que é realizada pelo utilizador (ver figura 5.8(a)).

A performance do classificador K-NN pode ser consideravelmente otimizado através da utilização de pesos nas *features* (ver figura 5.8(b)). Isto envolve a multiplicação da

distância sobre cada dimensão da *feature* por um número ou peso. A determinação do peso óptimo da *feature* pode ser realizado através de uma abordagem de força bruta, mas isso despenderia de muito tempo de computação. Assim o *Gamera* utiliza um algoritmo genético que determina os pesos óptimos dos vectores num modo *offline*, para uma maior eficiência computacional.



(a) Esta imagem apresenta as variáveis relativas ao treino do sistema de reconhecimento utilizando o classificador K-NN com otimização do algoritmo genético.

(b) No decorrer do treino os pesos das *features* são otimizados utilizando o algoritmo genético, de modo a implementar uma função de reconhecimento óptima.

Figura 5.8: O treino do sistema de reconhecimento de caracteres é realizado utilizando o classificador K-NN em conjunto com algoritmo genético para otimizar os pesos das *features*.

Depois de treinado, o sistema pode classificar automaticamente os símbolos que ainda não têm classificação. Na figura 5.9 pode observar-se os símbolos classificados de maneira automática, de notar que estes apresentam-se na tabela com a cor rosa, enquanto os

templates classificados manualmente possuem cor verde.

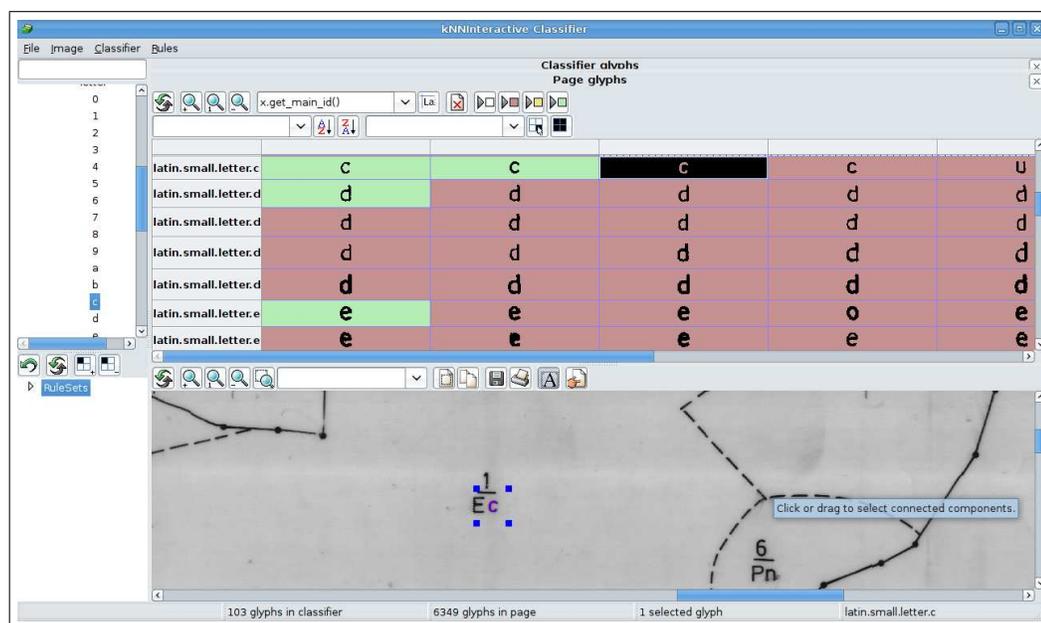


Figura 5.9: Esta figura apresenta o resultado da classificação automática. Na tabela da figura estão agrupados os símbolos em classes, apresentando-se a verde os símbolos classificados manualmente e a rosa os símbolos classificados automaticamente.

Finalmente é gravado os resultados em ficheiros no formato XML, que contém todos os dados do sistema de reconhecimento de modo estruturado, nomeadamente os dados relativos ao treino do sistema, à semântica dos *templates* e ao resultado da classificação automática. Na figura 5.10 pode observar-se o ficheiro *page_glyphs.xml* que apresenta todos os dados de cada símbolo classificado automaticamente, desde a sua semântica até à descrição de cada *feature* da sua imagem.

5.3 Segmentação de Caracteres Conexos

Os caracteres não reconhecidos são analisados relativamente ao seu tamanho e à sua largura e posteriormente comparados com as dimensões dos outros elementos reconheci-

```

<glyph uly="1535" ulx="1004" nrows="19" ncols="15">
  <ids state="AUTOMATIC">
    <id name="latin.small.letter.e" confidence="0.155085"/>
  </ids>
  <datas>
    5 6 8 7 7 9 5 12 3 4 4 4 3 3 6 4 1 4 3 2 1 4 1 29 1 14 1 5 4 4 2 4 11
    4 12 4 11 8 2 3 3 12 3 12 5 10 10 2 3 0
  </datas>
  <features scaling="1.0">
    <feature name="area">
      285.0
    </feature>
    <feature name="aspect_ratio">
      0.789473684211
    </feature>
    <feature name="black_area">
      179.0
    </feature>
    <feature name="compactness">
      0.482352941176
    </feature>
    <feature name="moments">
      0.293801169591 0.276638965836 0.0779167247142 0.120927076077
      0.057855188911 0.0133857371684 0.00729720284688 0.00386692350719
      -0.02931131487
    </feature>
    <feature name="ncols_feature">
      15.0
    </feature>
    <feature name="nholes">
      1.06666666667 0.315789473684
  </features>
</glyph>

```

Figura 5.10: Como resultado final é obtido um ficheiro XML com as classificações de cada símbolo, a sua probabilidade, bem como todos os valores do vector de *features* de cada um.

dos. No caso da largura ou altura apresentar uma diferença considerável relativamente aos símbolos reconhecidos então estes são analisados estudando a segmentação dos caracteres conexos.



Figura 5.11: Esta figura apresenta o histograma vertical e horizontal dos caracteres conexos 'CA'. A análise é feita em ambos os eixos de modo a encontrar as linha potenciais de segmentação. Este procedimento é apenas aplicável nos casos em que os elementos apresentam na zona de conexão uma fraca densidade de pixels.

Considerando o histograma vertical e horizontal do símbolo são encontradas as linhas potenciais de segmentação (ver figura 5.11), que possibilitam segmentar os caracteres conexos. Quando os elementos na sua zona de conexão, têm uma fraca densidade de pixels pretos então os caracteres podem ser segmentados. Porém se a conexão for muito densa e se os dois caracteres apresentarem tamanhos distintos então a detecção das linhas de segmentação não é um processo tão linear.

Segundo Song et al. [109], após segmentar os caracteres utilizando uma linha onde existe um mínimo global, ambos os caracteres são validados. No caso de ambos os caracteres serem reconhecidos e cada um apresentar uma taxa de reconhecimento aceitável, então ambos os caracteres são devidamente reconhecidos. Ao utilizar esta abordagem, Song et al. apresentam um sistema classificador de caracteres com *feedback* que decide sobre a linha de segmentação.

No entanto, nos casos problemáticos a linha de segmentação não separa convenientemente o símbolo, perdendo-se informação relativa aos caracteres. Nestes casos pode-se considerar as conexões como um só símbolo, por exemplo é comum o conjunto de caracteres 'CA' apresentar-se conexo. Assim, pode-se considerar *templates* com a conexão ('CA') de modo, a classificar directamente este tipo de símbolos.

5.4 Agrupamento de Caracteres

O agrupamento dos caracteres vizinhos é realizado com o objectivo de validar a semântica das palavras reconhecidas, verificando se estas compostas por conjuntos dos caracteres, existem num dicionário. Inicialmente, todos os caracteres são reconhecidos de forma isolada e sem a noção de vizinhança, no entanto nesta fase os caracteres são agrupados em

palavras, considerando que são elementos não conexos e que podem ser descritos considerando a sua posição, através de um rectângulo que define o seu limite.



(a) Antes do agrupamento dos caracteres.

(b) Depois do agrupamento dos caracteres.

Figura 5.12: O algoritmo de agrupamento de caracteres considera o rectângulo que define os limites de cada caracter. Considerando ambos os eixos de forma independente, são agrupados os elementos que apresentam uma distância euclidiana próxima e uma orientação semelhante entre dois caracteres. No final ambos os caracteres são agrupados e é definido um novo rectângulo que os engloba no seu todo.

O algoritmo desenvolvido agrupa caracteres vizinhos formando palavras, considera a distância euclidiana e a orientação dos caracteres. Deste modo, partindo da caixa que descreve os limites do componente, são procurados os elementos com uma distância euclidiana inferior à largura máxima ou à altura máxima do caracter, na lista que contém todos rectângulos que descrevem os limites dos caracteres. Se o caracter em estudo respeitar a condição prévia de proximidade e adicionalmente de orientação uniforme, i.e. um *offset* ligeiro entre os dois rectângulos, então os caracteres são agrupados e é calculado o novo limite, que se substância num novo rectângulo (ver figura 5.12).

O agrupamento é realizado para cada eixo e de forma independente, uma vez que as palavras estão alinhadas em todas as direcções. Este alinhamento considera a altura ou a largura do caracter consoante o eixo de agrupamento. No final do agrupamento são encontrados os rectângulos que englobam os caracteres vizinhos, obtendo-se conjuntos de caracteres que formam palavras e que podem posteriormente ser objecto de verificação

semântica.

5.5 Correção Semântica

Tal como referido, no início do presente capítulo, relativamente aos caracteres, os mapas cadastrais possuem apenas dois tipos de palavras enquanto informação útil para o seu reconhecimento, designadamente o identificador do prédio e parcela e o tipo da parcela. Estes dois tipos de palavras têm constituições diferentes, mais especificamente o identificador de prédio é constituído apenas por números enquanto que o identificador de parcela é constituído por uma fracção onde o numerador é formado por um número e denominador corresponde a um número de prédio idêntico ao número de prédio. O tipo de parcela é formado apenas por letras que correspondem a um determinado tipo de terreno.

A palavra com o número do prédio é validada verificando se todos os caracteres são formados restritivamente por números. No caso de prédios formados por parcelas é adicionalmente realizada uma validação considerando o facto de que, a palavra dentro do círculo é igual à palavra que está no denominador do número das parcelas. Caso não seja igual é calculado o número mais provável e é decidido sobre a palavra a considerar. Para os prédios sem parcelas, se a palavra não for formada apenas por números então é assinalada para a intervenção do utilizador.

Para validar o tipo de terreno é utilizado um dicionário que corrige os erros de reconhecimento dos caracteres, através da verificação da validade da palavra reconhecida. Como todas as palavras existentes no dicionário são válidas, as palavras reconhecidas são comparadas com as palavras do dicionário, calculando a sua distância através do algoritmo de Levenshtein [71]. Se a palavra reconhecida existir no dicionário então considera-se a pa-

lavra válida, caso contrario é encontrada uma lista de palavras no dicionário com a menor distância relativamente à palavra reconhecida. Como cada caracter tem uma taxa de *match* relativamente a todos os caracteres possíveis, então é calculada a probabilidade de cada palavra candidata². Assim, através do cálculo da probabilidade de cada caracter é encontrada a taxa de probabilidade de cada palavra do dicionário e finalmente é encontrada a palavra mais provável.

5.6 Conclusões

No reconhecimento de caracteres existem factores que facilitam e outros que dificultam o reconhecimento. Especificamente, o reconhecimento está facilitado porque todos os caracteres foram desenhados utilizando moldes, nomeadamente régua em forma de escantilhão, assim a semelhança entre os caracteres é maior e existem menos erros. Adicionalmente, os caracteres têm o mesmo tamanho e quase não apresentam rotação, deste modo, dois caracteres conexos podem ser separados utilizando histogramas horizontais e verticais. Contudo, a existência de ruído nos mapas origina a conexão de caracteres com outros elementos, tais como o contorno, as linhas a tracejado ou outros caracteres. Para ultrapassar este problema, o reconhecimento dos caracteres é realizado individualmente, sendo efectuada uma primeira segmentação, de maneira a separar e isolar cada caracter, de modo a possibilitar o seu correcto reconhecimento.

Cada método de reconhecimento apresenta características particulares e a sua utilização deve ser considerada consoante os casos em particular. Porém, o método de aprendizagem

²De notar que a palavra candidata é uma palavra que está no dicionário e tem uma distância mínima relativamente à palavra primeiramente reconhecida.

parece o mais adequado e rápido neste estudo uma vez que os caracteres apresentam variabilidade na sua morfologia.

O agrupamento dos caracteres e a sua análise semântica permite finalmente realizar a validação dos caracteres reconhecidos.

Capítulo 6

Extracção do Contorno

6.1 Introdução

A vectorização automática das secções cadastrais do CGPR ¹ apresenta várias dificuldades, que podem ser consideradas como interessantes desafios na área do reconhecimento de imagem. A extracção automática dos contornos é certamente um desses desafios, devido à sua complexidade, ao interesse do tema na literatura e ao seu carácter transversal. A maioria das dificuldades encontradas na extracção do contorno prendem-se com a degradação das secções cadastrais e com o modo como estas foram desenhadas, aumentando consideravelmente os problemas de análise, como de seguida poder-se-á verificar.

Do ponto de vista gráfico, as extremas de prédio são definidas, em regra geral, por segmentos de recta que unem os marcos de propriedade, contudo em casos que estas seguem elementos naturais do terreno onde, não seria viável a colocação de marcos, como seja em eixo de um rio ou um penhasco, as extremas apresentam curvaturas. À linha contínua

¹Cadastro Geométrico da Propriedade Rústica

que delimita a propriedade, sendo esta graficamente composta pelo conjunto do elemento cadastral estremo, é designada neste capítulo como contorno de prédio. Assim, é o contorno que define os limites de cada propriedade sendo esta informação de excepcional importância para o cadastro. Deste modo, os métodos de análise necessitam ser bastante precisos, de modo a identificar rigorosamente o limite de cada prédio.

A extracção automática dos contornos fechados neste tipo de mapas apresenta várias dificuldades, nomeadamente descontinuidades, elementos conexos e também o cruzamento de linhas. Adicionalmente, é necessário associar a cada prédio o seu contorno fechado (ver figura 6.1).

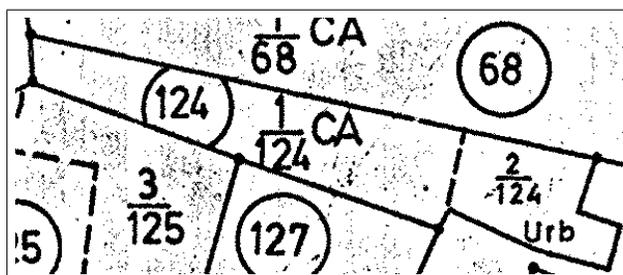


Figura 6.1: Exemplos de alguns problemas na extracção do contorno, nomeadamente descontinuidades no contorno, elementos conexos ao contorno, cruzamento de linhas e também é necessário associar a cada círculo um contorno.

Infelizmente, na literatura não existe um algoritmo que resolva por si só o variado número de problemas da extracção do contorno, mas apenas métodos simples que podem resolver cada problema de forma isolada. Assim, são investigados vários algoritmos, de modo a solucionar cada problema individualmente, em vez de tentar implementar métodos complexos, que resolvam todos os problemas de uma só vez.

Nas secções seguintes são estudados todas os problemas da extracção do contorno e são evidenciadas as possíveis soluções.

6.2 Problemas na Extração do Contorno

A humidade e a natural erosão dos documentos alteram a imagem relativamente à sua representação original. Quanto mais avançada for a idade do mapa, maior é a degradação da imagem e conseqüentemente maior a informação perdida, após o processo de *scanning*. Existem mapas de tal maneira degradados que mesmo na vectorização manual, os operadores necessitam de ver o mapa original no formato papel para decidirem sobre as zonas degradadas.

Para uma melhor compreensão é necessária uma análise mais detalhada aos problemas específicos da extração do contorno, nomeadamente as descontinuidades no contorno, o cruzamento das linhas e os elementos conexos ao contorno. Nas próximas sub-secções, estes problemas serão abordados individualmente.

6.2.1 Descontinuidades

O contorno do prédio é representado por segmentos de recta contínuos, designados por extremas de prédio. No entanto, as extremas de prédio apresentam descontinuidades devido ao ruído e degradação, introduzidos durante o processo de *scanning*. O ruído é introduzido pela adição e/ou eliminação dos pixels negros, processo que ocorre devido aos incorrectos níveis de *threshold* na digitalização.

As descontinuidades do contorno são frequentes, por exemplo a figura 6.2 apresenta dois tipos de descontinuidades, nomeadamente no lado esquerdo da imagem uma descontinuidade simples e no lado direito duas descontinuidades muito próximas no contorno. Em processamento de imagem, o algoritmo de *component labeling* é normalmente utilizado para segmentar os elementos relativamente ao seu tamanho, separando os elementos co-

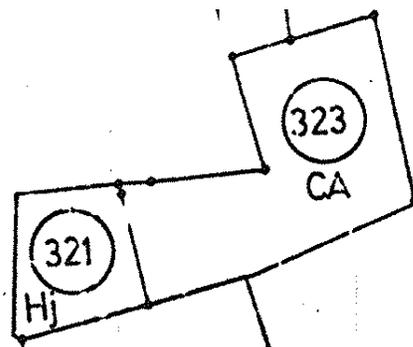


Figura 6.2: Esta imagem mostra dois tipos de descontinuidades no contorno: o lado esquerdo apresenta uma descontinuidade simples enquanto o lado direito tem duas descontinuidades muito próximas.

nexos (elementos maiores) dos elementos não conexos (elementos mais pequenos). Neste domínio, os elementos maiores são normalmente os elementos pertencentes ao contorno, compostos principalmente por segmentos de recta. Assim, aplicando este algoritmo, é possível separar o contorno entre os caracteres e as linhas a tracejado, porém a aplicação deste método, leva à eliminação de alguns elementos entre as duas descontinuidades. Desta forma, em vez de haver duas descontinuidades próximas, é introduzida uma grande descontinuidade com a perda do elemento não conexo.

Outra solução possível é utilizar a transformada de Hough [108] para restaurar as descontinuidades, contudo os resultados deste algoritmo não são satisfatórios, devido à existência de curvas, de elementos conexos ao contorno e das próprias descontinuidades (ver figura 6.3).

6.2.2 Cruzamento de Linhas

Nos mapas cadastrais, a geometria da propriedade é definida essencialmente pelas estremas do prédio e é representada por um contorno fechado, o qual é desenhado usando

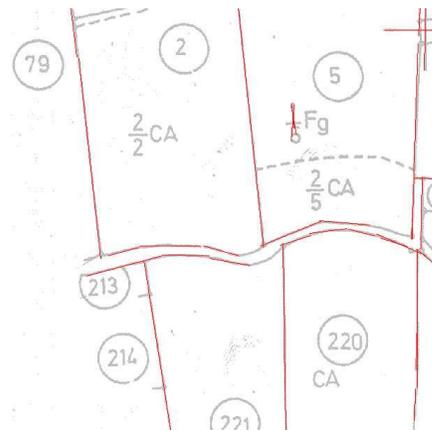


Figura 6.3: A detecção dos segmentos de recta através da transformada de Hough não resolve o problema da extração do contorno devido ao contorno apresentar descontinuidades e curvas.

segmentos de recta. Por outro lado, dois prédios vizinhos têm pelo menos um segmento de recta que partilha o contorno fechado de ambos. Deste modo, considerando o último facto, o mesmo segmento de recta está associado a dois prédios distintos.

Adicionalmente, seguindo a linha do contorno, são encontrados alguns pontos de intercepção no contorno do prédio, e nessas posições existem muitos caminhos possíveis para seguir. Mesmo utilizando métodos de decisão, tais como o seguimento do contorno sempre pelo lado de dentro ou sempre pelo lado de fora, o problema não é resolvido dada a existência de elementos conexos ao contorno. Os algoritmos baseados no seguimento de linhas apresentam alguns problemas para resolver o cruzamento de linhas, uma vez que os métodos de *backtracking* precisam de considerar o melhor caminho a seguir surgindo assim novas indecisões.

6.2.3 Elementos Conexos ao Contorno

Os elementos conexos ao contorno dos prédios podem ser de diferentes tipos, cruces, círculos, arcos, tracejados, caracteres ou linhas rectas.

Uma vez que os prédios partilham um ou mais segmentos de recta, é inicialmente aplicado o algoritmo de *skeletonization* [25], de modo a obter os pontos médios do contorno. Se no contorno existir elementos conexos, após a aplicação desse algoritmo, o contorno irá apresentar distorções (ver figura 6.14). Porém, se o algoritmo de *skeletonization* não for utilizado, sucede o mesmo problema, uma vez que os elementos conexos necessitam primeiro de ser segmentados das linhas do contorno, para uma correcta extração.

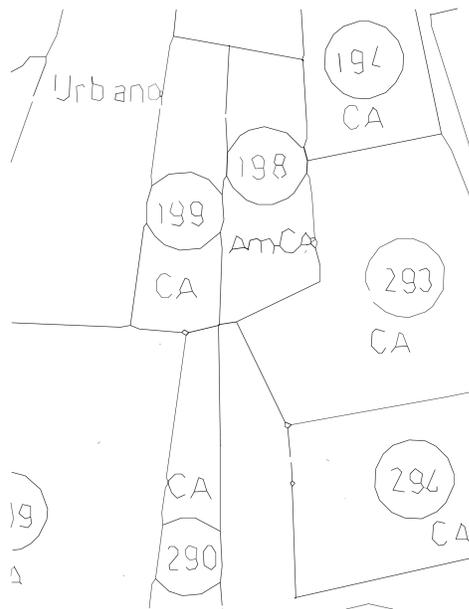


Figura 6.4: A aplicação do algoritmo de *skeletonization* origina distorções quando existem elementos conexos ao contorno.

6.3 Soluções para a Extracção do Contorno

Tal como foi anteriormente referido, existem duas opções distintas sobre o modo como uma máquina deve reconhecer um documento. Por um lado, descrevendo as primitivas básicas e depois interpretando-as ou por outro lado, simplificando o problema olhando para a imagem como um todo e sem conhecer as primitivas básicas pela qual a imagem é constituída. Como neste caso, o objectivo é reconhecer os prédios e parcelas, considerar, por exemplo, na extracção dos prédios a detecção das linhas rectas e a sua intersecção, iria aumentar consideravelmente a complexidade do problema. Assim, o problema da extracção dos prédios deve ser simplificada, partindo da posição do círculo, utilizando um algoritmo de preenchimento para detectar os pontos do contorno e finalmente associando o contorno vectorizado à entidade cadastral.

De qualquer forma, a extracção do contorno apresenta três principais tipos de problemas. Nas próximas secções, são analisadas algumas soluções para cada problema, de modo a analisar as vantagens e desvantagens de cada método.

6.3.1 Métodos de Processamento

A digitalização da imagem introduz ruído e deformação, o que vai originar sujidade e descontinuidades nas secções cadastrais. Os mapas cadastrais apresentam caracteres, tracejados e círculos sobre o contorno. Esta sobreposição dificulta consideravelmente a extracção, uma vez que é necessário primeiramente realizar a segmentação dos dois elementos conexos.

É deveras importante observar em profundidade cada problema da extracção do contorno, de maneira a escolher os algoritmos que melhor se adequem. As principais difi-

culdades desta análise são as descontinuidades, os elementos conexos e o cruzamento das linhas do contorno. A implementação do algoritmo de seguimento de linha com filas de espera [117] é um processo complexo, especialmente no caso de cruzamento de linhas do contorno, devido à variedade de caminhos a seguir. Adicionalmente o algoritmo de seguimento de linhas não resolve os problemas dos elementos conexos no contorno, da associação de cada parcela ao contorno e nem das descontinuidades sobre o mesmo. Este processo deve realizar a extração precisa do contorno fechado, as descontinuidades devem ser detectadas e corrigidas, numa fase posterior. Também é necessário associar uma entidade cadastral representada pela forma do círculo, a um contorno. Deste modo, é crucial considerar todas estas condições para a elaboração de um algoritmo de extração do contorno.

Na literatura não existe nenhum método genérico capaz de resolver todos estes problemas, contudo para cada problema existem pequenas possíveis soluções. É ponto assente que cada método deve resolver um problema específico, porque assim além do método aumentar a sua robustez, a implementação é mais simples. Cada algoritmo apresenta diferentes características ou propriedades e todos eles permitem em conjunto resolver a maioria das dificuldades.

Os métodos de processamento para a extração do contorno apresentam diferentes propriedades que se adequam a este domínio de análise. O algoritmo de preenchimento com blocos tenta obter o contorno desde a posição do círculo, usando o algoritmo de *flood fill*, mas utilizando blocos, com o objectivo de diminuir a sua sensibilidade às descontinuidades do contorno. O algoritmo de Lohmann segmenta uma região dentro do contorno, detectando as descontinuidades e o seu tamanho. O algoritmo de *seed-segment* é um algoritmo

de detecção de segmentos de recta e é usado para segmentar as linhas rectas dos elementos conexos ao contorno. O algoritmo de seguimento de linhas utiliza os segmentos do contorno para corrigir o contorno extraído do mapa.

6.3.2 Algoritmo de Preenchimento com Blocos

A extração do contorno é simplificada utilizando o conhecimento prévio, de que um círculo representa uma entidade cadastral e que nos mapas cadastrais os círculos encontram-se dentro dos prédios. Utilizando este conhecimento, é possível criar um algoritmo de preenchimento (*flood-fill*), começando na posição do círculo, preenchendo toda a área do prédio e finalmente, detectando todos os pontos do contorno (ver figura 6.5). Depois da detecção, os pontos do contorno são vectorizados de modo, a obter as linhas rectas que definem o prédio.

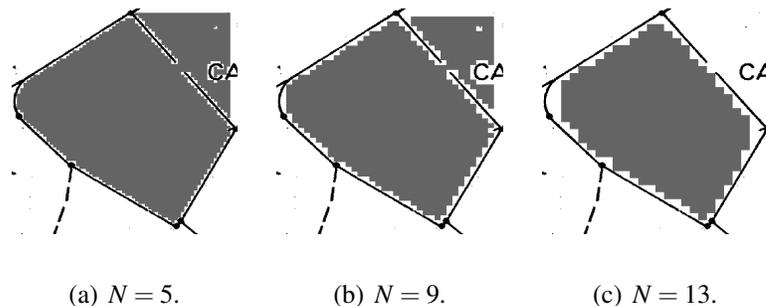
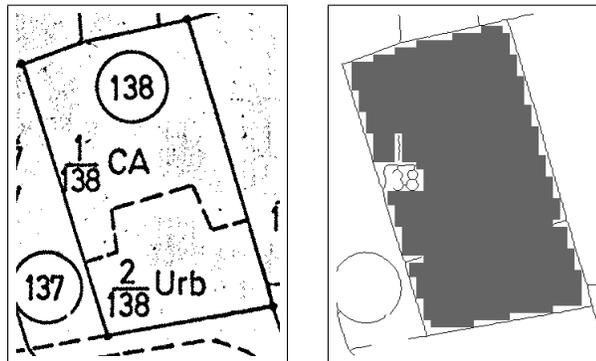


Figura 6.5: Um exemplo da aplicação do algoritmo de preenchimento, utilizando blocos de diferentes tamanhos (N).

Uma vez que, o contorno apresenta descontinuidades, o algoritmo de preenchimento normal sai para fora do contorno e neste caso não é aplicável. Contudo, utilizando a mesma ideia com blocos de tamanho variável ($N \times N$), considerando que é insensível às pequenas descontinuidades do contorno, o algoritmo não excede o limite de prédio (ver figura 6.5-c).

A imagem de entrada é pré-processada pelo algoritmo de *thinning* [124] para detectar os pontos centrais do contorno, dado ser necessário ter os mesmos pontos para os mesmos segmentos entre contornos vizinhos.



(a) A imagem inicial de um contorno fechado.

(b) O algoritmo de preenchimento com blocos na imagem emagrecida.

Figura 6.6: Um exemplo da aplicação do algoritmo de preenchimento utilizando blocos.

O algoritmo de preenchimento começa com a detecção da posição do círculo e preenche o prédio com blocos de tamanho $N \times N$. Este algoritmo é iterativo, na primeira iteração a pilha apresenta apenas a posição do círculo; na segunda, os primeiros quatro vizinhos do primeiro bloco, caso os blocos não tenham qualquer pixel a preto no seu interior, o algoritmo continua o preenchimento procurando pelos blocos vizinhos, de modo a preencher todo o interior do contorno (ver figura 6.6).

A detecção dos pontos do contorno é realizada considerando que os blocos tocam sobre o contorno. Cada bloco tem blocos vizinhos fora do contorno e estes são testados em cada direcção, de modo a detectar todos os pontos do contorno. O objectivo é detectar todos os pontos possíveis, contudo como existem partes do contorno onde é impossível detectar

todos os pontos, é necessário utilizar o método de seguimento do contorno para detectar todos os pontos do contorno que estão entre duas partes já detectadas. O algoritmo de seguimento de linhas é aplicável, uma vez que é utilizada a imagem *thinning* dos contornos e sem os elementos conexos, resultando um contorno com linhas de largura igual a um pixel.

Infelizmente, este método não funciona em caso de grandes descontinuidades, mais precisamente quando estas ultrapassam o tamanho dos blocos. Normalmente, as grandes descontinuidades acontecem quando duas pequenas descontinuidades estão próximas, uma vez que o algoritmo de *component labeling* recolhe o pequeno elemento entre as duas descontinuidades (ver figura 6.2).

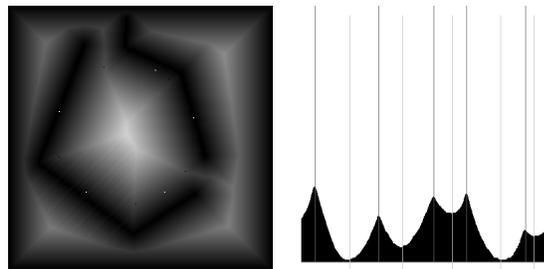
Assim, o algoritmo de preenchimento utilizando blocos não é aplicável para grandes descontinuidades. Também utilizando este método não é possível saber se um bloco se encontra fora do prédio, uma vez que o algoritmo apenas tem o valor de *threshold* para o tamanho da maior descontinuidade aceite (tamanho N).

O algoritmo de preenchimento com blocos é a base para a extração do contorno e os próximos algoritmos descritos são implementados de modo a resolver algumas das suas limitações.

6.3.3 Algoritmo de Lohmann

O primeiro problema do algoritmo de preenchimento com blocos, já descrito anteriormente, é que aquando da existência de grandes descontinuidades os blocos ultrapassam as extremas de prédio. O algoritmo de Lohmann [77] é assim utilizado para segmentar as regiões fechadas dentro do prédio, que está aberto devido a grandes descontinuidades. As

ideias-chave são utilizar a transformada da distância da imagem (ver figura 6.7(a)) para encontrar rapidamente a distância entre um ponto e o seu vizinho mais próximo e por outro lado, a ideia de cobrir a região com discos máximos, uma vez que a utilização de discos permite detectar grandes discontinuidades no contorno.



(a) A detecção dos valores máximos nos pontos da circunferência.
(b) A detecção dos extremos nos pontos da circunferência.

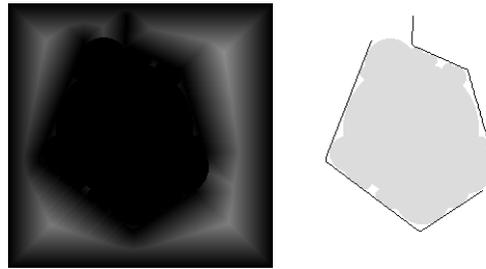
Figura 6.7: Na primeira iteração do algoritmo de Lohmann são detectados os extremos do círculo máximo centrado na posição do máximo local.

Este algoritmo necessita de três parâmetros para iniciar o processamento, nomeadamente um ponto dentro do contorno, o tamanho mínimo do disco aceite e o tamanho máximo da descontinuidade aceite. O tamanho mínimo do disco aceite permite obter mais ou menos detalhe na cobertura da região, enquanto o tamanho da máxima descontinuidade aceite permite controlar se a descontinuidade do contorno é realmente uma descontinuidade ou uma característica do contorno. As zonas mais fechadas são exemplos disso e dependendo do valor da máxima descontinuidade permitida, podem ser consideradas como descontinuidades ou como características do contorno.

Tal como o método de preenchimento com blocos, este algoritmo começa a segmentação também num ponto dentro do prédio, nomeadamente na posição do círculo. O algoritmo de

Lohmann foi ligeiramente otimizado começando no máximo local ao invés de começar no ponto inicial, tal como no algoritmo original. Deste modo, a partir do ponto inicial é seguido o gradiente máximo na imagem da transformada da distância e até encontrar o máximo local. O método converge de forma mais célere começando no máximo local ao invés do ponto inicial, uma vez que o primeiro disco tem aí o seu raio máximo (ver figura 6.7(a)) e a região é mais rapidamente coberta usando o maior disco inicial.

O algoritmo de Lohmann inicia o seu processamento num ponto dentro do contorno e para esse ponto é calculado o máximo local seguindo o gradiente da imagem da transformada da distância, i.e., onde o disco tem o seu máximo raio. Na posição do máximo local, são obtidos todos os pontos da máxima circunferência, de modo a que por um lado, identifique possíveis pontos de descontinuidade no contorno (pontos terminais) e por outro, encontre os máximos locais da circunferência de modo, a utilizar estes pontos para preencher novos discos centrados nestas posições. Tanto o teste à descontinuidade assim como o teste à verificação se o círculo encontra-se dentro do prédio, são utilizados para os pontos terminais, com os objectivos de detectar se a posição do disco está dentro do contorno e se o tamanho da descontinuidade é aceitável. Se o disco não for descartado, por estar fora da região, é realizada uma nova procura na imagem da transformada da distância de modo a encontrar os máximos locais no limite do disco e todos estes máximos são adicionados novamente à *stack* (ver figura 6.7(b)). Uma vez que a imagem da transformada da distância é marcada com pixels a preto para todos os discos encontrados (ver figura 6.8(a)), a região é coberta, obtendo-se assim uma região fechada dentro do prédio (ver figura 6.8(b)). O processo é iterativo e apenas acaba quando a *stack* não tiver pontos da posição central de novos discos possíveis.



(a) A imagem da transformada da distância após 21 iterações. (b) O contorno aberto é segmentado cobrindo a região com discos.

Figura 6.8: O resultado do algoritmo de Lohmann aplicado ao contorno com grandes descontinuidades.

Os dois métodos que afectam directamente a performance do algoritmo são a detecção dos máximos locais e o teste que verifica se o círculo está dentro ou fora do contorno. Estes dois métodos são bastante sensíveis e necessitam de ser devidamente testados de modo a obter-se os melhores resultados.

O procedimento que verifica se um ponto está ou não dentro da região do contorno é usado para verificar se um disco está dentro ou fora do contorno (ver figura 6.9). Para encontrar a linha da descontinuidade são utilizados o ponto de descontinuidade da circunferência do disco e o vizinho mais próximo do contorno, de maneira a medir o tamanho da descontinuidade. Se a linha recta que junta os dois pontos está dentro da região relativamente ao centro do disco, então o disco é rejeitado.

A detecção dos máximos locais é implementada colocando os valores da transformada da distância dos limites da circunferência dentro de uma lista circular. Os valores na lista são suavizados e os picos são encontrados considerando a concavidade dos valores (ver figura 6.7(b)).

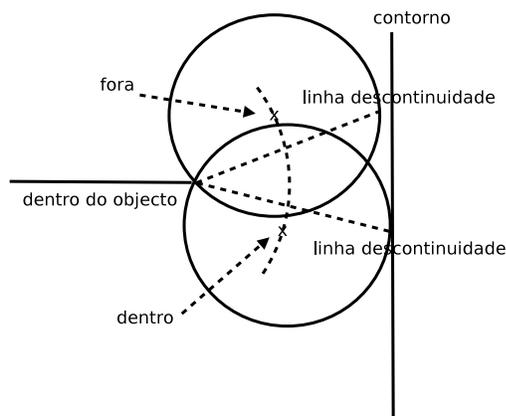


Figura 6.9: O teste dentro do prédio verifica se um círculo está dentro ou fora da região.

Depois de obter a lista de círculos que segmentam o contorno fechado, podemos considerar os blocos obtidos do algoritmo de preenchimento com blocos, mas apenas considerando os blocos que estão dentro de pelo menos um desses círculos. Deste modo podemos utilizar o algoritmo de preenchimento com blocos para detectar os pontos do contorno dentro do prédio, mesmo tendo grandes descontinuidades.

Este algoritmo apresenta algumas desvantagens, nomeadamente contornos com zonas mais fechadas são consideradas como zonas com descontinuidades, uma vez que os parâmetros de *threshold* não são adaptativos. E adicionalmente, o método não considera elementos conexos ao contorno.

6.3.4 Algoritmo de *Seed-Segment*

Até então, a extração do contorno inicia-se sempre na posição do círculo e não considera os elementos conexos. A extração dos pontos do contorno utilizando o algoritmo de preenchimento com blocos, quando o contorno apresenta elementos conexos, tem como resultado a obtenção do contorno de prédios vazio ou incompleto. Tal como referido an-

teriormente, uma possível solução é segmentar primeiro os elementos conexos ao contorno utilizando algoritmos de detecção de segmentos de recta, tal como o algoritmo de Hough [108], mas após a realização de alguns testes, verificou-se que não solucionava o problema dada a existência de algumas curvas (ver figura 6.3).

Uma linha é caracterizada por ter uma orientação e uma largura características. As linhas rectas apresentam uma propriedade importante, nomeadamente as características das linhas são constantes, por exemplo se um segmento normal for usado para descrever parte de uma linha recta, então o mesmo elemento pode ser utilizado com a mesma orientação e a mesma largura para descrever as partes que restam dessa linha (ver figura 6.10).

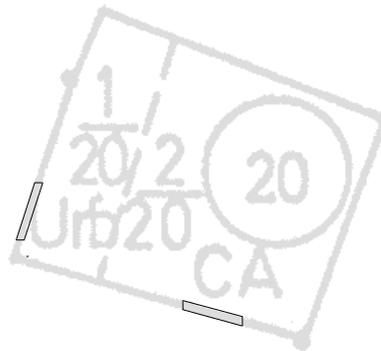


Figura 6.10: O segmento padrão (*seed-segment*) é um rectângulo que descreve as características elementares de um segmento de recta, nomeadamente a largura de linha e a sua orientação.

O algoritmo de *seed-segment* [111, 114] inicia a detecção das linhas rectas a partir dos elementos conexos, encontrando primeiramente os segmentos elementares (ver figura 6.11) que são utilizados para decompor um elemento conexo em segmentos de recta. Um segmento padrão é um pequeno rectângulo com uma orientação que é característica da linha e é encontrado a partir de um conjunto de condições, tais como a simetria de ambos os lados da linha e a simetria inferior e superior da linha. Adicionalmente, os segmentos elemen-

tares ou padrão, têm um comprimento de linha pelo menos quatro vezes maior do que a largura da linha. Tal como nos últimos algoritmos, a procura começa a partir da posição de entrada, existindo um intervalo de procura para encontrar os segmentos elementares. No final, o segmento elementar é o padrão elementar para a detecção de linhas rectas.

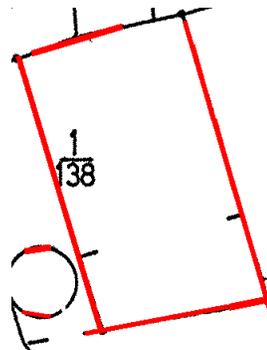


Figura 6.11: Um exemplo da detecção de *seed-segments* descrevendo as características básicas da linha, nomeadamente a orientação e a largura. Estes *seed-segments* são os padrões básicos para realizar o seguimento da linha no processo de detecção das linhas rectas.

Este algoritmo tem dois passos distintos, a procura pelos *seed-segments* e o seguimento da linha. No primeiro, os segmentos padrão são encontrados utilizando os métodos anteriormente descritos e utilizando o seguimento para a frente e para trás, são obtidos alguns rectângulos com orientação arbitraria. Esses rectângulos descrevem completamente os segmentos detectados e fazem uma aproximação às linhas rectas do contorno.

Definitivamente, o objectivo do método é decompor elementos complexos conexos em linhas rectas. Assim, como os arcos e as curvas não são detectadas, os elementos conexos são segmentados do contorno, uma vez que, o contorno é composto principalmente por linhas rectas (ver figura 6.11).

Felizmente, depois de detectar as linhas rectas, é fácil detectar os elementos conexos ao contorno. Após a segmentação, os elementos conexos são processados pelo reconheci-

mento de caracteres (ver figura 6.12). Finalmente, o contorno é reconstituído e são extraídas correctamente as linhas rectas.

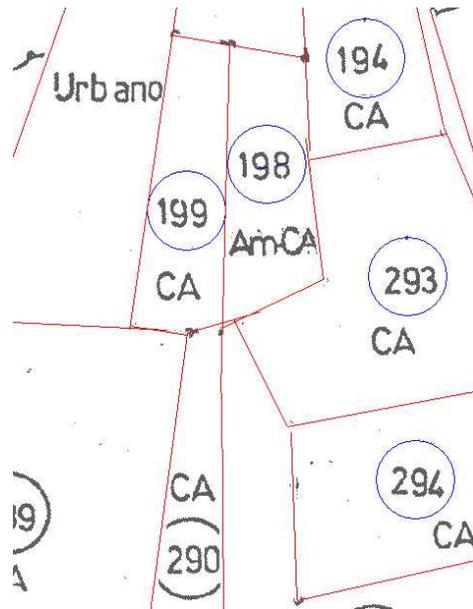


Figura 6.12: Esta figura mostra o resultado final do algoritmo de *seed-segments*, após a aplicação da segmentação dos elementos conexos relativamente ao contorno.

6.3.5 Algoritmo de Seguimento de Linhas

Outro método possível para a extração do contorno é o algoritmo de seguimento de linhas. Até aqui, a posição do círculo foi sempre utilizada para extrair os contornos; contudo, este método não necessita de uma posição inicial.

O objectivo do algoritmo de seguimento de linhas é extrair os pontos do contorno através do seguimento das linhas do contorno. Este método é implementado segmentando os elementos não conexos do contorno e utilizando algoritmos de *component labeling* para obter na extração apenas o contorno. Em maior detalhe, o algoritmo começa num ponto do contorno e segue o caminho até chegar ao ponto de partida.

Este método apresenta alguns problemas neste domínio em particular, tais como o cruzamento de linhas, os elementos conexos ao contorno e a presença de descontinuidades. Por exemplo, nos pontos de intersecção, a escolha do caminho correcto a seguir é realizado sempre na mesma direcção (sempre pela direita ou sempre pela esquerda) (ver figura 6.13). Como é pretendido obter um contorno fechado, a direcção de seguimento é escolhida considerando o ângulo entre dois pontos próximos no contorno. As decisões nos pontos de intersecção consideram a posição do círculo que está sempre dentro do contorno e é sempre seguida a direcção mais interna. Contudo, se existirem elementos conexos ao contorno, então não é possível extrair correctamente todos os pontos. Adicionalmente, nas descontinuidades no contorno, o caminho não é completamente definido e é necessário tomar mais decisões.

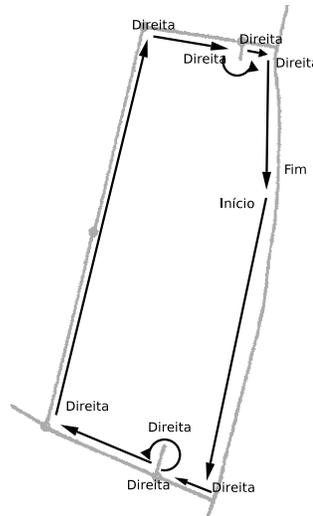


Figura 6.13: No algoritmo de seguimento de linha, é sempre tomada a mesma direcção em todos os pontos de intersecção.

A implementação do algoritmo necessita do emagrecimento das linhas do contorno, de modo a obter as linhas com largura igual a um pixel, para mais facilmente seguir o

contorno. Algumas vezes, este processamento distorce a imagem e introduz alguns erros na extracção do contorno. O algoritmo é muito sensível ao ruído, pois uma má ligação no contorno é suficiente para seguir um caminho errado.

Este método demonstra muitas dificuldades quando existe conexão de caracteres, tornando-se numa solução impraticável da extracção do contorno para determinados mapas cadastrais. Contudo, em condições ideais, i.e. sem descontinuidades e sem elementos conexos ao contorno, é um método aplicável.

6.4 Algoritmo de Vectorização Rosin-West

A extracção do contorno é implementada conjugando o algoritmo de preenchimento com blocos e com o processamento complementar dos algoritmos anteriormente descritos. Depois de serem obtidos todos os pontos do contorno de um prédio, estes são vectorizados de modo a compactar a informação no contorno. O método de vectorização utilizado foi o algoritmo de Rosin-West [100], permitindo obter uma elevada precisão para os segmentos de recta (ver figura 6.14).

6.5 Solução Geral - Método Composto

Depois de se ter estudado todos os problemas de forma individual, é agora necessário realizar o agrupamento dos diferentes métodos de modo a utilizar todas as características específicas de cada um, obtendo um método geral capaz que extrair automaticamente os contornos. Tal como foi previamente referido, a extracção dos contornos apresenta três problemas principais, nomeadamente as descontinuidades, os elementos conexos ao con-

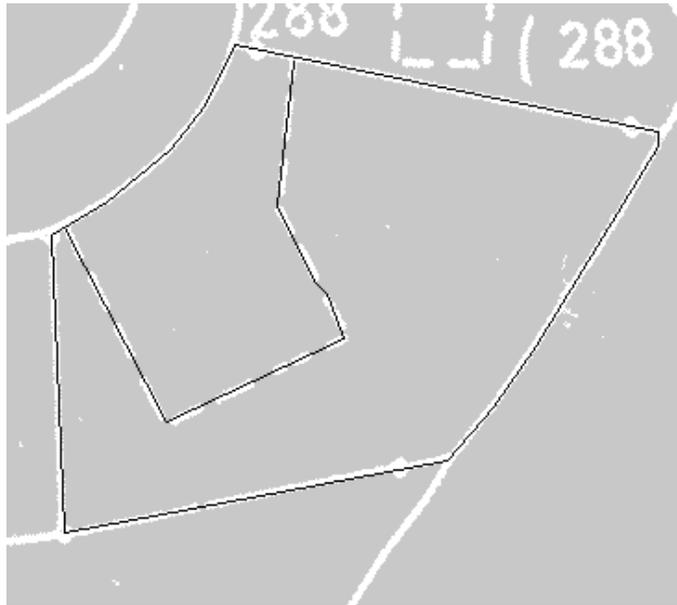


Figura 6.14: Vectorização das linhas do contorno utilizando o algoritmo de Rosin-West.

torno e o cruzamento das linhas do contorno.

Esses problemas têm uma hierarquia para a sua resolução, uma vez que os métodos encadeados devem seguir uma ordem específica de maneira a processar correctamente todos os problemas. O encadeamento dos métodos é simples e é descrita nos seguintes passos:

1. O algoritmo de Block-Fill é aplicado e se o contorno extraído for válido (fechado e com uma área mínima), então ir para o passo 5;
2. Se o contorno obtido tiver poucos pontos, é aplicado o algoritmo de Seed-Segment para segmentar os elementos conexos ao contorno. (Assim a imagem tem apenas linhas rectas e curvas e não arcos, círculos, tracejados ou caracteres);
3. O algoritmo de Block-Fill é novamente aplicado e se a extracção do contorno for válida, então ir para o último passo;

4. Se o contorno extraído estiver aberto, é aplicado o algoritmo de Lohmann para segmentar o interior do contorno. Usando a região fechada dentro do contorno, são obtidos os pontos através do algoritmo de Block-Fill;
5. Os pontos dos marcos detectados são adicionados sobre o contorno extraído. Fazendo o seguimento da linha do contorno são ligados os marcos adjacentes através de um segmento de recta. Se o segmento de recta for coincidente com o contorno considerar o segmento, caso contrário adicionar este intervalo de linha para vectorização através do algoritmo de Rosin-West [100]. Os pontos do contorno são validados usando o algoritmo de Line-Following.

Na sequência de todos os passos é obtido o contorno. O contorno é fechado e descrito principalmente por linhas rectas, dada ser uma característica das secções cadastrais. Nesta detecção, os contornos são validados usando o algoritmo de *Line-Following*. Finalmente, os contornos extraídos incorrectamente são colocados na lista de contornos problemáticos, enquanto os outros são aceites.

6.6 Conclusões

Uma vez que os problemas são complexos e muito diferentes de mapa para mapa, a análise deve ser realizada com métodos simples. A abordagem deve ser geral e não particular, assim um método deve resolver apenas um problema. Porque não é possível extrair todos os contornos do mapas, os algoritmos devem ser desenhados para extrair a sua maioria. Assim, os problemas precisam de ser completamente identificados de modo a escolher os algoritmos com características específicas e de maneira a obter a maior taxa de reconhe-

cimento possível.

Os algoritmos estudados resolvem todos os problemas da extração do contorno nos mapas cadastrais Portugueses. A taxa de reconhecimento pode ser melhorada através da resolução de problemas particulares, porém o sistema tornar-se-ia mais complexo e essa melhoria de resultados não teria uma consequência directa. Usando os métodos apresentados, podemos concluir que o problema da extração do contorno é globalmente resolvida.

Capítulo 7

Integração e Interface

Em cada capítulo são descritos vários métodos de análise que realizam o reconhecimento de apenas um determinado tipo de objecto gráfico, num ponto de vista individual. O reconhecimento é realizado detectando cruces, círculos, arcos, tracejados, caracteres, marcos e extremas de prédio. No entanto, após o reconhecimento de cada objecto gráfico é necessária a integração de todos os métodos de análise com vista à realização do reconhecimento e à interpretação da informação contida num mapa completo, de um ponto de vista global.

Os módulos não utilizam a informação de outros módulos de análise para iniciar o seu processamento. Não existe dependência entre os módulos, excepto para o caso da extracção do contorno, onde é utilizada a posição do círculo, uma vez que a extracção é assim mais fácil de implementável e é realizada de um modo mais organizado.

A integração dos módulos de análise permite passar de um conjunto de módulos que individualmente reconhecem objectos gráficos para um sistema de reconhecimento automático dos mapas cadastrais. O sistema reconhece todos os elementos cadastrais, agrupa-

os em conjuntos e faz a sua validação, considerando algumas regras e especificidades do domínio cadastral.

A integração de todos os métodos de análise apresenta várias dificuldades, sendo necessário criar uma arquitectura para a implementação do sistema, organizar toda a informação detectada e criar um interface para a validação da informação. A implementação da integração dos módulos num sistema é analisada nas próximas secções.

7.1 Problemas de Integração

A integração dos métodos de análise num sistema de reconhecimento, origina naturalmente uma série de problemas relacionados principalmente com a ligação dos módulos de reconhecimento (estratégia de reconhecimento) e a robustez de cada método de análise. Um sistema de reconhecimento não é apenas a ligação de um conjunto de métodos, mas é um conjunto de decisões, estratégias e abstracções que permitem com que um sistema opere, reconheça, interprete e classifique de um ponto de vista global.

7.1.1 Abordagem Global

O processo de desenvolvimento de um sistema de reconhecimento automático, atravessa várias fases de implementação. A partir duma pequena amostra, que se quer representativa do problema global, são inicialmente desenvolvidos módulos que reconhecem os objectos gráficos. Esses métodos de análises são implementados e são obtidos resultados satisfatórios para uma pequena amostra, no entanto não é suficiente, pois aplicando os mesmos métodos a um mapa completo, surgem novas dificuldades, sendo assim necessário

considerar mais características na análise. Uma vez que o objecto em estudo é um conjunto diversificado de secções (ver figura 7.1), com características várias, devem-se considerar todas as características semelhantes da mesma classe de mapas.

Para alcançar robustez dos algoritmos é necessário conhecer o universo dos mapas cadastrais. Neste caso, a enorme variedade de mapas torna o problema complexo sendo necessário a implementação de métodos simples e directos, de modo a extrair dos mapas cadastrais a maior quantidade de informação possível.

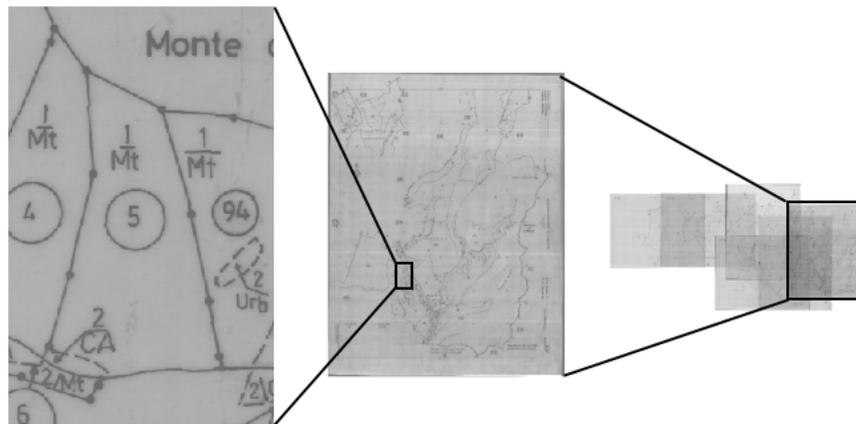


Figura 7.1: Esta imagem apresenta uma amostra com alguns detalhes das características dos mapas cadastrais. No processo de implementação, analisa-se o universo dos mapas cadastrais considerando todas as características e detalhes, identifica-se o que têm em comum, estabelecendo padrões de forma a generalizar e dar robustez ao sistema.

7.1.2 Qualidade do Método

A implementação de um método de reconhecimento é um processo iterativo, no sentido em que inicialmente existe a ideia de resolução com vista à detecção de um determinado objecto, que apenas poderá ser validada após a implementação e análise dos resultados,

aplicando dados reais. Muitas vezes, depois dessa primeira implementação a ideia é reformulada e são considerados novas características que não eram tão evidentes até então. O reconhecimento de imagem é uma área com uma enorme variedade de métodos, permitindo a aplicação de um conjunto de algoritmos que embora simples, quando combinados e implementados tornam-se num método sofisticado capaz de resolver todas as características do problema em estudo. Deste modo, são os resultados que evidenciam a boa ou má implementação dos métodos em estudo, assim como todas as características que devem ser consideradas nessa implementação. Por vezes, apenas se conhece verdadeiramente a profundidade do problema após analisar todas as dificuldades e potenciais soluções.

7.1.3 Generalidade do Método

A generalidade da análise é fundamental quando o problema em estudo é consideravelmente complexo. Por exemplo, num caso prático onde a amostra é pequena e a variabilidade da informação é enorme, é necessário pensar de um ponto de vista abstracto, de modo a tomar consciência de qual é a fronteira entre a especialização e a generalidade na detecção.

Por vezes, seria muito satisfatório aumentar a taxa de reconhecimento numa dada família de mapas, mas ao especializar-se os métodos perderia-se a generalidade global da análise. Assim, torna-se necessário tomar consciência do limite das características a reconhecer, para que deste modo, nunca se perca a generalidade. Por exemplo, no caso dos círculos, reconhecer todos os círculos, inclusive os mal desenhados teria como consequência aumentar o nível de aceitação e assim aumentariam as falsas classificações. Deste modo, é importante saber que um determinado algoritmo apenas pode reconhecer até determinada

qualidade de imagem e que os objectos degradados são uma excepção, não devendo ser considerados como importantes na análise, uma vez que o mais importante é a generalidade do método, ou seja reconhecer todos os objectos que pertencem a determinada classe e apresentem a qualidade necessária para uma correcta classificação.

7.1.4 Ligação dos módulos

Ao implementar o sistema de reconhecimento são tomadas algumas decisões com vista a agrupar todos os módulos de reconhecimento. A sequência do reconhecimento faz-se dos elementos mais simples para os mais complexos, uma vez que a probabilidade de reconhecimento é maior para objectos gráficos simples do que para objectos mais complexos. Depois de detectados, os objectos gráficos são imediatamente eliminados da imagem em análise, para assim facilitar reconhecimentos posteriores.

Os métodos de análise são implementados de modo a não existirem dependências com outros módulos de extracção de informação. A independência entre os módulos é necessário para existir flexibilidade no sistema e para as taxas de reconhecimento não dependerem de diferentes módulos de análise.

7.1.5 Dificuldades na Implementação

A implementação é um processo consideravelmente demorado, uma vez que a programação de um algoritmo necessita de uma atenção particular e um *debug*¹ exaustivo. A imagem não é uma estrutura de dados fácil de trabalhar, uma vez que apesar de se conseguir observar a imagem num determinado instante, dificilmente consegue-se monitorizar os milhares

¹Debug é uma palavra que na óptica da programação significa ver todos os estados do programa de modo a identificar e corrigir falhas.

de iterações a nível pixel de um dado algoritmo. Deste modo, é fundamental encontrar estratégias de *debug* gráfico que apontem não apenas o resultado final, mas que criem sequências de imagens demonstrativas do algoritmo de modo a identificar mais rapidamente potenciais problemas.

7.2 Arquitectura do Software

A arquitectura do software é importante para aumentar a flexibilidade do processo de análise e para integrar um conjunto de componentes de software estável, de maneira a reutilizar o código da aplicação em outros programas. Um sistema de software deve estar suficientemente bem organizado de tal modo, que possa crescer facilmente sem aumentar proporcionalmente a sua complexidade. Assim, é necessário criar uma arquitectura de software que minimize o código de programação e o erro, aumente a produtividade, e crie níveis de abstracção sucessivos, de tal modo que o programador apenas se foque no essencial, ou seja nos métodos de detecção.

A arquitectura do software está relacionada com o conceito de *pattern design*, que evidência o facto de na construção de uma aplicação, o problema de integração dos módulos ser padrão no desenvolvimento de software. Assim, a identificação de padrões conhecidos na literatura [3] é uma ajuda para descobrir os modelos apropriados, aumentando a generalidade, a flexibilidade e a reutilização dos componentes em outras aplicações.

Os sistemas modernos de reconhecimento de imagem [97, 30] utilizam principalmente a linguagem de programação C++. Nomeadamente porque existem várias livrarias com código para manipular estruturas de dados com imagem, assim ao invés de se realizar código para manipular imagens, pode utilizar-se código já realizado e bem testado, de

modo a aumentar consideravelmente o tempo de implementação, bem como o desempenho nas sucessivas optimizações.

A linguagem C++ apresenta também recursos valiosos na implementação de qualquer sistema, nomeadamente o recurso aos objectos e à aplicação de métodos genéricos que operam sobre diferentes estruturas de dados [3] (*templates*). Através destes recursos, são programados objectos em vários níveis de abstracção, de modo a minimizar o número de linhas, e mais importante do que isso, a minimizar o número de erros, pois como se sabe, tão importante como programar o código de forma simples, é escrever de forma o mais legível possível de maneira a alterar o código com facilidade. Esta linguagem está também no patamar da excelente performance e da boa abstracção. A linguagem Python é uma linguagem de alto nível e apresenta uma performance mais lenta, excepto nos casos em que os módulos de Python são compilados em linguagens de baixo nível e chamadas desde o Python. Por outro lado, a linguagem C é um exemplo de uma linguagem com uma excelente performance, devido sobretudo a ser uma linguagem mais próxima da máquina e também à grande evolução dos compiladores (GNU GCC 4.3), o que aumenta consideravelmente a velocidade de execução das aplicações compiladas nesta linguagem.

A figura 7.2 mostra a arquitectura do software e as ligações dos seus componentes. O *backend* apresenta um conjunto de classes que realizam a implementação do sistema a baixo nível, nomeadamente no *interface* com a imagem, no processamento de imagem e em algoritmos básico de imagem. Seguidamente, os métodos de processamento são implementados de modo a detectar as entidades cadastrais presentes na imagem. Os objectos cadastrais detectados são mostrados no interface e de seguida são gravados dois ficheiros com os dados detectados, um em formato XML (com os dados estruturados) e outro em

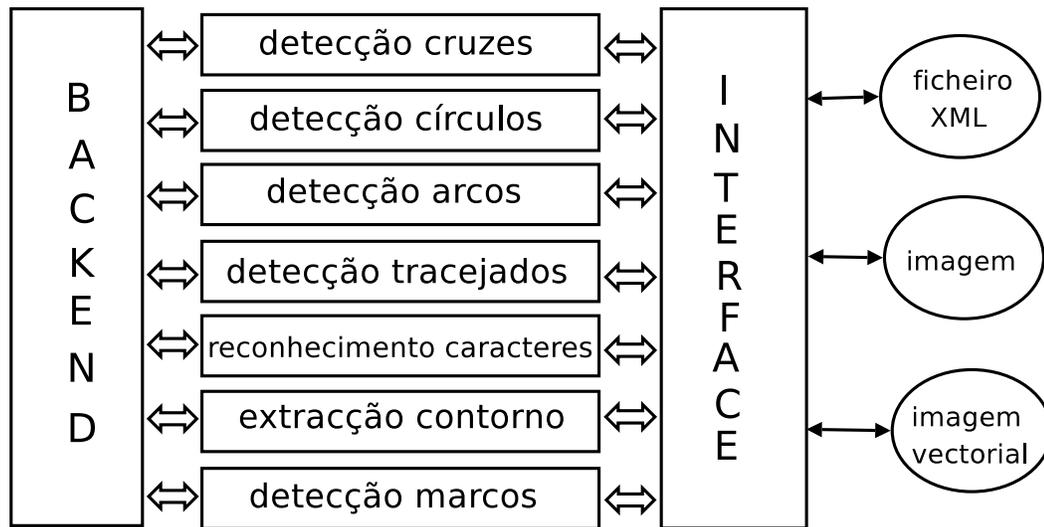


Figura 7.2: Esta imagem apresenta a ligação de todos os componentes de software do sistema de reconhecimento. Existe um *backend* com os métodos de análise que realizam a detecção de entidades básicas para os processos de detecção e que posteriormente enviam os dados para o interface, no sentido de visualizar e exportar essa informação para ficheiros no formato XML, vectorial ou de imagem.

formato vectorial (formato *Shapefile* ou *DGN*).

7.3 Organização da Informação

Os métodos de reconhecimento de imagem permitem extrair uma vasta informação sobre os vários objectos cadastrais presentes em cada mapa. Após a primeira detecção, a informação é agrupada de modo a realizar a validação de um ponto de vista da semântica cadastral.

Para realizar a organização e validação da informação é criada uma entidade cadastral para cada contorno de prédio detectado. Depois de conhecer o polígono que representa o prédio, são encontrados os objectos pertencentes ao mesmo constatando se um objecto cadastral está dentro ou fora do polígono, para tal considera o número de linhas do polígono

à esquerda e à direita do objecto e verifica a paridade do número. Após agrupar todos os elementos do prédio, tornam-se conhecidos os seus atributos, nomeadamente o número do prédio, o número das parcelas, o tipo de terreno e as estremas de prédio e de parcela.

A informação sobre as entidades cadastrais fica assim organizada e é seguidamente gravada num ficheiro XML com os dados estruturados de modo, a ser facilmente lida pelo interface. Esta informação fica também em condições de ser exportada para um ficheiro em formato vectorial, pronto a ser editada (ver figura 7.3).

```

<detecção>
  <image name="/home/tiago/disk/work/remap/src/debug/out_scan1rot.pbm"/>
  <processing>
    <entity x="7485" y="1275" number="53" type="">
      <circle x="7485" y="1275" radius="26" width="5"/>
      <contour x="7485" y="1275" color="1" width="5">
        <line x1="7489" y1="1220" x2="7531" y2="1220" color="1"/>
        <line x1="7531" y1="1220" x2="7533" y2="1222" color="1"/>
        <line x1="7533" y1="1222" x2="7544" y2="1342" color="1"/>
        <line x1="7544" y1="1342" x2="7475" y2="1376" color="1"/>
        <line x1="7475" y1="1376" x2="7453" y2="1333" color="1"/>
        <line x1="7453" y1="1333" x2="7446" y2="1222" color="1"/>
        <line x1="7446" y1="1222" x2="7489" y2="1220" color="1"/>
      </contour>
      <symbol x="7470" y="1261" color="0" text="53" width="5"/>
    </entity>
    <entity x="3351" y="3456" number="24" type="">
      <circle x="3351" y="3456" radius="26" width="5"/>
      <contour x="3351" y="3456" color="1" width="5">
        <line x1="2894" y1="2994" x2="2816" y2="3013" color="1"/>
        <line x1="2816" y1="3013" x2="2754" y2="3031" color="1"/>
        <line x1="2754" y1="3031" x2="2751" y2="3034" color="1"/>
        <line x1="2751" y1="3034" x2="2736" y2="3390" color="1"/>
        <line x1="2736" y1="3390" x2="2756" y2="3403" color="1"/>
        <line x1="2756" y1="3403" x2="2798" y2="3402" color="1"/>
        <line x1="2798" y1="3402" x2="2801" y2="3405" color="1"/>
        <line x1="2801" y1="3405" x2="2804" y2="3448" color="1"/>
        <line x1="2804" y1="3448" x2="2807" y2="3451" color="1"/>
        <line x1="2807" y1="3451" x2="2817" y2="3439" color="1"/>
        <line x1="2817" y1="3439" x2="2828" y2="3434" color="1"/>
        <line x1="2828" y1="3434" x2="2844" y2="3436" color="1"/>
      </contour>
    </entity>
  </processing>
</detecção>

```

Figura 7.3: A informação detectada é apresentada num ficheiro XML de modo estruturado. A detecção considera as entidades cadastrais com os seus atributos. De notar que a posição do círculo é a chave da entidade cadastral.

7.4 Interface com o Utilizador

A ligação de todos os módulos de reconhecimento permite dar ao sistema a forma de uma aplicação, capaz de interpretar as secções cadastrais. O reconhecimento de cada mapa

é gravado num ficheiro XML que apresenta todos os objectos detectados.

Tal como anteriormente referido, neste tipo de sistemas é muito importante visualizar a taxa de detecção, de modo a otimizar os métodos de análise. Assim, é necessário implementar um interface que sobreponha a informação detectada ao mapa original de modo a identificar potenciais problemas e analisar sobre a qualidade da detecção. Cada tipo de entidade está presente num determinado *layer* apresentando uma determinada cor consoante a sua classe.

O interface foi implementado na linguagem Python utilizando a livreria *wxPython* (www.wxpython.org) e o módulo *floatcanvas*, que apresenta a funcionalidade básica de visualização de um SIG, permitindo navegar no mapa e observar o resultado da detecção (ver figura 7.4). A comunicação entre o *backend* e o interface foi considerada, implementando-se um *wrapping* do backend desde C++ para Python através da ferramenta *SWIG*² [10, 20].

7.5 Exportação da Informação

A partir dos dados em XML é exportada a informação para um SIG³, com o objectivo de georeferenciar a informação e de a sobrepor ao mapa administrativo de Portugal⁴. A exportação da informação é realizada partindo do ficheiro em XML com os dados estruturados para um ficheiro no formato vectorial DGN⁵. Este processo foi implementado criando um script na linguagem Python [13, 105] e utilizando a biblioteca OGR que faz parte do

²Simplified Wrapper and Interface Generator (www.swig.org)

³Sistema de Informação Geográfico

⁴CAOP - Carta Administrativa Oficial de Portugal

⁵Ficheiro vectorial do software Microstation v7. De notar que também existe a possibilidade de exportar a informação no formato *Shapefile*, que se apresenta como o formato standard dos SIG, considerando o software ArcView da ESRI como o software mais utilizado neste dominio

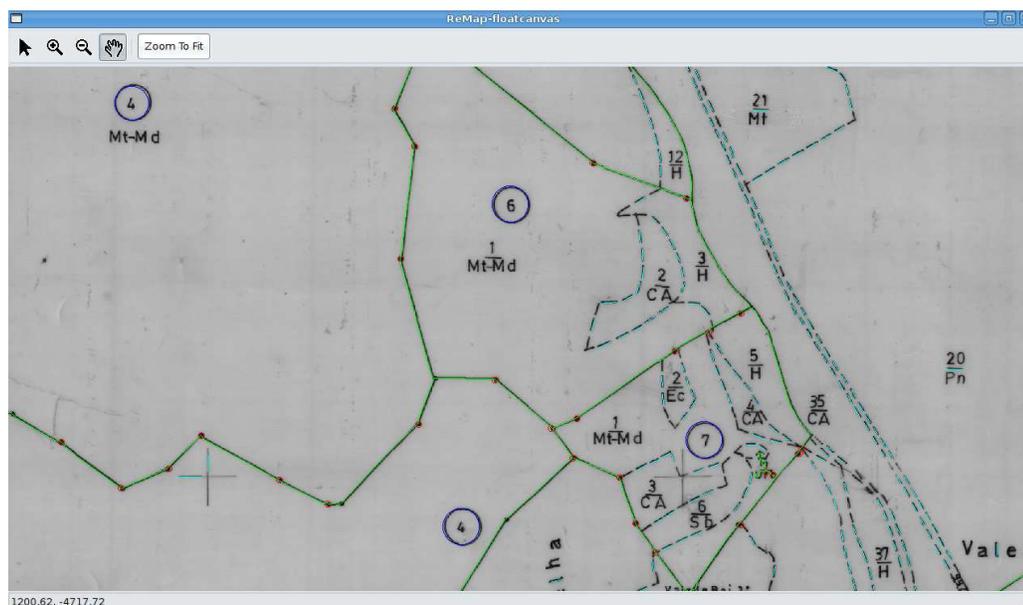


Figura 7.4: O interface apresenta a sobreposição da imagem cadastral com a informação detectada. Deste modo pode-se visualizar a detecção, realizar a validação e facilmente otimizar o desenvolvimento do sistema.

pacote GDAL ⁶, de modo a poder editar os dados num sistema de informação geográfico (ver figura 7.5).

A implementação é realizada considerando que os prédios estão numa *layer* de polígonos, sendo que cada polígono é uma *feature* e apresenta atributos, tais como o número do prédio, o número de parcelas, o código da secção, o *DICOFRE* ⁷, etc (ver figura 7.6). Outra possibilidade é utilizar uma *layer* de linhas e no final converter essa *layer* para o tipo polígono considerando para tal o conjunto de linhas que formam cada polígono.

⁶GDAL (<http://www.gdal.org/>), Geospatial Data Abstraction Library - é uma livreria que faz a tradução de vários formatos de dados georeferenciados.

⁷O *DICOFRE* é a concatenação do código do distrito, o código do concelho e o código da freguesia

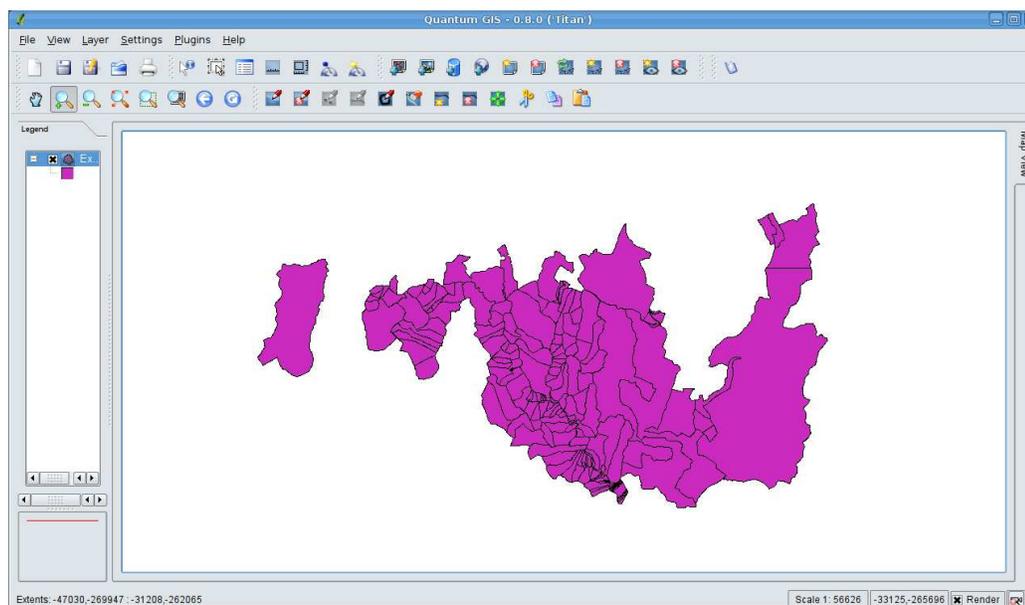


Figura 7.5: Esta imagem apresenta o resultado da vectorização integrado no sistema de informação geográfico QGIS. Os dados podem ser georeferenciados e sobrepostos à CAOP que contém os limites administrativos de Portugal. Deste modo a informação cadastral está pronta a ser utilizada para os mais diversos fins.

7.6 Conclusões

A integração apresenta dificuldades e é estudada a melhor maneira de estruturar o sistema de detecção. Neste domínio específico, foram consideradas estratégias com a finalidade de extrair a informação pretendida. O interface permite visualizar o resultado da detecção evidenciando a taxa de reconhecimento e a qualidade global da extracção. Finalmente a exportação da informação permite ao sistema disponibilizar os dados e realizar a sua validação, de modo a tornar a informação partilhada.

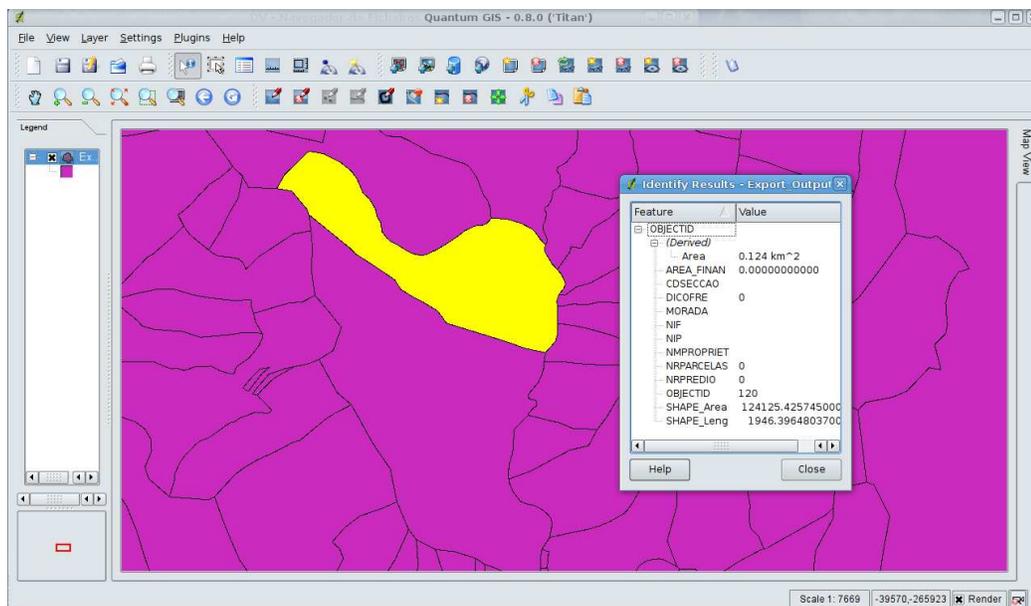


Figura 7.6: Nesta imagem estão evidenciados os atributos da entidade cadastral. É de notar que a área do prédio é atribuída automaticamente pelo SIG visto ser um *layer* de polígonos. Estes dados ao encontrarem-se colocados num sistema de informação permitem realizar consultas (*queries* à base de dados, de modo a obter a informação solicitada.

Capítulo 8

Resultados e Discussão

Este capítulo apresenta uma análise à amostra das secções cadastrais, que foram utilizadas para desenvolver o sistema protótipo e posteriormente faz inferência estatística dos resultados para todos os mapas portugueses. Os resultados de cada módulo são também apresentados, primeiro de um ponto de vista individual, de modo a evidenciar as dificuldades e os problemas do processo de desenvolvimento. E seguidamente, de um ponto de vista global, de modo a analisar a taxa de reconhecimento do sistema e o tempo economizado pela utilização do software, sendo no final realizada uma avaliação global ao sistema. Finalmente, é demonstrado os resultados da execução do projecto *Monchique* utilizando o sistema protótipo.

8.1 Amostra

Os algoritmos de reconhecimento de imagem foram inicialmente desenvolvidos considerando os problemas de detecção de um conjunto de 8 secções cadastrais ¹ gentilmente cedidas (não havendo hipótese de escolha). Estas secções são pouco significativas relativamente ao número existente de mapas cadastrais portugueses (aproximadamente 24 mil), no entanto este conjunto de secções cadastrais serviu como primeira amostra para o desenvolvimento do sistema, uma vez que contém uma enorme variedade de informação, permitindo assim estudar os problemas representativos dos mapas cadastrais portugueses. Estes mapas foram obtidos ainda no formato de papel (em tamanho A0) e posteriormente foram digitalizados obtendo-se imagens binárias em formato digital ². Em cada mapa existe um número significativo de problemas e os algoritmos desenvolvidos consideram todas essas dificuldades. Nesta fase foi desenvolvido um sistema protótipo que integra todos os métodos de análise para cada tipo de objecto gráfico existente neste tipo de mapas.

Depois da implementação do sistema protótipo, foram visitadas diversas entidades que executam a informatização do cadastro geométrico em Portugal, nomeadamente o Instituto Geográfico Português e seguidamente a Câmara Municipal de Monchique. Na visita ao IGP em Lisboa no ano de 2005, ficou patente a utilidade desta ferramenta em acelerar o processo da informatização do cadastro em Portugal. Responsáveis da execução cadastral do IGP informaram que poderia ser requerido uma amostra de mapas ao abrigo do programa FIGIEE ³, no entanto após a solicitação, o Instituto indeferiu o pedido alegando não

¹As secções cadastrais utilizadas para o desenvolvimento do sistema protótipo foram: 080503.t, 081101_o, 080301_n1, 080504_o, 080504_n, 080101_f, 080201_040, 080301_n2.

²Os mapas foram digitalizados para ficheiros de imagem no formato TIF (*Tagged Image File Format*). O scanner utilizado não apresentava a possibilidade de digitalizar em escala cinza, pelo que foram obtidas imagens binárias.

³Programa de Apoio ao Fornecimento de Informação Geográfica para Investigação, Ensino e Edição

poder disponibilizar devido ao facto da verba do fundo FIGIEE, destinada anualmente e por instituição, já ter sido utilizado pela Universidade do Algarve nesse presente ano. Com o objectivo de obter mapas para o estudo do sistema, foi visitada a Câmara de Monchique em 2005, tendo esta a amabilidade de ceder 7 secções cadastrais⁴. O sistema foi continuamente desenvolvido considerando mais estes mapas, contribuindo assim para o aumento da robustez do sistema.

Numa conferência em 2008 sobre Sistemas de Informação Geográfica, ocorrida na delegação do Algarve do Ministério da Agricultura, foi possível conversar com o Presidente do IGP, tendo sido esclarecido que poderia requerer os mapas, uma vez que o fornecimento era em função da pessoa e não da instituição. Deste modo, foi solicitado uma amostra de 50 secções cadastrais ao IGP, de modo a estudar a robustez do sistema a um conjunto de mapas pertencentes a todos os distritos com cadastro. Como resposta ao pedido, apenas foram gentilmente cedidas 14 secções no formato *raster*, de modo a estudar a robustez do sistema, mais 2 secções vectorizados como exemplo. Ficou assim evidenciada a dificuldade em aceder aos mapas cadastrais portugueses. O elevado valor monetário que o IGP requer por secção cadastral e a confidencialidade da informação dos mapas tornou a amostra pequena, porém tentou-se com que esta fosse o mais representativa possível.

Na tabela 8.1 são apresentados os detalhes das secções cadastrais utilizadas nesta tese. O primeiro bloco de 8 mapas, apresenta os detalhes dos mapas binários recolhidos inicialmente, seguido do segundo bloco de 7 mapas em escala cinza e finalmente um terceiro bloco de 14 secções com mapas coloridos e representativos da realidade cadastral em Portugal. Esta tabela apresenta todos os detalhes das secções cadastrais de modo a permitir que

⁴As secções cadastrais cedidas pela Câmara de Monchique para o aperfeiçoamento do sistema protótipo foram: 0809039_ef, 0809039_eg 0809039_eh, 0809039_ei, 0809039_ej, 0809039_el e 0809039_ef.

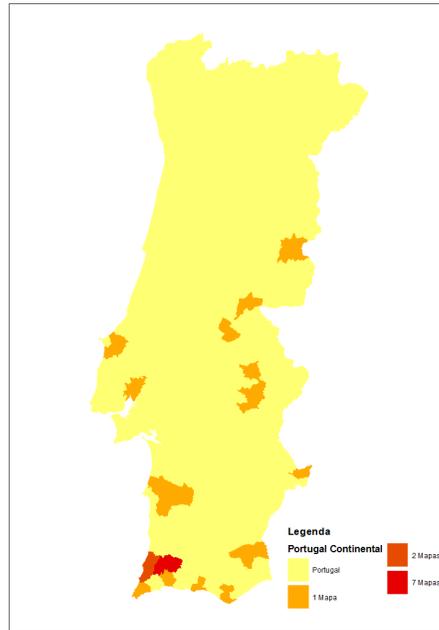


Figura 8.1: Esta figura apresenta a distribuição por concelhos das secções cadastrais presentes na tabela 8.1 no mapa administrativo de Portugal Continental.

qualquer outro sistema possa efectuar os mesmos testes de reconhecimento e assim obter uma comparação relativo ao sistema descrito. Na figura 8.1 pode observar-se a distribuição das secções cadastrais presentes na tabela 8.1.

8.1.1 Amostra das Secções Cadastrais

O cadastro geométrico da propriedade rústica foi desenvolvido em diferentes períodos de tempo, ao longo de cinco décadas (ver figura 8.2). No entanto, nesse intervalo de tempo, o CGPR não foi executado em alguns concelhos, sobretudo no norte de Portugal, devido à enorme densidade florestal dificultar o levantamento cadastral, segundo técnicos do IGP.

Consultando as secções cadastrais do CGPR, disponíveis no site oficial do IGP pode-se inferir que as secções cadastrais apresentam diferentes características consoante o seu

DICOFRE	Distrito	Concelho	Secção	Largura	Altura	Data
080503	Faro	Faro	t	10784	7853	4
081101	Faro	Portimão	o	10784	7860	4
080301	Faro	Aljezur	n1	10720	7794	4
080504	Faro	Faro	o	10720	7844	4
080504	Faro	Faro	n	10720	7947	4
080101	Faro	Albufeira	f	10720	7754	4
080201	Faro	Alcoutim	040	10720	8032	4
080301	Faro	Aljezur	n2	10720	7777	4
0809039	Faro	Monchique	ef	5572	7140	4
0809039	Faro	Monchique	eg	5511	7094	4
0809039	Faro	Monchique	eh	7123	5571	4
0809039	Faro	Monchique	ei	5603	7166	4
0809039	Faro	Monchique	ej	7091	5544	4
0809039	Faro	Monchique	el	7102	5550	4
0809039	Faro	Monchique	em	5561	7137	4
0204013	Beja	Barrancos	p	8529	6310	1
0507059	Castelo Branco	Penamacor	d	6626	5183	4
0511030	Castelo Branco	Vila Velha Rodão	au	6787	5294	4
0704059	Évora	Estremoz	a	7019	5204	2
0704091	Évora	Estremoz	c	6892	5528	2
0802034	Faro	Alcoutim	024	6714	5253	4
0815039	Faro	Vila do Bispo	q	5190	6588	4
1005022	Leiria	Bombarral	i	6547	5090	2
1012045	Leiria	Obidos	v	6665	5242	3
1108050	Lisboa	Lourinha	p	5213	6876	3
1114042	Lisboa	Vila Franca Xira	c	7185	5540	2
1208020	Portalegre	Fronteira	s	5229	6617	2
1209035	Portalegre	Gavião	o	6622	5226	3
1509063	Setúbal	Santiago Cacém	h	5054	734	1

Tabela 8.1: Esta tabela apresenta a amostra das secções cadastrais utilizadas, identificando o código *DICOFRE* (o *DICOFRE* é a concatenação dos códigos do distrito, concelho e freguesia) nome da secção, largura, altura da imagem e período de execução cadastral (o período está identificado na figura 8.2, sendo o valor 1 correspondente a 1944-1954, o valor 2 a 1955-1966, o valor 3 a 1967-1979 e o valor 4 a 1980-1995).

período de execução, por exemplo, no distrito de Beja, os mapas cadastrais apresentam-se completamente degradados, com informação sobreposta. Enquanto no distrito de Faro, a

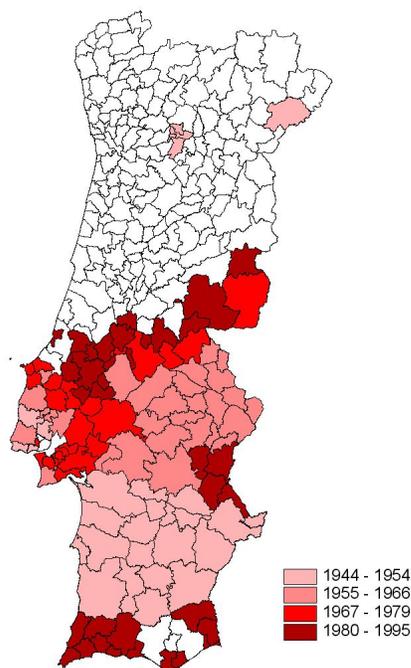


Figura 8.2: Esta imagem mostra os períodos de execução das secções cadastrais por concelho em função das áreas administrativas de Portugal continental.

grande maioria dos mapas apresenta uma melhor qualidade de imagem, sem degradação e sem informação sobreposta significativas. Observa-se que as secções cadastrais apresentam melhor qualidade nas regiões onde o cadastro foi executado no período entre 1980-1995 (ver figura 8.2). A variabilidade das secções cadastrais ao nível do desenho é insignificante entre as regiões, uma vez que as regras e normas que orientam a execução cadastral são idênticas.

As secções cadastrais apresentam-se distribuídas em Portugal Continental, tal como mostra a figura 8.3. Neste mapa, pode-se observar que as zonas a vermelho apresentam uma maior quantidade de secções, enquanto a amarelo uma menor quantidade. De notar, que o número de mapas cadastrais existentes em Portugal é cerca de 24 mil, mas apenas foi possível identificar a distribuição das 19667 secções cadastrais consideradas na figura 8.3

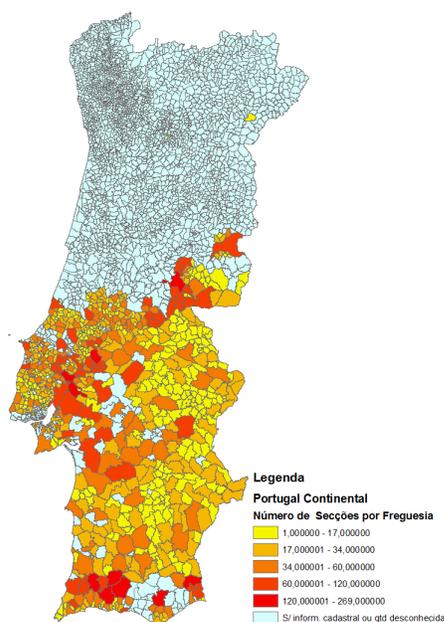


Figura 8.3: Esta imagem apresenta a distribuição por freguesia da quantidade de secções cadastrais, no mapa administrativo de Portugal Continental.

(estas imagens podem ser visualizadas directamente no site do IGP, em www.igeo.pt). A quantidade de mapas por distrito está também relacionada com a escala dos mapas, por existirem propriedades rústicas de maiores áreas e não necessitarem de tanto detalhe, sendo as secções executadas numa escala maior de 1:5000.

Considerando que o período de execução das secções cadastrais, está directamente relacionado com a qualidade da imagem dos mapas e esta com o nível de performance do prototipo desenvolvido, estratificou-se a população em 4 sub-populações, obtendo-se 4 sub-amostras tal como mostra a tabela 8.2. Embora limitado pelo número de mapas cedido pelo IGP, foram seleccionados para amostra estratificada secções de 4 períodos diferentes de execução cadastral em quantidade percentual igual à da população.

Limitado pelo fundo FIGIEE em 14 mapas, foram seleccionados para cada estrato da

Período	Nr Secções População	% População	Nr Secções Amostra
1944-1954	3005	15%	2
1955-1966	3749	19%	3
1967-1979	3992	20%	3
1980-1995	8921	45%	6
Total	19667	100%	14

Tabela 8.2: Esta tabela mostra a quantidade das secções cadastrais em função dos períodos de execução, calculando a sua percentagem relativamente ao total, de modo a encontrar uma sub-amostra que se aproxime da representação da população.

amostra a mesma quantidade percentual existente em cada estrato da população. Na tabela 8.3 pode-se visualizar a identificação das secções cadastrais presentes em cada estrato da amostra.

Tipo 1	Tipo 2	Tipo 3	Tipo 4
0204013_p	0704059_a	1012045_v	0507059_d
1509063_h	1005022_i	1108050_p	0511030_au
	1114042_c	1209035_o	0802034_024
			0815039_q
			0809039_ei
			0809039_ej

Tabela 8.3: Esta tabela identifica as nomenclaturas das secções presentes em cada estrato da amostra.

8.2 Análise dos Módulos

Neste trabalho, são sempre utilizadas todas as secções cadastrais para proceder ao estudo dos métodos de análise. Os algoritmos de detecção, são métodos genéricos que extraem informação das secções cadastrais baseando-se em características das imagens. No entanto, existem casos onde devido à natureza gráfica das secções, não é possível aplicar os algoritmos de detecção. Por exemplo, a secção *081101_o* apresenta características de

cadastro urbano, com elevado número de zonas completamente preenchidas com pixels pretos, não representando as características do cadastro geométrico rústico. Por outro lado, a secção *080301_n1* foi digitalizada a partir de uma folha de papel obtendo-se uma imagem binária, contudo a imagem original era colorida e assim apresenta linhas de água sobrepostas à informação cadastral, não sendo possível a segmentação desta camada. A secção *080101_f* apresenta em simultâneo a combinação das duas características anteriormente descritas.

Como a quantidade de mapas é reduzida, decidiu-se utilizar sempre todos mapas disponíveis, uma vez que mesmo que uma secção não se puder utilizar para a extracção de uma determinada *feature*, pode ser utilizada para a detecção de outra. Deste modo, nos casos onde não é possível aplicar os métodos de detecção, os resultados não são considerados e não aparecem na tabela de resultados e nem no gráfico respectivo.

A análise aos resultados obtidos pelos módulos de detecção permite estimar sobre a avaliação do sistema. Os resultados do processamento são observados individualmente para cada tipo de análise, com o objectivo de evidenciar os problemas e as soluções encontradas para cada método.

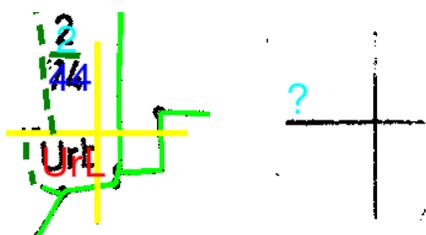
8.2.1 Cruzes Georeferenciadas

A detecção das cruzes georeferenciadas é o processo mais simples de detecção e não apresenta muitas dificuldades (ver figura 8.4(a)). No entanto, é necessário confrontar os problemas de cada método de reconhecimento com as imagens em estudo, de modo a analisar sobre as limitações do processo de detecção.

No processo de detecção de cruzes acontece um problema de reconhecimento, sempre

que existe uma região de pixels pretos muito densa. Nesta situação não há uma clara distinção entre o *foreground* e o *background*, e mesmo para um humano apenas é nítido que existe uma cruz considerando que duas cruces vizinhas se encontram sempre a uma distância constante, formando no seu conjunto uma grelha.

Também quando existem duas linhas perpendiculares, formando uma cruz mas não o sendo, é realizada uma falsa detecção. De um ponto de vista de reconhecimento, a detecção está correcta por se tratar de uma cruz, porém a detecção apenas é validada considerando a distância constante entre o conjunto das cruces.



(a) A detecção de uma cruz georeferenciada. (b) Uma cruz não detectada.

Figura 8.4: O exemplo de uma cruz bem detectada e outro de uma cruz não detectada devido ao efeito da degradação da imagem. Para ser correctamente detectada a cruz necessita de estar bem formada e de se distinguir do *background*.

Por outro lado, o ruído provoca a degradação da imagem levando a que algumas cruces percam a sua forma característica, introduzindo problemas na sua correcta detecção (ver figura 8.4(b)). Novamente, estes problemas são facilmente resolvidos considerando que todas as cruces têm uma distância e orientação constantes relativamente ao seu conjunto.

Através dos resultados das más detecções (falsos alarmes) e não detecções das cruces, é possível verificar que todas as cruces podem ser encontradas de um modo indirecto, considerando o conjunto das posições de todas as cruces e a sua distância constante entre

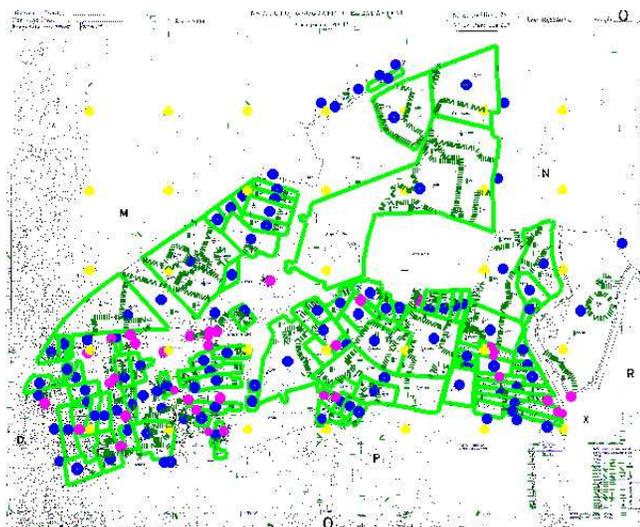


Figura 8.5: Nesta figura pode ser observada a detecção das cruzes georeferenciadas. A detecção destes elementos está representada com a cor amarela. Observar-se ainda que o conjunto das cruzes está disposto na forma de uma grelha.

cada cruz vizinha (ver figura 8.5).

De modo, a visualizar-se as taxas de detecção de um modo pormenorizado, foram criadas tabelas com as taxas de detecção das secções cadastrais. Estas tabelas estão presente no apêndice A, enquanto neste capítulo estão presentes os gráficos que resumem a performance da detecção.

A figura 8.6 apresenta um gráfico com as taxas de detecção e má detecção das cruzes em todas as secções cadastrais, tal como apresentado na tabela A.1. Através da observação do gráfico é possível apontar as secções problemáticas e identificar casos com cruzes graficamente mal definidas.

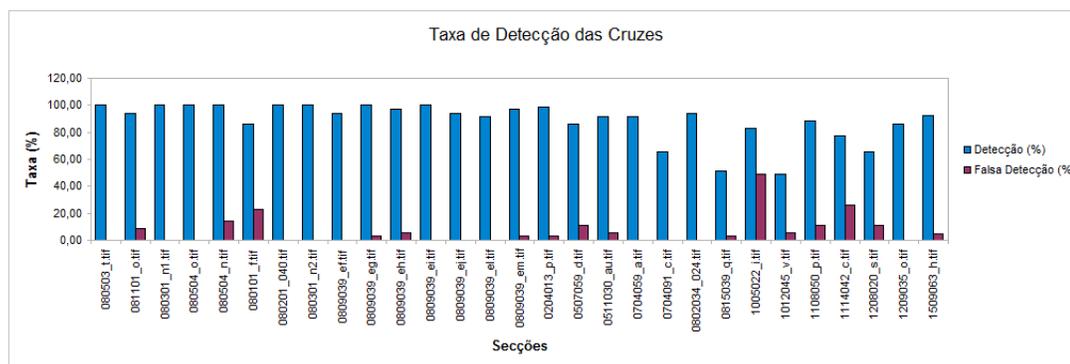


Figura 8.6: Esta imagem apresenta um gráfico de barras que mostra as taxas de detecção e de falsas detecções no processo de reconhecimento das cruzes.

8.2.2 Círculos

A detecção dos círculos é um processo que se apresenta consolidado, tendo o prototipo reconhecido a quase totalidade dos círculos nos mapas. Esta detecção é muito importante a nível global, uma vez que a extracção do contorno inicia-se na posição dos círculos detectados (ver figura 8.7).

Nesta detecção existem alguns problemas de nível prático, nomeadamente alguns caracteres são reconhecidos como círculos, por exemplo o caracter 'O', devido à sua forma circular e por outro lado, alguns círculos muito pequenos não são reconhecidos. O segundo problema é resolvido diminuindo o raio mínimo de procura, tendo como consequência o aumento do tempo de processamento. A detecção do caracter 'O' como círculo é normal, uma vez que o caracter tem a forma circular com um pouco de deformação, assim este é um problema semântico e é resolvido numa fase posterior do reconhecimento.

Assim, todos os círculos são na sua totalidade bem detectados (ver figura 8.8) obtendo em todos os mapas uma taxa de 100% ou muito próximo.

Como resumo desta detecção pode-se visualizar a figura 8.9, tal como apresentado na

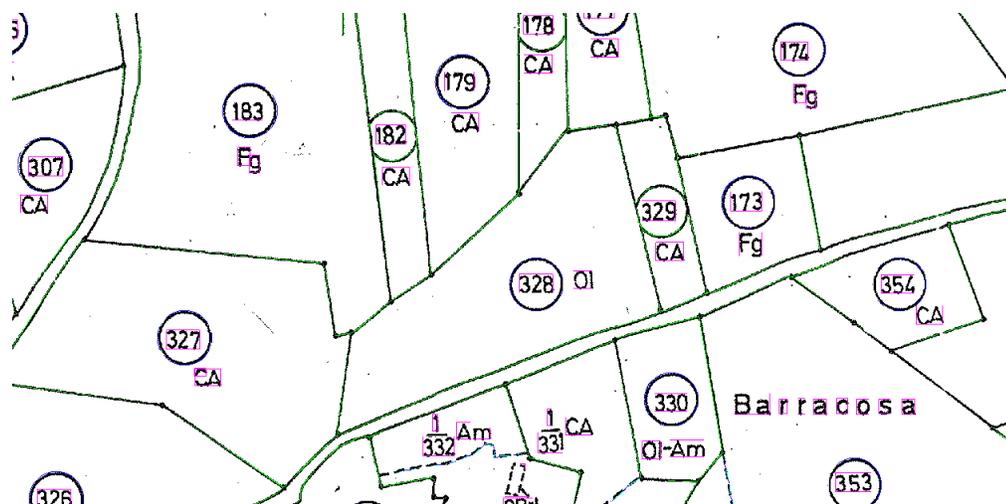


Figura 8.7: Exemplo da detecção dos círculos numa secção cadastral. Neste caso pode-se observar que os círculos apresentam uma forma elipsoidal e não a estrutura de círculos perfeitos.

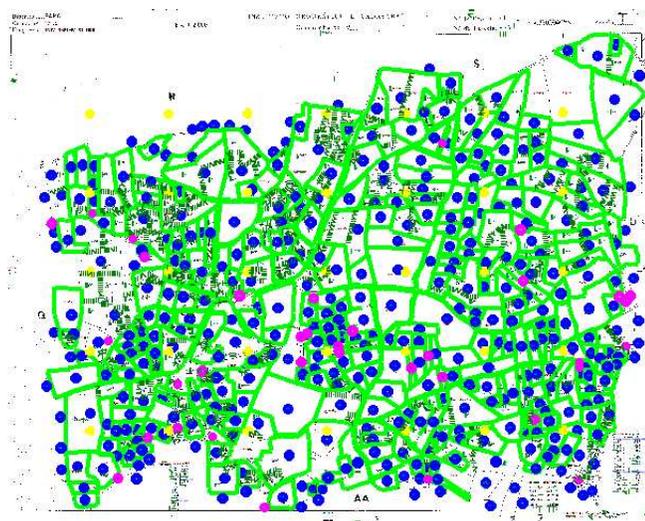
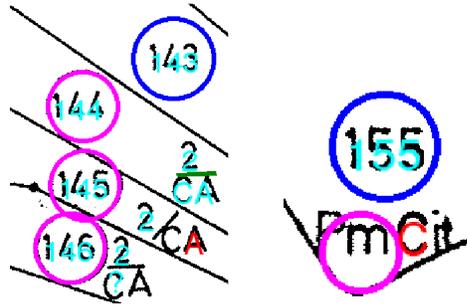


Figura 8.8: Um exemplo da detecção global dos círculos. Estes elementos foram detectados e os seus parâmetros de detecção estão representados a azul.

porque a imagem apresenta degradação o que introduz deformações na forma dos arcos. A consequência natural deste facto é uma dificuldade acrescida na detecção e eliminação do arco da imagem, após a sua detecção.



(a) Detecção dos arcos em prédios com áreas reduzidas. (b) Má detecção de um arco sobre um contorno de prédio que apresenta uma forma curva.

Figura 8.10: Duas figuras representativas dos problemas na detecção dos arcos.

Por outro lado, quando o prédio apresenta uma área muito pequena, o círculo encontra-se partido ao meio formando dois pequenos arcos (ver figura 8.10(a) - prédio com o número 145). Acontece muitas vezes que a posição do centro destes dois arcos apresenta um desfasamento, introduzindo assim mais dificuldades na detecção.

Quando os arcos ainda estão conexos ao contorno do prédio, a detecção deve considerar esta situação e dar alguma margem de tolerância ao encontrar pixels a preto fora da trajectória do arco. Os arcos que não se apresentam conexos ao contorno são detectados com maior probabilidade (ver figura 8.11).

Ao considerar todas estas dificuldades pode-se afirmar que a detecção dos arcos é um

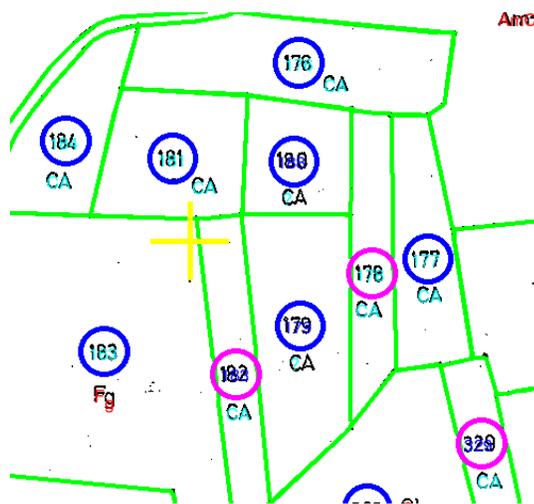


Figura 8.11: A detecção dos arcos está representada com círculos cor de rosa, que se apresentam conexos ao contorno do prédio.

processo muito sensível. Por um lado, se for dada margem de tolerância de pixels a preto, surgem as más detecções (ver figura 8.10(b)), enquanto se a detecção for mais rígida, não são detectados todos os casos. Está assim evidenciado o compromisso entre uma detecção rígida mas segura e uma detecção mais dinâmica mas com maior probabilidade de erro.

A tabela A.3, presente no apêndice A, apresenta os resultados da detecção dos arcos nas secções cadastrais estudadas. De notar, que no caso dos arcos, o tempo de processamento aumenta comparativamente ao dos círculos, devido aos primeiros possuírem maior número de *features* para analisar.

Esta tabela mostra que algumas secções apresentam elevadas taxas de más detecções, devido aos factos de existirem poucos arcos nos mapas e da quantidade dos arcos mal detectados ser superior aos dos bem detectados. A detecção dos arcos é um processo muito sensível, onde é necessário dar bastante tolerância para que se verifique. Contudo, ao permitir essa tolerância é obtida uma elevada taxa de más detecções. Os resultados com uma elevada taxa de má detecção não são contabilizados na média final, porque não é

aplicável no caso da secção *081101_o* e nos outros casos a taxa de más detecções não deve ser sobrevalorizada, em virtude de não existir uma quantidade considerável de arcos.

Na figura 8.12 pode-se observar os resultados obtidos na detecção dos arcos. A detecção apresenta uma performance inferior relativamente às cruzes e aos círculos. Esta situação deve-se principalmente aos arcos apresentarem menos pontos e uma fraca taxa de *macth* com o traçado do arco teórico.

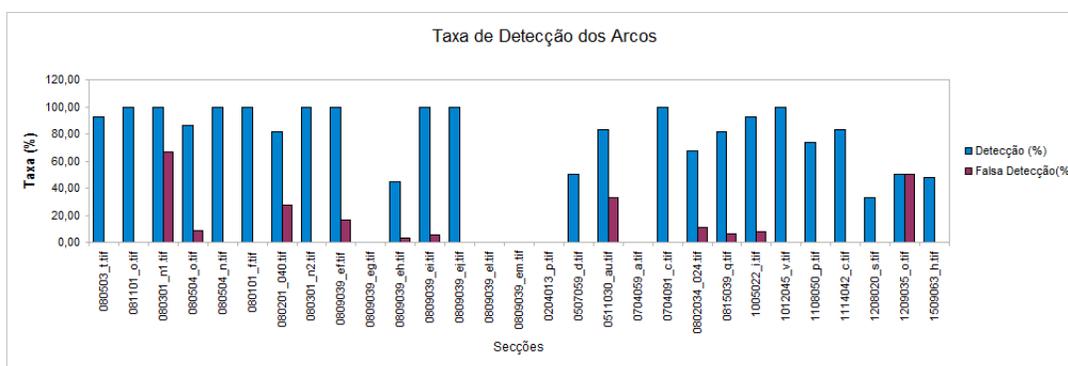


Figura 8.12: Esta imagem apresenta os níveis de detecção e de falsos alarmes dos arcos. De notar que nos casos onde não está definida a percentagem de detecção deve-se a não existirem arcos nessas secções cadastrais ou a não terem sido considerados (ver tabela A.3).

8.2.4 Linhas a Tracejado

A detecção das linhas a tracejado é efectuada em dois passos. Em primeiro lugar é considerada a detecção individual dos tracejados elementares e só depois é realizada a junção entre tracejados vizinhos, de modo a formar linhas contínuas que representem as linhas a tracejado. Esta detecção apresenta várias dificuldades, que tornam a detecção num desafio interessante.

Uma vez que as linhas a tracejado não podem ser reconhecidas como um todo, são primeiro reconhecidos os tracejados elementares. A detecção dos tracejados elementares com

forma rectangular e completamente preenchida não apresenta qualquer problema, sendo identificados, em regra geral, todos os elementos (ver figura 8.13).

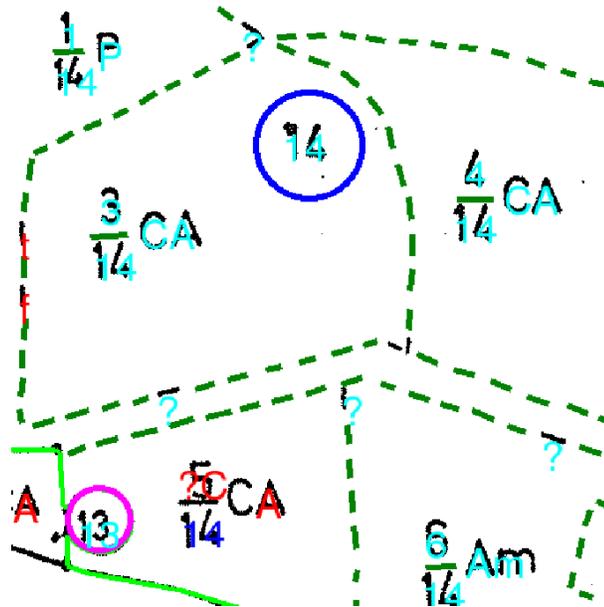


Figura 8.13: Esta figura mostra um exemplo da detecção dos tracejados elementares. Esta detecção apresenta alguns traçados não detectados (que podem ser observados onde está o sinal de interrogação) nomeadamente no caso das junções e dos tracejados com forma de curva ou que não formam um rectângulo completamente preenchido.

A detecção dos tracejados conexos ao contorno é também realizada sem problemas, considerando o contorno como um todo e procurando segmentos de recta apenas com uma extremidade ligada. Posteriormente, é realizada a validação considerando que a extremidade ligada tem que ter um tracejado elementar vizinho próximo.

Os tracejados elementares que são formados por curvas não são detectados como tracejados elementares, no entanto quando apresentam apenas duas extremidades são facilmente detectáveis. Caso os tracejados apresentem uma curvatura e mais de duas extremidades, então, estamos a considerar o caso das junções (ver figura 8.14).

Os tracejados com um comprimento muito pequeno não são detectados, uma vez que

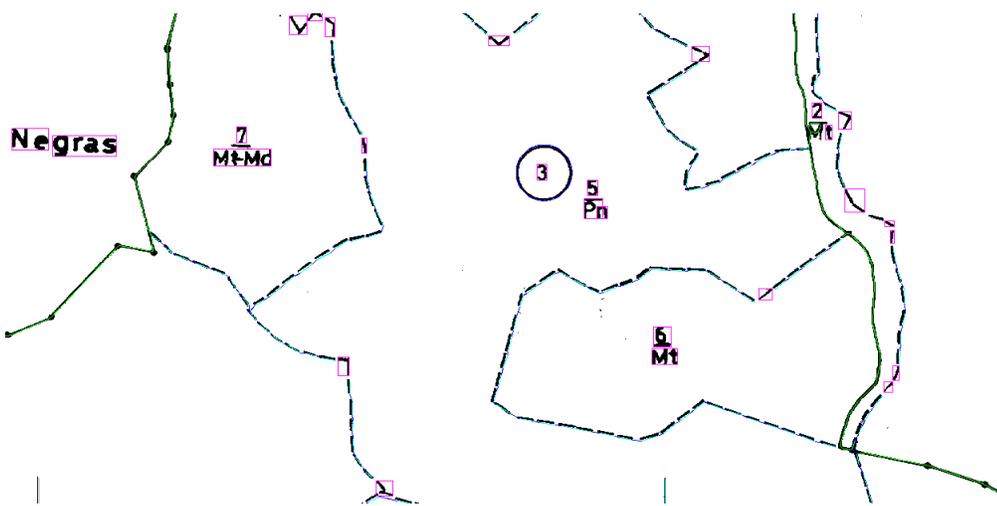


Figura 8.14: Esta figura apresenta a visão geral sobre o processo de ligação dos tracejados elementares. De notar que os tracejados elementares não reconhecidos apresentam-se dentro de um rectângulo.

não apresentam a forma de tracejado. Estes elementos parecem-se com o acento da letra 'i'. Para detectar estes tipo de casos apenas é possível considerando a sua vizinhança, o que torna o método menos robusto.

Por outro lado, os tracejados elementares que estão conexos com outros tracejados elementares e apresentam mais do que duas extremidades designam-se por junções e não apresentando a forma rectangular. Para realizar esta detecção é necessário decompor a junção em tracejados elementares e posteriormente verificar a potencial ligação de cada extremidade não ligada a cada tracejado elementar vizinho, de modo a validar a junção candidata. O processo de detecção das junções não foi implementada a nível prático, devido às limitações de tempo no desenvolvimento deste projecto, aparecendo cada junção como um elemento não detectado.

Os tracejados elementares são ligados considerando cada extremidades não ligada. Para cada tracejado elementar são encontrados os elementos com a extremidade mais próxima

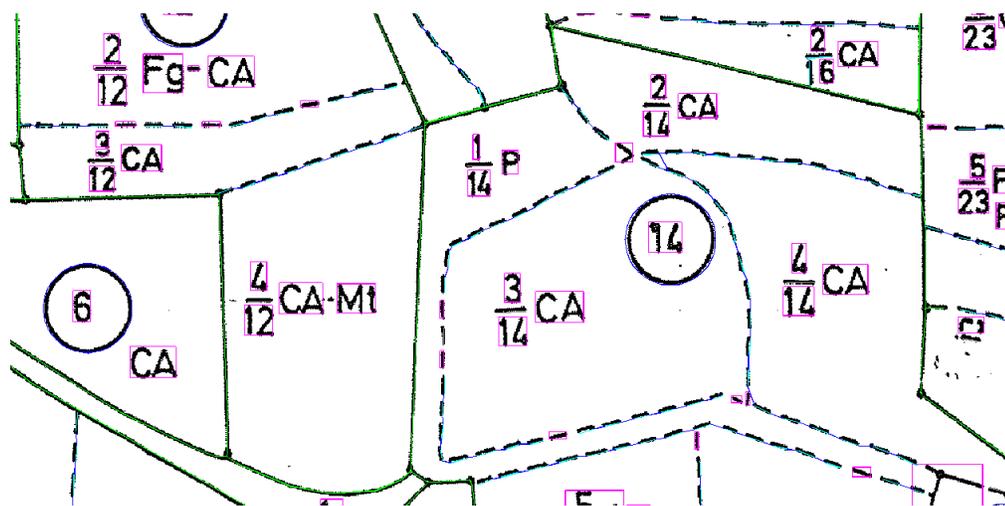


Figura 8.15: Nesta imagem pode observar-se as dificuldades da ligação dos tracejados elementares. Os casos problemáticos acontecem na ligação das junções e dos tracejados elementares não detectados.

e com uma orientação semelhante à sua. A implementação da ligação dos tracejados elementares encontra-se realizada, mas os resultados ainda não são robustos (ver figura 8.15). O módulo reconhece correctamente os tracejados elementares, mas quando existem vários tracejados elementares muito próximos, em linhas a tracejado paralelas, a decisão de qual o tracejado a ligar, ainda carece de mais análise na implementação prática. Contudo, nos casos de uma linha a tracejado simples, o método apresenta muito bons resultados.

A tabela A.4 mostra os níveis obtidos na detecção dos tracejados elementares. A taxa de detecção é boa e pode ainda ser melhorada, nos casos dos tracejados com forma curva e nas junções, considerando a vizinhança dos tracejados detectados.

A figura 8.16 apresenta os resultados descritos na tabela A.4. Neste caso, verificar-se diferentes taxas para os mapas binários, com escala cinza e coloridos, sendo estes últimos aqueles que apresentam valores mais baixos. Estes fracos resultados são devida às secções coloridas necessitarem de uma primeira segmentação baseada na cor que origina um ligeiro

emagrecimento dos tracejados e dá-lhes uma aparência curva.

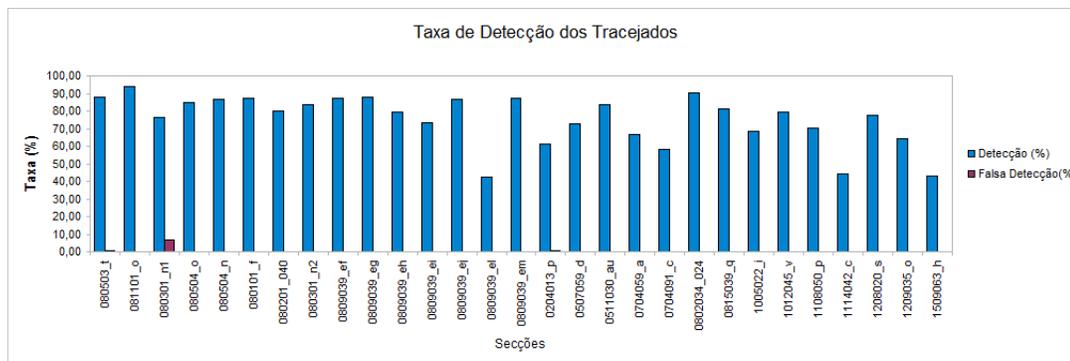


Figura 8.16: Esta imagem apresenta um gráfico de barras que mostra as taxas de detecção dos tracejados.

8.2.5 Marcos

A detecção dos marcos está baseada no facto destes elementos, que se apresentam sobre as extremas de prédio, serem formados por pequenos círculos completamente preenchidos e se distinguirem visualmente, de uma forma bem legível, das extremas de prédio. A figura 8.17 mostra uma boa detecção dos marcos, devido a estes se apresentarem bem evidenciados sobre o contorno.

No caso de algumas destas condições falhar, pode a taxa de detecção diminuir. Quando os marcos não se distinguem claramente do contorno, i.e. a largura de linha do contorno é aproximadamente igual aos marcos, o processo de convulsão não segmenta convenientemente os marcos, originando más detecções, sendo a detecção impraticável (ver figuras 8.18 e 8.19). Este problema também acontece quando o contorno apresenta uma grande largura de linha, como é o caso das zonas onde se unem várias linhas no contorno.

A secção *080301_n1* (ver figura 8.18) apresenta zonas do contorno com dilatações e

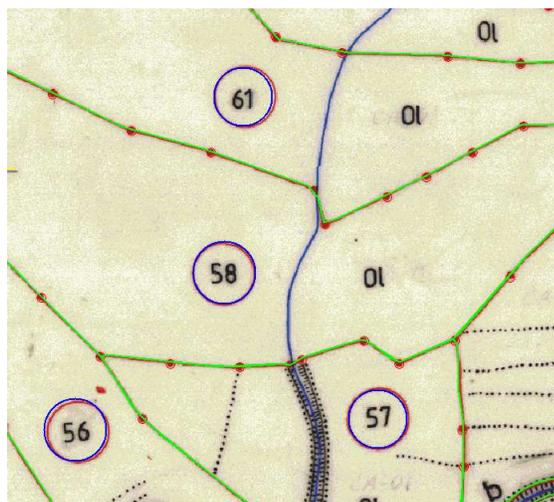


Figura 8.17: Esta imagem apresenta uma boa detecção dos marcos.



Figura 8.18: Esta imagem apresenta a detecção dos marcos na secção 080301_n1. De notar que são detectados dois marcos muito próximos devido à dilatação do contorno nalgumas zonas.

outras zonas com pequenos círculos completamente preenchidos mas não sobre o contorno. Noutra situação, a secção 081101_o (ver figura 8.19) apresenta traços conexos ao contorno que aparentam ser marcos. Nestes dois casos não é aplicável a detecção dos marcos, uma vez estas imagens apresentarem características excepcionais.

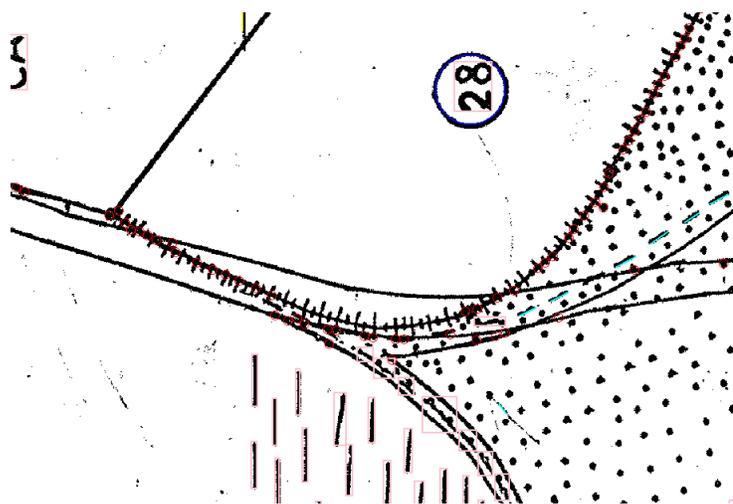


Figura 8.19: Esta imagem apresenta a detecção dos marcos na secção 081101_o. Neste caso a detecção dos marcos não é aplicável devido às características excepcionais desta secção.

Nos mapas coloridos, existem alguns casos onde o contorno com a cor vermelha apresenta no seu interior marcos com a mesma cor mas de tonalidade mais escura. Como o contorno é segmentado pela cor vermelha, perde-se a informação relativa aos marcos e estes não são detectados (ver figura 8.20). Deste modo, verifica-se que a aplicação desta detecção está condicionada pelo não cumprimento das convenções gráficas por parte de quem desenhou o mapa.

A tabela A.5 mostra os resultados obtidos na detecção dos marcos sobre o contorno. A detecção dos marcos não é aplicável nas secções 080101_f e 0704091_c, devido à fraca distinção dos marcos sobre o contorno e à cor excessivamente vermelha escura dos marcos, respectivamente. Foram também obtidos, alguns falsos alarmes noutras secções devido à dilatação do contorno ou devido às características excepcionais desses mapas, tal como descrito anteriormente.

A figura 8.21 apresenta os resultados mostrados na tabela A.5. Nesta figura algumas

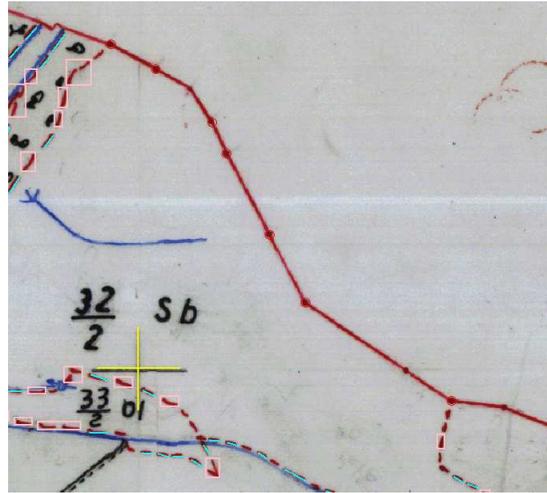


Figura 8.20: Esta imagem apresenta a detecção dos marcos na secção 1209035_o. De notar que sobre o contorno aparecem marcos com a cor vermelha escura e que depois da segmentação baseada na cor vermelha padrão, a cor de dentro do marco é perdida, por se apresentar muito próxima da cor preta. Assim, os marcos não são detectados.

secções não são consideradas, mas é possível observar-se que os resultados são bons caso os marcos se evidenciem sobre o contorno.

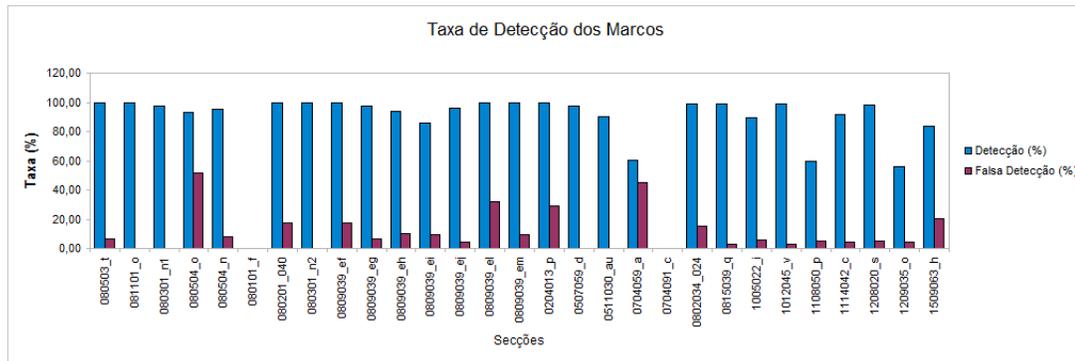


Figura 8.21: Esta imagem apresenta um gráfico de barras que mostra as taxas de detecção dos marcos.

8.2.6 Caracteres

O reconhecimento de caracteres nos mapas cadastrais apresentam simultaneamente factores que facilitam e dificultam o processo de análise. No geral, pode-se considerar que esta detecção é bem sucedida, uma vez que são detectados todos os caracteres e reconhecidos na sua maioria (ver figura 8.22(a)).

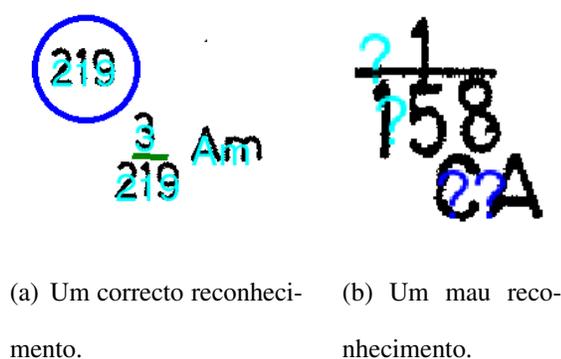


Figura 8.22: Exemplos do reconhecimento de caracteres.

Os mapas cadastrais apresentam factores que tornam a detecção e o reconhecimento dos caracteres num processo muito fácil. Nomeadamente o conjunto de caracteres que forma a palavra apresenta a mesma orientação, o universo de palavras ser conhecido, podendo deste modo se implementar um dicionário para realizar a validação, e o facto dos caracteres serem desenhados com escantilhão, existindo num mapa pouca variação.

Os mapas cadastrais podem apresentar os caracteres com diferentes orientações de mapa para mapa, no entanto essa orientação é a mesma para todos os caracteres do mesmo mapa. Este facto, torna a detecção de caracteres bastante fiável, uma vez que a detecção e agrupamento dos caracteres considera sempre o alinhamento nos eixos horizontal e vertical.

Os caracteres apresentam um tamanho normalizado. Quando existem dois caracteres

conexos, realiza-se uma segmentação considerando um tamanho fixo. No entanto, esta análise não é assim tão fácil se considerarmos o facto das letras de largura pequena poderem estar conexas com letras de maior tamanho. Deste modo, é necessário tomar decisões e os erros podem suceder-se.

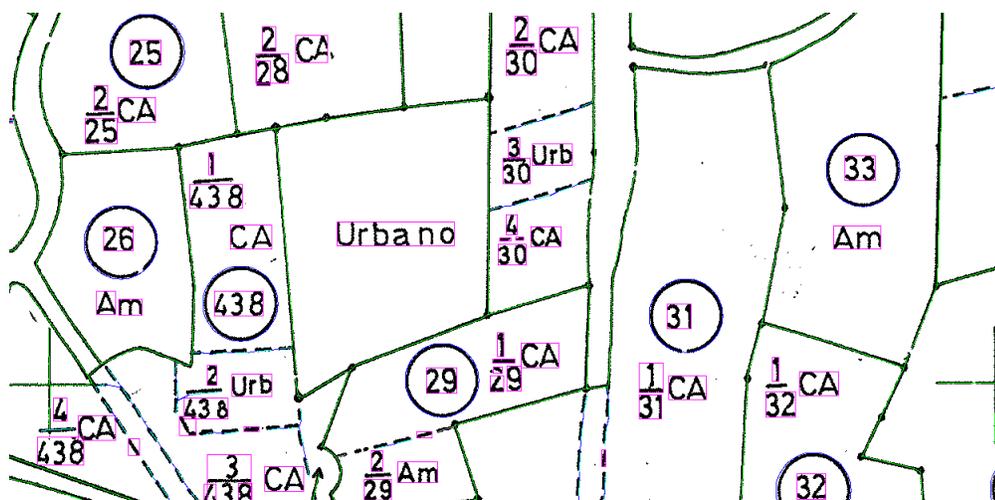


Figura 8.23: Nesta figura é apresentado o processo de agrupamento de caracteres, sendo os caracteres agrupados nos eixos vertical e horizontal, tendo em consideração a vizinhança de cada um.

Acresce a dificuldade das letras poderem estar conexas ao traço de fracção (ver figuras 8.22(b) e 8.23). Neste caso, o processo de segmentação torna-se mais complexo e a remoção do sinal de fracção origina novas deformações em cada caracter conexo.

Quando as letras se encontram conexas ao contorno surgem novos problemas e caso o contorno apresente alguma curvatura, podem ocorrer novas deformações. O processo de resolução é complexo e de um ponto de vista prático, pode não justificar uma nova solução. Tal como no caso dos arcos, quando vários elementos se encontram conexas, os erros de detecção são consideráveis e mesmo que alguns arcos sejam detectados, a eliminação origina novos problemas em posteriores detecções. Assim, como este problema é complexo,

necessária de uma demorada análise e a taxa de reconhecimento não seria motivação suficiente para esta implementação.

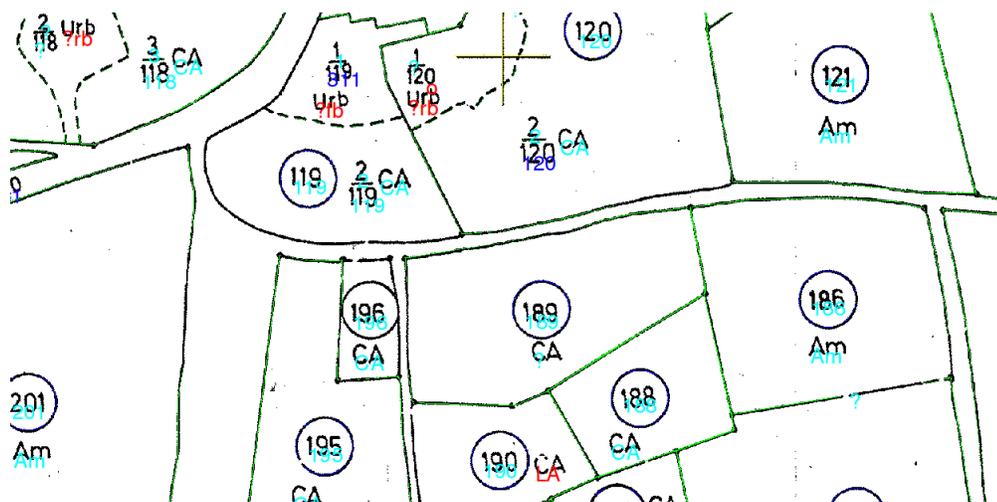


Figura 8.24: Exemplo do reconhecimento e classificação dos caracteres.

O reconhecimento dos caracteres (ver figura 8.24), ao contrário dos outros métodos de análise, não pode ser visto como um método único que reconhece todos os caracteres em todos os mapas. Uma vez que existem muitas variâncias, se considerarmos por um lado, o tipo, o tamanho e a rotação da letra e por outro lado, o ruído e a degradação da imagem. Assim, de um ponto de vista prático, pode ser implementável um método que detecte os caracteres, os agrupe em palavras, os reconheça (classifique) e realize a sua validação através de um dicionário. A classificação pode utilizar um método de análise, tal como o *template matching* ou *zoning*, mas considerando que o formato dos caracteres pode variar de mapa para mapa, devido ao uso de diferentes escantilhões, parece mais sensato utilizarem-se métodos iterativos, em que o utilizador classifica um conjunto de caracteres (aprendizagem *off-line*) e posteriormente o sistema classifica os restantes de modo automático, agrupando os elementos em classes de palavras e números.

A secção 0704091_c apresenta uma elevada taxa de ruído com uma enorme quantidade de caracteres com características distintas entre si, não apresentando diferenças significativas que pudessem ser facilmente agrupadas em classes. Este facto deve-se aos números terem sido escritos à mão, ao invés do uso de um escantilhão, existindo assim sempre muitas diferenças entre caracteres da mesma classe, o que complica o treino.

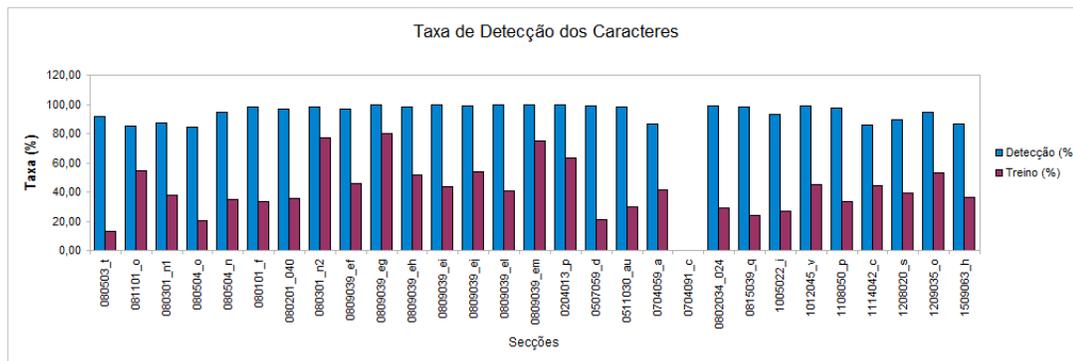


Figura 8.25: Esta imagem apresenta um gráfico de barras que mostra a taxa de detecção dos caracteres e a taxa de caracteres utilizados no treino do sistema.

8.2.7 Contorno

A extracção do contorno é realizada em duas partes distintas, primeiro são identificados os pontos do contorno e depois são vectorizadas as linhas formadas por esses mesmos pontos (ver figura 8.26), de modo a obterem-se segmentos de recta representativos do limite dos prédios. Esta extracção apresenta dificuldades ao nível da identificação dos pontos do contorno nos casos em que este apresenta elementos conexos. Por outro lado, na vectorização é necessário também considerar os marcos sobre o contorno do prédio, de modo a minimizar os segmentos de recta representativos do contorno.

Mais uma vez e tal como referido anteriormente, esta abordagem pode ainda ser me-

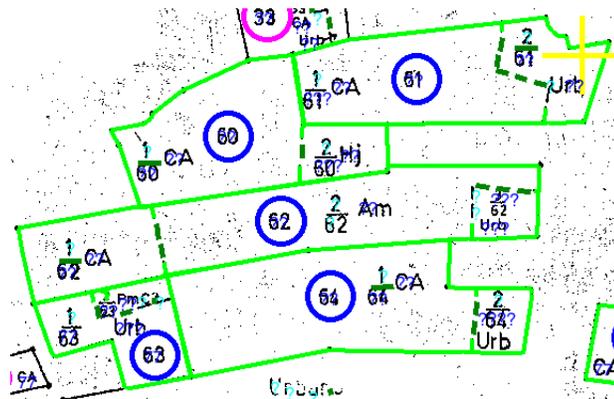


Figura 8.26: Esta figura mostra a vectorização do contorno dos prédios.

lhorada, de modo a se considerarem mais problemas de reconhecimento. Por exemplo, a extracção dos contornos parte da posição do círculo ou do arco e isso poderia ser ultrapassado considerando a extracção sem uma prévia detecção dos círculos, com o objectivo de evitar dependências. Uma ideia para segmentar os mapas em zonas de prédio é utilizar a transformada da distância e detectar os gradientes máximos, que são os pontos centrais mais afastados dos limites do prédio. Essas posições seriam o ponto de partida do algoritmo de extracção, mas como se verifica existe sempre decisão e erro.

A extracção dos contornos apresenta problemas no caso dos prédio com uma área muito reduzida (ver figura 8.27), onde a informação encontra-se sobreposta ao contorno. Como neste caso, existem elementos que estão conexos, a extracção dos pontos do contorno não é feita da melhor forma, apresentando erros. Assim, uma tentativa de resolução representaria uma investigação extremamente detalhada e conseqüentemente demorada.

Na tabela A.7 observar-se que existem secções onde a extracção do contorno não é aplicável. Este facto é devido a algumas secções apresentarem prédios incompletos que têm continuidade noutras secções, encontrando-se abertos. Ou por outro lado, no caso dos

reconhecimento de cada módulo, podendo assim analisarem-se os resultados de um ponto de vista global.

A partir dos dados das performance individuais de cada mapa foi elaborada a tabela 8.4.

Sub-Amostras	Tipo 1	Tipo 2	Tipo 3	Tipo 4	Total (%)
Cruzes	95,46	83,81	74,28	86,19	84,94
Círculos	100,00	100,00	100,00	100,00	100,00
Arcos	48,00	87,82	74,56	80,33	72,68
Tracejados	52,23	59,60	71,47	81,33	66,16
Marcos	91,94	80,45	71,49	94,50	84,59
Caracteres	93,44	88,46	97,13	98,86	94,47
Contorno	0,00	35,06	57,85	74,20	41,78
Total	68,72	76,46	78,11	87,91	77,80

Tabela 8.4: Tabela de dupla entrada, que apresenta as médias das taxas de detecção.

Análise do Reconhecimento por Módulo

A detecção das cruzes apresenta uma taxa média de reconhecimento elevada e não depende da idade do mapa, mas sim da grossura das linhas da cruz.

A detecção dos círculos apresenta um resultado excelente. Em todos os tipos de mapa a média das taxas é de 100%.

A detecção dos arcos apresenta um resultado bastante satisfatório. A performance parece não depender da idade do mapa. É nos mapas tipo 2 que a taxa de detecção é mais elevada, uma vez que estes detêm as melhores condições para a sua extracção, como a não sobreposição de informação.

A detecção dos tracejados apresenta uma taxa satisfatória. A performance deste módulo depende muito da idade do mapa e consequentemente da qualidade da imagem. Assim, embora nos mapas tipo 1 detecte apenas 52,23%, nos mapas tipo 4 a média da taxa aumenta

consideravelmente para 81,33%.

A detecção dos marcos apresenta uma taxa bastante satisfatória. A performance do módulo de detecção não depende só da idade dos mapas, uma vez que as secções do tipo 1, as mais antigas, detêm uma taxa de reconhecimento muito elevada, quase idêntica à dos tipo 4. Sendo os mapas do tipo 3 aqueles que apresentam um nível de detecção mais baixo, devido aos marcos não se evidenciarem significativamente do contorno. Assim, o nível baixo de detecção é devido ao modo excepcional como foi elaborado o desenho.

A detecção dos caracteres apresenta uma taxa muito boa. A performance deste módulo depende sobretudo da qualidade de imagem. Contudo, a taxa de detecção das secções do tipo 2 é mais baixa que a do tipo 1. Ao analisar estas secções tipo 2, constata-se que existem muitas conexos e alguns caracteres manuscritos.

A detecção do contorno apresenta uma taxa média reduzida. Analisando estes resultados pode-se concluir que a performance deste módulo depende bastante da idade dos mapas e consequentemente da qualidade da imagem. Assim, para mapas do tipo 1 o software é ineficaz, porém, para mapas do tipo 4 a sua performance é bastante satisfatória.

Análise do Reconhecimento por Tipo de Mapa

A análise da performance baseada no tipo de mapa mostra que a taxa de detecção aumenta inversamente à idade do mapa. Assim, para mapas do tipo 4, com melhor qualidade de imagem a média da taxa de performance ascende 87,91%.

Análise das Falsas Detecções

A partir dos dados dos falsos alarmes ou falsa detecção, de cada mapa, foi elaborada a tabela 8.5.

Sub-Amostras	Tipo 1	Tipo 2	Tipo 3	Tipo 4	Total (%)
Cruzes	4,09	24,76	5,71	3,33	9,47
Círculos	0,00	7,41	7,14	0,00	3,64
Arcos	0,00	3,85	16,67	38,22	14,68
Tracejados	0,43	0,13	0,00	0,17	0,16
Marcos	24,73	18,33	3,95	5,35	13,09
Caracteres	6,57	11,54	2,87	1,14	5,53
Contorno		4,36	0,95	3,29	2,86
Total	5,97	10,05	5,33	7,34	7,06

Tabela 8.5: Tabela de dupla entrada, que apresenta as médias dos falsos alarmes na detecção.

Sendo o objectivo do presente trabalho, a construção de um software capaz de reconhecer informação contida nas secções cadastrais portuguesas, possibilitando a automatização de uma tarefa altamente morosa e repetitiva, é necessário analisar não só o que é bem detectado, que se traduz num ganho de tempo, como também o que é mal detectado, originando perda de tempo. Assim, se na detecção de um mapa, um módulo de análise possuir elevadas taxas tanto de reconhecimento como de falso alarme, deve-se equacionar o uso deste módulo.

Análise de Falsas Detecções por Módulo

O módulo de detecção dos tracejados apresenta um excelente resultado. Os módulos das cruzes, círculos, caracteres e contornos apresentam resultados muito bons com a média da taxa de falsas detecções bastante reduzida.

A taxa de falsa detecção dos módulos dos arcos e marcos apresentam valores médios de 14,68% e 13,09%, respectivamente.

Análise de Falsas Detecções por Tipo de Mapa

A ocorrência de falsas detecções em nada têm haver com a idade do mapa. Os falsos alarmes sucedem devido às características do desenho dos mapas. Estes problemas já foram descritos.

8.3 Projecto Executado

O processo de informatização do cadastro rústico é realizado por técnicos do IGP ou das Câmaras Municipais, sendo a vectorização a actividade mais morosa de todo o processo. Os técnicos concretizam-na de modo manual e sem qualquer automatismo, necessitando em média de oito horas para a execução de cada secção cadastral, segundo fontes do IGP. Este trabalho é realizado geralmente através do software *MicroStation* ou de um sistema de informação geográfico, tais como o *ArcGis* ou o *Geomedia*.

Após a conclusão do sistema prototipo, foi executado o projecto *Monchique* de modo a comparar o método automático com o manual. A execução deste projecto teve como objectivo, o estudo da viabilidade do sistema desenvolvido, para realizar a exportação da informação a partir das secções cadastrais para um sistema de informação geográfica (SIG).

Foram realizadas detecções automáticas e exportada a vectorização automática das estremas de prédio para um ficheiro vectorial no formato *DGN* (*DGN Microstation v7*), para posterior correcção e validação. Foram também realizadas vectorizações manuais dos mesmos mapas, de modo a comparar o tempo de correcção do *output* da vectorização au-



Figura 8.31: Esta imagem apresenta a sobreposição de várias secções cadastrais, com uma camada de polígonos sobreposta, que representa as extremas de prédios.

A figura 8.32 apresenta uma consulta ao SIG. Este projecto de informação geográfico foi realizado utilizando uma camada de polígonos e definindo um conjunto de atributos para cada polígono. Os dados vectoriais (polígonos) são assim associados à informação alfanumérica (atributos), de modo a obter informação relativa a cada prédio. Assim, pode observar-se na figura 8.32 os atributos relativos ao prédio seleccionado.

A figura 8.33 apresenta o conjunto de secções cadastrais georeferenciadas e sobrepostas ao mapa administrativo do Algarve. Esta sobreposição identifica toda a funcionalidade da informação cadastral, nomeadamente a identificação directa dos proprietários de cada terreno e o cálculo automático das áreas. Adicionalmente, uma nova camada com imagens satélite poder-se-ia observar a sobreposição das propriedades à fotografia real.

8.3.2 Tempo de Execução

Na realização do projecto *Monchique* foi contabilizado os seus tempos de execução. Através da comparação dos métodos manuais e do método automático com correcção, pode observar-se os tempos de execução na tabela 8.6.

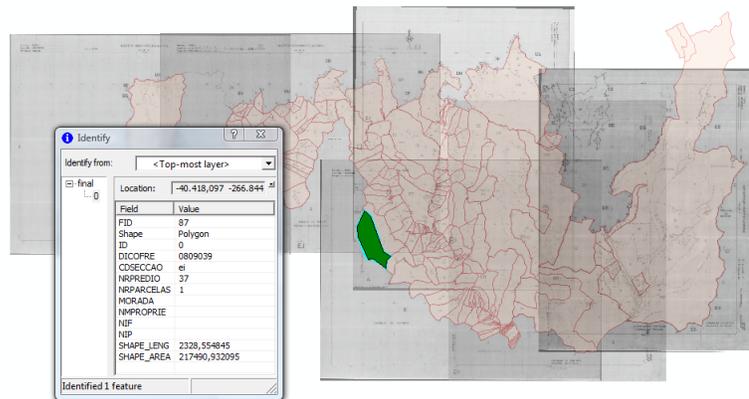


Figura 8.32: Nesta figura pode observar-se uma consulta no sistema de informação cadastral, de modo a obter os atributos de um polígono, que representa um prédio.



Figura 8.33: A sobreposição do mapa administrativo do Algarve permite identificar todas as vantagens da informatização do cadastro predial, ou seja, realizar consultas e encontrar os proprietários das propriedades de uma forma automática e centralizada.

Secção	Tempo Vect. Manual	Tempo Vect. Correção
0809039_ef	4h55min	1h30min
0809039_eg	1h20min	1h05min
0809039_eh	3h40min	1h45min
0809039_ei	2h00min	1h15min
0809039_ej	1h50min	1h03min
0809039_el	50min	55min
0809039_em	35min	25min

Tabela 8.6: Tabela que compara o tempo de execução da vectorização.

O tempo de execução através da correção da informação detectada automaticamente é consideravelmente mais rápida e permite acelerar o processo de vectorização. Na figura 8.34 pode-se observar um gráfico com os tempos comparativos de modo a compreender-se a vantagem do processo desenvolvido neste trabalho.

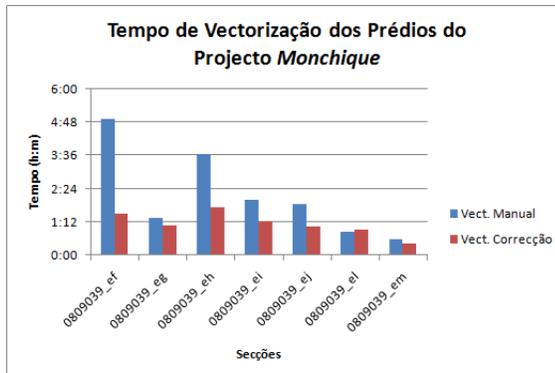


Figura 8.34: Esta imagem apresenta um gráfico de barras que mostra os tempos de vectorização dos prédios.

Capítulo 9

Conclusões Sumárias

Para o cadastro, as principais etapas do reconhecimento dos mapas cadastrais são a detecção dos círculos, o reconhecimento do número do prédio e a extração do contorno. Uma vez que, qualquer prédio tem um contorno associado, o algoritmo de extração do contorno tem como entrada a posição do círculo, para permitir uma fácil associação ao contorno.

A variabilidade das secções cadastrais ao nível do desenho é insignificante entre as regiões, uma vez que as regras e normas que orientaram a execução cadastral são idênticas. Sendo que a degradação da imagem, que está associada à idade dos mapas, é a característica principal que mais distingue os mapas cadastrais.

A aplicação desenvolvida encontra-se no estado maduro, sendo um protótipo que permite realizar trabalho técnico. Todos os problemas foram bem estudados e foram encontrados métodos simples, estáveis e robustos que resolvem a maioria dos problemas. Uma vez que os problemas são complexos e diferentes em cada mapa, a análise é realizada utilizando métodos simples. A abordagem utilizada é geral e não particular, uma vez que cada

método resolve apenas um tipo de problema. Os algoritmos estudados são suficientes para resolver todos os problemas da extracção dos mapas cadastrais portugueses.

Após a análise dos resultados, foi decidido não considerar a detecção dos arcos, sendo assim necessária a intervenção do utilizador. Esta decisão deve-se sobretudo ao excesso de sobreposição de informação dentro dos prédios que contêm os arcos, pois mesmo que existisse uma boa detecção, a informação sobreposta seria eliminada originando problemas no reconhecimento posterior.

A taxa de reconhecimento pode ser melhorada de modo a resolver problemas particulares, mas ao fazer isto, a complexidade do sistema aumenta e o melhoramento dos resultados não têm uma consequência directa. Isto acontece porque o sistema deve estar balanceado entre a complexidade e as taxas de reconhecimento. Utilizando os métodos presentes nesta tese, podemos concluir que o problema de reconhecimento está globalmente resolvido.

Uma vez que não é possível extrair a totalidade do conteúdo cadastral dos mapas, então os algoritmos devem ser implementados de modo a extrair o máximo de informação. Deste modo, os problemas necessitam de ser completamente identificados, escolhendo os algoritmos cujas características se adequem, de modo a obter a maior taxa de reconhecimento.

Extrapolando os dados da amostra para a população, i.e. multiplicando-se a média da taxa de reconhecimento de cada tipo de mapa pela correspondente quantidade percentual das secções existentes na população, é obtido o valor de 80,02% de taxa de detecção. Isto significa, que o software desenvolvido consegue extrair informação contida nas secções cadastrais de forma automática considerando a totalidade das secções do cadastro nacional.

Bibliografia

- [1] Sébastien Adam, Jean-Marc Ogier, Claude Cariou, Rémy Mullot, Jacques Labiche, and Joël Gardes. Symbol and character recognition: application to engineering drawings. *IJDAR*, 3(2):89–101, 2000.
- [2] Gady Agam, Huizhu Luo, and Its'hak Dinstein. Morphological approach for dashed lines detection. In Kasturi and Tombre [64], pages 92–105.
- [3] Andrei Alexandrescu. *Modern C++ design*. Addison Wesley Professional, 2001.
- [4] Franz L. Alt. Digital pattern recognition by moments. *J. ACM*, 9(2):240–258, 1962.
- [5] J. F. Arias, R. Kasturi, and A. Chhabra. Efficient techniques for telephone company line drawing interpretation. In *ICDAR '95: Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 2)*, page 795, Washington, DC, USA, 1995. IEEE Computer Society.
- [6] P. Arrighi and P. Soille. From scanned topographic maps to digital elevation models. In D. Jongmans, E. Pirard, and P. Trefois, editors, *Proc. of Geovision'99: International Symposium on Imaging Applications in Geology*, pages 1–4. Université de Liège, Belgium, May 1999.
- [7] Henry S. Baird. State of the art of document image degradation modeling. In *Proceedings of the 4th IAPR Workshop on Document Analysis Systems (DAS 2000)*, Rio de Janeiro, December 2000. Invited plenary talk.
- [8] D. H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.
- [9] Orit Baruch. Line thinning by line following. *Pattern Recogn. Lett.*, 8(4):271–276, 1988.
- [10] David M. Beazley. Swig: an easy to use tool for integrating scripting languages with c and c++. In *TCLTK'96: Proceedings of the 4th conference on USENIX Tcl/Tk Workshop, 1996*, pages 15–15, Berkeley, CA, USA, 1996. USENIX Association.

- [11] H. Blum. A transformation for extracting new descriptors of shape. In *Proceedings of the Symposium on Models for the Perception of Speech and Visual Form*, pages 362–380, 1967.
- [12] G. Borgerfors. Distance transforms in digital images. *Computer Vision, Graphics, and Image Processing*, 34:344–371, 1986.
- [13] Howard Butler. A guide to the python universe for esri users. In *Proceedings of the Twenty-Fourth Annual ESRI User Conference*. ESRI, 2004.
- [14] R.G. Casey and G. Naagy. Recursive segmentation and classification of composite patterns. In *ICPR82*, pages 1023–xx, 1982.
- [15] Richard G. Casey and Eric Lecolinet. A survey of methods and strategies in character segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(7):690–706, 1996.
- [16] Fu Chang, Ya-Ching Lu, and Theo Pavlidis. Feature analysis using line sweep thinning algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(2):145–158, 1999.
- [17] Y. Chang and D. Lopresti. The effects of document image defects on line drawing analysis algorithms. In *Proceedings of the IAPR Workshop on Machine Vision Applications*, pages 522–527, Kawasaki, Japan, December 1994.
- [18] J.Y. Chiang, S.C. Tue, and Y.C. Leu. A new algorithm for line image vectorization. *Pattern Recognition*, 31(10):1541–1549, October 1998.
- [19] Luigi P. Cordella and Mario Vento. Symbol recognition in documents: a collection of techniques? *IJDAR*, 3(2):73–88, 2000.
- [20] Teresa L. Cottom. Using swig to bind c++ to python. *Computing in Science and Engg.*, 5(2):88–96, c3, 2003.
- [21] T. Cover and P. Hart. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967.
- [22] Amitava Datta and Swapan K. Parui. A robust parallel thinning algorithm for binary images. *Pattern Recognition*, 27(9):1181–1192, 1994.
- [23] E. R. Davies. *Machine Vision : Theory, Algorithms, Practicalities*. Academic Press, 2nd edition edition, 1997.
- [24] P.A. Devijver and K. Kittler. *Pattern Recognition: A Statistical Approach*. Prentice Hall, 1982.
- [25] G.S. di Baja. Well-shaped, stable and reversible skeletons from the (3,4)-distance transform. *Journal of Visual Communication and Image Representation*, 5(1):107–115, March 1994.

- [26] G. Dimauro, S. Impedovo, G. Pirlo, and A. Salzo. Zoning design for handwritten numeral recognition. In *ICIAP (2)*, pages 592–599, 1997.
- [27] D. Dori, Y. Liang, J. Dowel, and I. Chai. Sparse pixel recognition of primitives in engineering drawings. *Machine Vision and Applications*, 6:69–82, 1993.
- [28] D. Dori, L. Wenyin, and K. Tombre. *Graphics Recognition - Methods and Applications*, chapter How to Win a Dashed Line Detection Contest. Springer-Verlag, 1996.
- [29] D. Dori and Liu Wenyin. Sparse pixel vectorization: An algorithm and its performance evaluation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(3):202–215, 1999.
- [30] D. Liu Wenyin Dori. Automated cad conversion with the machine drawing understandingsystem: concepts, algorithms, and performance. *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, 29:411–416, 1999.
- [31] Dov Dori. Vector-based arc segmentation in the machine drawing understanding system environment. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(11):1057–1068, 1995.
- [32] Dov Dori and Liu Wenyin. Vector-based segmentation of text connected to graphics in engineering drawings. In Petra Perner, Patrick Shen-Pei Wang, and Azriel Rosenfeld, editors, *SSPR*, volume 1121 of *Lecture Notes in Computer Science*, pages 322–331. Springer, 1996.
- [33] Ph. Dosch. *Un environnement pour la reconstruction 3D d'édifices à partir de plans d'architecte*. Thèse de doctorat, Université Henri Poincaré Nancy 1, Vandoeuvre-lès-Nancy, June 2000.
- [34] Ph. Dosch, K. Tombre, C. Ah-Soon, and G. Masini. A complete system for analysis of architectural drawings. *International Journal on Document Analysis and Recognition*, 3(2):102–116, 2000.
- [35] Michael Droettboom, Ichiro Fujinaga, Karl MacMillan, G. Sayeed Chouhury, Tim DiLauro, Mark Patton, and Teal Anderson. Using the gamera framework for the recognition of cultural heritage materials. In *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 11–17, New York, NY, USA, 2002. ACM.
- [36] Lloyd A. Fletcher and Rangachar Kasturi. A robust algorithm for text string separation from mixed text/graphics images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(6):910–918, 1988.

- [37] David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2002. FOR d 02:1 1.Ex.
- [38] Nakano Y. Fujisawa H. and Kuriro K. Segmentation methods for character recognition: from segmentation to document structure analysis. *Proc. IEEE*, 80:1079–1092, 1992.
- [39] E. Gabarra and S. Tabbone. Combining global and local threshold to binarize document images. In *Proceedings of 2nd Iberian Conference on Pattern Recognition and Image Analysis*, Estoril, Portugal, June 2005.
- [40] P. Gader, B. Forester, M. Ganzberger, A. Gillies, B. Mitchell, M. Whalen, and T. Yocum. Recognition of handwritten digits using template and model matching. *Pattern Recognition*, 24:421–431, 1991.
- [41] Basilios Gatos, Ioannis Pratikakis, and Stavros J. Perantonis. An adaptive binarization technique for low quality historical documents. In Marinai and Dengel [82], pages 102–113.
- [42] J. M. Gloger. Use of hough transform to separate merged text/graphics in forms. In *Proceedings of 11th International Conference on Pattern Recognition, Den Haag (The Netherlands)*, volume 2, 1992.
- [43] R. Gonzalez and R. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, 1992.
- [44] Chin-Chuan Han and Kuo-Chin Fan. Skeleton generation of engineering drawings via contour matching. *Pattern Recognition*, 27(2):261–275, 1994.
- [45] R. Haralick. Performance characterization in image analysis: Thinning, a case in point. In *Pattern Recognition Letters*, volume 13, pages 5–12, 1992.
- [46] X. Hilaire. *Segmentation robuste de courbes discrètes 2D et applications à la rétroconversion de documents techniques*. Thèse de doctorat, Institut National Polytechnique de Lorraine, January 2004.
- [47] Xavier Hilaire and Karl Tombre. Robust and accurate vectorization of line drawings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(6):890–904, 2006.
- [48] John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge, MA, USA, 1975.

- [49] Jared Hopkins and Tim L. Andersen. A fourier-descriptor-based character recognition engine implemented under the gamera open-source document-processing framework. In *Document Recognition and Retrieval XII, 16-20 January 2005, San Jose, California, USA, Proceedings*, pages 111–118, 2005.
- [50] Osamu Hori and David S. Doermann. Quantitative measurement of the performance of raster-to-vector conversion algorithms. In Kasturi and Tombre [64], pages 57–68.
- [51] P. V. C. Hough. Method and means for recognizing complex patterns, 1962.
- [52] Ramesh Jain, Rangachar Kasturi, and Brian G. Schunck. *Machine Vision*. McGraw Hill, 1995. JAI r2 95:1 1.Ex.
- [53] Marc Pierrot-Deseilligny Jean-Marc Viglino. A vector approach for automatic interpretation of the french cadatral map. In *ICDAR*, pages 304–308, 2003.
- [54] F. Jurie and M. Dhome. Un algorithme de template matching simple et efficace. In *Colloque Reconnaissance des Formes et Intelligence Artificielle, Angers, France*, pages 59–66, January 2002.
- [55] Toru Kaneko. Line structure extraction from line-drawing images. *Pattern Recognition*, 25(9):963–973, 1992.
- [56] T. Kanungo, R.M. Haralick, and I.T. Phillips. Global and local document degradation models. In *Proceedings of International Conference on Document Analysis and Recognition ICDAR93*, pages 730–734, Tsukuba, Japan, October 1993. IEEE Press.
- [57] T. Kanungo, R. Haralick, and D. Doff. Understanding engineering drawings: A survey. In *1st International Conference on Graphics Recognition (GREC'95), Pennsylvania State University*, pages 119–130, 1995.
- [58] Tapas Kanungo. *Document degradation models and a methodology for degradation model validation*. PhD thesis, University of Washington, Seattle, WA, USA, 1996.
- [59] Tapas Kanungo, Robert M. Haralick, Henry S. Baird, Werner Stuezle, and David Madigan. A statistical, nonparametric methodology for document degradation model validation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1209–1223, November 2000.
- [60] Tapas Kanungo and Qigong Zheng. Estimating degradation model parameters using neighborhood pattern distributions: An optimization approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(4):520–524, 2004.
- [61] R. Kasturi and J. Alemany. Information extraction from images of paper-based maps. *IEEE Transactions on Software Engineering*, 14(5):671–675, 1988.

- [62] R. Kasturi, S. T. Bow, W. El-Masri, J. Shah, J. R. Gattiker, and U. B. Mokate. A system for interpretation of line drawings. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(10):978–992, 1990.
- [63] R. Kasturi, Sing-Tze Bow, J. Gattiker, J. Shah, W. El-Masri, U. Mokate, and S. Honenahalli. A system for recognition and description of graphics. *International Conference on Pattern Recognition*, 88:255–259, 1988.
- [64] Rangachar Kasturi and Karl Tombre, editors. *Graphics Recognition, Methods and Applications, First International Workshop, University Park, PA, USA, August 10–11, 1995, Selected Papers*, volume 1072 of *Lecture Notes in Computer Science*. Springer, 1996.
- [65] J. Kilian. Simple image analysis by moments. Technical report, Freely distributable, version 0.2, 2001.
- [66] B. Kong. A benchmark: Performance evaluation of dashed line detection algorithms. In *Graphics Recognition: Methods and Applications*, pages 270–285, 1995.
- [67] V A Kovalevsky. New definition and fast recognition of digital straight segments and arcs. In *Proceedings of 10th ICPR*, pages 31–34, Los Alamitos, CA, 1990. IEEE.
- [68] Bernd Lauterbach and Walter Anheier. Segmentation of scanned maps in uniform color spaces. In *MVA*, pages 222–225, 1994.
- [69] V.F. Leavers. The dynamic generalized hough transform: Its relationship to the probabilistic hough transforms and an application to the concurrent detection of circles and ellipses. *Computer Vision, Graphics, Image Understanding*, 56(3):381–398, November 1992.
- [70] Serguei Levachkine. Raster to vector conversion of color cartographic maps. In Josep Lladós and Young-Bin Kwon, editors, *GREC*, volume 3088 of *Lecture Notes in Computer Science*, pages 50–62. Springer, 2003.
- [71] V. I. Levenshtein. Binary codes capable of correcting insertions and reversals. In *Soviet Physics Doklady*, volume 10(8), pages 707–710, 1966.
- [72] Yanhong Li, Daniel P. Lopresti, George Nagy, and Andrew Tomkins. Validation of image defect models for optical character recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(2):99–108, 1996.
- [73] S. Liang, M. Shridhar, and M. Ahmadi. Segmentation of touching characters in printed document recognition. *Pattern Recognition*, 27(6):825–840, June 1994.
- [74] X. Liu and J. R. Ramirez. Automatic extraction of hydrographic features in digital orthophoto images. *Proceedings of GIS/LIS'1997*, pages 365–373, 1997.

- [75] Jaume Vergés Llahí. *Color Constancy and Image Segmentation Techniques for Applications to Mobile Robotics*. PhD thesis, Universitat Politècnica de Catalunya, 2005.
- [76] Hervé Locteau, Sébastien Adam, Eric Trupin, and Jacques Labiche. Détection d'arcs de cercle par comparaison du tracé théorique de bresenham. *Conférence Internationale Francophone sur l'Écrit et le Document (CIFED 04)*, 2004.
- [77] Gabriele Lohman. *Computer Analysis of images and Patterns*, chapter A new method of extracting closed contours using maximal discs. *Lecture Notes in Computer Science 970*, Springer-Verlag. V. Hlavac and R. Sara, 1995.
- [78] Y. Lu. On the segmentation of touching characters. In *International Conference on Document Analysis and Recognition*, pages 440–443, 1993.
- [79] Huizhu Luo and Rangachar Kasturi. Improved directional morphological operations for separation of characters from maps/graphics. In *GREC '97: Selected Papers from the Second International Workshop on Graphics Recognition, Algorithms and Systems*, pages 35–47, London, UK, 1998. Springer-Verlag.
- [80] Karl Macmillan, Michael Droettboom, and Ichiro Fujinaga. Gamera: A structured document recognition application development environment. In *Proceedings of the 2nd Annual International Symposium on Music Information Retrieval*, pages 15–16, 2001.
- [81] Sriganesh Madhvanath and Venu Govindaraju. The role of holistic paradigms in handwritten word recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(2):149–164, 2001.
- [82] Simone Marinai and Andreas Dengel, editors. *Document Analysis Systems VI, 6th International Workshop, DAS 2004, Florence, Italy, September 8-10, 2004, Proceedings*, volume 3163 of *Lecture Notes in Computer Science*. Springer, 2004.
- [83] J. Matas, C. Galambos, and J. Kittler. Progressive probabilistic hough transform. Technical report, University of Surrey/Czech Technical University, 1998.
- [84] G. Monagan and M. Roosli. Appropriate base representation using a run graph. *Proc. ICDAR93, Tsukuba, Japan*, pages 623–626, 1993.
- [85] S. Mori, C.Y. Suen, and K. Yamamoto. Historical review of ocr research and development. *Proceedings of IEEE*, 80(7):1029–1058, July 1992.
- [86] George Nagy. Twenty years of document image analysis in pami. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(1):38–62, 2000.

- [87] C. Wayne Niblack, Phillip B. Gibbons, and David W. Capson. Generating skeletons and centerlines from the distance transform. *CVGIP: Graph. Models Image Process.*, 54(5):420–437, 1992.
- [88] Wayne Niblack. *An introduction to digital image processing*. Strandberg Publishing Company, Birkerød, Denmark, Denmark, 1985.
- [89] Instituto Geográfico Português - Centro para a Informação Cadastral. Especificações técnicas para a informatização do cadastro geométrico da propriedade rústica. *Versão 2.4*, 2005.
- [90] James R. Parker. *Algorithms for Image Processing and Computer Vision*. Wiley, 1996. PAR j 96:1 1.Ex.
- [91] T. Pavlidis. A vectorizer and feature extractor for document recognition. *Computer Vision, Graphics, and Image Processing*, 35(1):111–127, July 1986.
- [92] Maria Petrou and Panagiota Bosdogianni. *Image Processing: The Fundamentals*. Wiley, 1999. PET m2 99:1 1.Ex.
- [93] Karl Tombre Philippe Dosch, Gérald Masini. Improving arc detection in graphics recognition. In *Proceedings of the 15th International Conference on Pattern Recognition (ICPR'2000)*, 2000.
- [94] Ihsin T. Phillips and Atul K. Chhabra. Empirical performance evaluation of graphics recognition systems. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(9):849–870, 1999.
- [95] G. Priestnall, R.E. Marston, and D.G. Elliman. Arrowhead recognition during automated data capture. *Pattern Recognition Letters*, 17(3):277–286, March 1996.
- [96] Vitorino Ramos and Fernando Muge. Map segmentation by colour cube genetic k-mean clustering. In *ECDL '00: Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries*, pages 319–323, London, UK, 2000. Springer-Verlag.
- [97] Jan Rendek, Gérald Masini, Philippe Dosch, and Karl Tombre. The search for genericity in graphics recognition applications: Design issues of the qgar software system. In Marinai and Dengel [82], pages 366–377.
- [98] J. Rocha and T. Pavlidis. Character-recognition without segmentation. *Pattern Analysis and Machine Intelligence*, 17(9):903–909, September 1995.
- [99] Azriel Rosenfeld and John L. Pfaltz. Sequential operations in digital picture processing. *J. ACM*, 13(4):471–494, 1966.

- [100] P. L. Rosin and G. A. West. Segmentation of edges into lines and arcs. *Image and Vision Computing*, 7(2):109–114, 1989.
- [101] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, 2nd edition edition, 2003.
- [102] Jaakko J. Sauvola and Matti Pietikäinen. Adaptive document image binarization. *Pattern Recognition*, 33(2):225–236, 2000.
- [103] Robert Sedgewick. *Algorithms in C++: Part 1-4 Fundamentals, Data Structures, Sorting, Searching*. Addison Wesley, 3rd edition edition, 2001.
- [104] H. Shahbazkia. *Reconnaissance invariante et acquisition de connaissance: application au traitement automatique des plans de cadastre français*. PhD thesis, Université Louis Pasteur de Strasbourg, 1998.
- [105] Xuan Shi. Python for internet gis applications. *Computing in Science and Engg.*, 9(3):56–59, 2007.
- [106] Elisa H. Barney Smith and Xiaohui Qiu. Statistical image differences, degradation features, and character distance metrics. *IJDAR*, 6(3):146–153, 2003.
- [107] J. Song, M. Cai, M.R. Lyu, and S. Cai. Graphics recognition from binary images: One step or two steps. In *Proc. IAPR International Conference on Pattern Recognition (ICPR02), Quebec City, Canada, 2002*.
- [108] J. Song, M. Cai, M.R. Lyu, and S. Cai. A new approach for line recognition in large-size images using hough transform. In *Proc. IAPR International Conference on Pattern Recognition (ICPR02), Quebec City, Canada, August 11-15*, pages I:33–36, 2002.
- [109] J. Song, Z. Li, M.R. Lyu, and S. Cai. Recognition of merged characters based on forepart prediction, necessity-sufficiency matching, and character-adaptive masking. *IEEE Trans. Systems, Man and Cybernetics*, 35(1):2–11, February 2005.
- [110] J. Song, M.R. Lyu, M. Cai, and S. Cai. Graphic object recognition from binary images: a survey and an integrated paradigm. *Transactions on Systems, Man and Cybernetics, part C : Applications and Reviews (TSMCC)*, 2005.
- [111] J. Song, F. Su, J. Chen, and S. Cai. A knowledge-aided line network oriented vectorization method for engineering drawings. *Pattern Analysis and Application*, 3(2):142–152, 2000.
- [112] J. Song, F. Su, H. Li, and S. Cai. Raster to vector conversion of construction engineering drawings. *Automation in Construction*, 11(5):597–605, 2002.

- [113] Jiqiang Song, Michael R. Lyu, and Shijie Cai. Effective multiresolution arc segmentation: Algorithms and performance evaluation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(11):1491–1506, 2004.
- [114] Jiqiang Song, F. Su, C. Tai, and S. Cai. An object-oriented progressive-simplification based vectorization system for engineering drawings: Model, algorithm and performance. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24(8):1048–1060, 2002.
- [115] Jiqiang Song, F. Su, C. Tai, J. Chen, and S. Cai. Line net global vectorization: an algorithm and its performance evaluation. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'00), South Carolina, U.S.A.*, pages 383–388, 2000.
- [116] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing: Analysis and Machine Vision*. O'Reilly, 1999. SON m 99:1 1.Ex.
- [117] Salvatore Spinello and Pascal Guitton. Contour line recognition from scanned topographic maps. In *WSCG (Winter School of Computer Graphics)*, feb 2004.
- [118] R. Stefanelli and A. Rosenfeld. Some parallel thinning algorithms for digital pictures. *J. ACM*, 18(2):255–264, 1971.
- [119] Chun ta Ho and Ling-Hwei Chen. A fast ellipse/circle detector using geometric symmetry. *Pattern Recognition*, 28(1):117–124, 1995.
- [120] K. Tombre. Ten years of research in the analysis of graphics documents: Achievements and open problems. In *Proceedings of 10th Portuguese Conference on Pattern Recognition*, pages 11–17, 1998.
- [121] K. Tombre. Graphics recognition: The last ten years and the next ten years. In Wenyin Liu and Josep Lladós, editors, *Graphics Recognition—Ten Years Review and Future Perspectives*, volume 3926 of *Lecture Notes in Computer Science*, pages 422–426. Springer Verlag, 2006. Report from final panel discussion at GREC'05.
- [122] K. Tombre, C. Ah-Soon, Ph. Dosch, A. Habed, and G. Masini. Stable, robust and off-the-shelf methods for graphics recognition. In *Proceedings of the 14th International Conference on Pattern Recognition*, pages 406–408, 1998.
- [123] K. Tombre, C. Ah-Soon, et al. Stable and robust vectorization: How to make the right choices. *Proceedings of Third IAPR International Workshop on Graphics Recognition (Jaipur, India)*, 1941/2000:3–16, 1999.
- [124] K. Tombre and S. Tabbone. Vectorization in graphics recognition: To thin or not to thin. *Proceedings of 15th International Conference on Pattern Recognition, Barcelona (Spain)*, 2:91–96, 2000.

- [125] Karl Tombre. Analysis of engineering drawings: State of the art and challenges. In *GREC '97: Selected Papers from the Second International Workshop on Graphics Recognition, Algorithms and Systems*, pages 257–264, London, UK, 1998. Springer-Verlag.
- [126] Karl Tombre, Salvatore Tabbone, Loïc Pélissier, Bart Lamiroy, and Philippe Dosch. Text/graphics separation revisited. In *DAS '02: Proceedings of the 5th International Workshop on Document Analysis Systems V*, pages 200–211, London, UK, 2002. Springer-Verlag.
- [127] Feng Su Tong Lu, Chiew-Lan Tai and Shijie Cai. A new recognition model for electronic architectural drawings. *Computer Aided Design*, 37(10):1053–1069, 2005.
- [128] O. Trier, A. Jain, and T. Taxt. Feature extraction methods for character recognition - a survey, 1996.
- [129] Ø Trier, T. Taxt, and A. Jain. Recognition of digits in hydrographic maps: Binary versus topographic analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):399–404, 1997.
- [130] Øivind Due Trier, T. Taxt, and A.K. Jain. Gray scale processing of hydrographic maps. In *International Conference on Pattern Recognition*, pages 870–874, 1996.
- [131] Øivind Due Trier and Torfinn Taxt. Evaluation of binarization methods for document images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(3):312–315, 1995.
- [132] Øivind Due Trier and Torfinn Taxt. Improvement of "integrated function algorithm" for binarization of document images. *Pattern Recognition Letters*, 16(3):277–283, 1995.
- [133] M. van Ginkel, L.J. van Vliet, P.W. Verbeek, M.A. Kraaijveld, E.P. Reding, and H.J. Lammers. Robust curve detection using a radon transform in orientation space applied to fracture detection in borehole images. In *Proc. ASCI 2001, 7th Annual Conf. of the Advanced School for Computing and Imaging (Heijen, NL, May 30-June)*, ASCI, Delft, pages 84–91, 2001.
- [134] D. Vernon. *Machine Vision*. Prentice Hall, 1991.
- [135] Liu Wenyin. *Algorithms for 2D Engineering Drawings Recognition: Implementation and Evaluation*. PhD thesis, Faculty of Industrial Engineering and Management, Technion-Israel, Institute of Technology, Haifa, Israel, 1998.
- [136] Liu Wenyin and Dov Dori. A protocol for performance evaluation of line detection algorithms. *Mach. Vision Appl.*, 9(5-6):240–250, 1997.

-
- [137] C. Wiedemann, C. Heipke, H. Mayer, and S. Hinz. Automatic extraction and evaluation of road networks from moms-2p imagery. In *Automatic Extraction of GIS Objects from Digital Imagery*, 1998.
- [138] K. Wu, E. Otoo, and A. Shoshani. Optimizing connected component labeling algorithms. In *Medical Imaging 2005: Physiology, Function, and Structure from Medical Images. Edited by Amini, Amir A.; Manduca, Armando. Proceedings of the SPIE, Volume 5747, pp. 1965-1976 (2005).*, pages 1965–1976, April 2005.
- [139] Hok Sum Yam and Elisa H. Barney Smith. Estimating degradation model parameters from character images. In *ICDAR*, pages 710–714. IEEE Computer Society, 2003.
- [140] S. D. Yanowitz and A. M. Bruckstein. A new method for image segmentation. *Comput. Vision Graph. Image Process.*, 46(1):82–95, 1989.
- [141] Y.Y. Zhang and P.S.P. Wang. A maximum algorithm for thinning digital patterns. In *International Conference on Pattern Recognition*, pages 942–944, 1988.
- [142] Y. Zhu, L.D. Seneviratne, and S.W.E. Earles. New line-based thinning algorithm. *IEEE Proceedings of Vision, Image and Signal Processing*, 142(6):351–358, December 1995.

Apêndice A

Taxas de Detecção

Neste apêndice são mostrados os resultados obtidos na extracção da informação cadastral. Estes resultados são apresentados na forma de tabela, de modo a evidenciar detalhadamente as detecções de cada secção cadastral executada.

A tabela A.1 apresenta a taxa de detecção das cruces em cada secção cadastral. A coluna *NrCruzes* apresenta o número de cruces existentes na secção, a coluna *Sim* o número de cruces bem detectadas, a coluna *Não* o número de cruces não detectadas, a coluna *Mal* o número de cruces mal detectadas, a coluna *TxDetec* apresenta a taxa de detecção ($Sim/NrCruzes$) e finalmente a coluna *TxMáDetec* apresenta a taxa de más detecções (falsos alarmes) das cruces ($Mal/NrCruzes$).

Seguidamente são apresentadas todas as tabelas com os resultados das outras detecções deste sistema.

Secção	NrCruzes	Sim	Não	Mal	Tempo(s)	TxDetec.(%)	TxMáDetec.(%)
080503_t	35	35	0	0	99	100,00	0,00
081101_o	35	33	2	3	110	94,29	8,57
080301_n1	35	35	0	0	63	100,00	0,00
080504_o	35	35	0	0	80	100,00	0,00
080504_n	35	35	0	5	37	100,00	14,29
080101_f	35	30	5	8	179	85,71	22,86
080201_040	35	35	0	0	46	100,00	0,00
080301_n2	35	35	0	0	48	100,00	0,00
0809039_ef	35	33	2	0	35	94,28	0,00
0809039_eg	35	35	0	1	26	100,00	2,86
0809039_eh	35	34	1	2	20	97,14	5,71
0809039_ei	35	35	0	0	28	100,00	0,00
0809039_ej	35	33	2	0	29	94,28	0,00
0809039_el	35	32	3	0	20	91,43	0,00
0809039_em	35	34	1	1	26	97,14	2,86
0204013_p	63	62	3	2	48	98,41	3,17
0507059_d	35	30	11	4	81	85,71	11,43
0511030_au	35	32	3	2	67	91,43	5,71
0704059_a	35	32	3	0	84	91,43	0,00
0704091_c	35	23	12	0	50	65,71	0,00
0802034_024	35	33	2	0	36	94,29	0,00
0815039_q	35	18	17	1	137	51,43	2,86
1005022_i	35	29	6	17	91	82,86	48,57
1012045_v	35	17	18	2	87	48,57	5,71
1108050_p	35	31	4	4	71	88,57	11,43
1114042_c	35	27	8	9	71	77,14	25,71
1208020_s	35	23	12	4	69	65,71	11,43
1209035_o	35	30	5	0	43	85,71	0,00
1509063_h	40	37	3	2	62	92,50	5,00
Média						88,75	6,49

Tabela A.1: Esta tabela apresenta as taxas de detecção das cruzes. A taxa de detecção média é de 88.75% e a taxa de má detecções médias é de 6,49%.

Secção	NrCírculos	Sim	Não	Mal	Tempo(s)	TxDetec.(%)	TxMáDetec.(%)
080503_t	412	412	0	0	251	100,00	0,00
081101_o	37	36	0	1	249	97,30	2,70
080301_n1	57	57	0	0	242	100,00	0,00
080504_o	109	109	0	0	246	100,00	0,00
080504_n	45	44	1	0	248	97,78	0,00
080101_f	103	103	0	0	247	100,00	0,00
080201_040	91	91	0	0	251	100,00	0,00
080301_n2	40	40	0	0	241	100,00	0,00
0809039_ef	31	31	0	0	115	100,00	0,00
0809039_eg	26	26	0	0	110	100,00	0,00
0809039_eh	72	72	0	0	126	100,00	0,00
0809039_ei	66	66	0	0	128	100,00	0,00
0809039_ej	61	61	0	0	124	100,00	0,00
0809039_el	10	10	0	0	126	100,00	0,00
0809039_em	19	19	0	0	127	100,00	0,00
0204013_p	8	8	0	0	155	100,00	0,00
0507059_d	68	68	0	0	101	100,00	0,00
0511030_au	79	79	0	0	109	100,00	0,00
0704059_a	9	9	0	2	117	100,00	22,22
0704091_c	20	20	0	0	121	100,00	0,00
0802034_024	97	97	0	0	112	100,00	0,00
0815039_q	91	91	0	0	101	100,00	0,00
1005022_i	98	98	0	0	150	100,00	0,00
1012045_v	64	64	0	0	251	100,00	0,00
1108050_p	115	115	0	0	133	100,00	0,00
1114042_c	74	74	0	0	126	100,00	0,00
1208020_s	26	26	0	0	111	100,00	0,00
1209035_o	14	14	0	3	110	100,00	21,43
1509063_h	42	42	0	0	120	100,00	0,00
Média						99,83	1,60

Tabela A.2: Esta tabela apresenta a taxa de detecção dos círculos. A taxa de detecção média é de 99.83% e a taxa de má detecção média é de 1,60%.

Secção	NrArcos	Sim	Não	Mal	Tempo(s)	TxDetec.(%)	TxMáDetec.
080503_t	40	37	3	0	620	92,50	0,00
081101_o	1	1	0	26	2270	100,00	2600,00(*)
080301_n1	3	3	0	2	674	100,00	66,67
080504_o	36	31	5	3	634	86,11	8,33
080504_n	1	1	1	4	616	100,00	400,00(*)
080101_f	25	25	0	85	2109	100,00	340,00(*)
080201_040	11	9	2	3	654	81,82	27,27
080301_n2	7	7	0	0	661	100,00	0,00
0809039_ef	6	6	0	1	434	100,00	16,67
0809039_eg	0	0	0	2	306	(*)	
0809039_eh	29	13	16	1	562	44,83	3,45
0809039_ei	37	37	0	2	640	100,00	5,41
0809039_ej	2	2	0	2	328	100,00	100,00(*)
0809039_el	0	0	0	2	289	(*)	
0809039_em	0	0	0	2	340	(*)	
0204013_p	0	0	0	0	510	(*)	
0507059_d	2	1	1	4	474	50,00	200,00(*)
0511030_au	12	10	2	4	266	83,33	33,33
0704059_a	0	0	0	1	838	(*)	
0704091_c	4	4	0	0	848	100,00	0,00
0802034_024	46	31	15	5	362	67,39	10,87
0815039_q	16	13	3	1	258	81,25	6,25
1005022_i	26	24	2	2	706	92,31	7,69
1012045_v	8	8	0	0	333	100,00	0,00
1108050_p	19	14	5	0	284	73,68	0,00
1114042_c	6	5	1	0	344	83,33	0,00
1208020_s	9	3	6	0	736	33,33	0,00
1209035_o	2	1	1	1	252	50,00	50,00
1509063_h	25	12	13	0	750	48,00	0,00
Média						82,00	8,14

Tabela A.3: Esta tabela apresenta a taxa de detecção dos arcos. A taxa de detecção média é de 82.00% e a taxa de má detecção média é de 8, 14%. (*) Nos casos onde não existem arcos ou a taxa de má detecção é superior ou igual a 100% a taxa de detecção não é contabilizada, devido às razões apresentadas anteriormente.

Secção	NrTrac.	Sim	Não	Mal	Tempo(s)	TxDetec.(%)	TxMáDetec.(%)
080503_t	2465	2170	285	10	10	88,03	0,41
081101_o	1570	1475	95	0	11	93,95	0,00
080301_n1	1992	1522	332	138	7	76,41	6,93
080504_o	1765	1493	272	0	8	84,59	0,00
080504_n	2491	2160	331	0	7	86,71	0,00
080101_f	2330	2028	302	0	22	87,04	0,00
080201_040	2345	1877	468	0	6	80,04	0,00
080301_n2	997	834	163	0	4	83,65	0,00
0809039_ef	2266	1981	284	1	8	87,42	0,04
0809039_eg	1653	1452	201	0	5	87,84	0,00
0809039_eh	1366	1088	278	0	5	79,65	0,00
0809039_ei	1966	1443	523	0	7	73,40	0,00
0809039_ej	1315	1138	177	0	6	86,54	0,00
0809039_el	1249	528	712	0	5	42,27	0,00
0809039_em	548	479	69	0	4	87,41	0,00
0204013_p	1402	861	529	12	2	61,41	0,86
0507059_d	2204	1609	594	1	4	73,00	0,05
0511030_au	1531	1277	250	4	2	83,41	0,26
0704059_a	2822	1873	944	5	3	66,37	0,18
0704091_c	3063	1780	1279	4	4	58,11	0,13
0802034_024	1689	1528	161	0	6	90,47	0,00
0815039_q	1219	989	229	1	2	81,13	0,08
1005022_i	1672	1142	529	1	2	68,30	0,06
1012045_v	946	753	193	0	2	79,60	0,00
1108050_p	998	704	294	0	2	70,54	0,00
1114042_c	1989	878	1108	3	3	44,14	0,15
1208020_s	3282	2551	726	5	2	77,73	0,15
1209035_o	2643	1699	944	0	2	64,28	0,00
1509063_h	3902	1680	2222	0	5	43,05	0,00
Média						75,40	0,32

Tabela A.4: Esta tabela apresenta a taxa de detecção dos tracejados. A taxa de detecção média é de 75.40% e a taxa de má detecção média dos tracejados é de 0,32%.

Secção	NrMarcos	Sim	Não	Mal	Tempo(s)	TxDetec.(%)	TxMáDetec.(%)
080503_t	1370	1366	4	87	60	99,71	6,35
081101_o	79	79	0	583	57	100,00	737,97 (*)
080301_n1	74	72	2	461	55	97,30	622,97 (*)
080504_o	443	414	29	230	55	93,45	51,92
080504_n	137	131	6	11	50	95,62	8,03
080101_f	(*)						
080201_040	309	309	0	54	51	100,00	17,48
080301_n2	130	130	0	317	50	100,00	243,85 (*)
0809039_ef	537	536	1	92	28	99,81	17,13
0809039_eg	301	293	8	20	23	97,34	6,64
0809039_eh	673	629	44	67	25	93,46	9,96
0809039_ei	453	390	63	42	25	86,09	9,27
0809039_ej	472	453	19	20	24	95,97	4,24
0809039_el	267	267	0	86	24	100,00	32,21
0809039_em	97	97	0	9	24	100,00	9,28
0204013_p	120	120	0	35	32	100,00	29,17
0507059_d	311	302	9	0	20	97,11	0,00
0511030_au	441	398	43	0	22	90,25	0,00
0704059_a	98	59	39	44	23	60,20	44,90
0704091_c	(*)						
0802034_024	418	412	6	64	21	98,56	15,31
0815039_q	398	394	4	13	21	98,99	3,27
1005022_i	285	255	30	16	20	89,47	5,61
1012045_v	233	230	3	6	21	98,71	2,58
1108050_p	165	99	66	8	22	60,00	4,85
1114042_c	156	143	13	7	24	91,67	4,49
1208020_s	454	446	8	24	21	98,24	5,29
1209035_o	113	63	45	5	20	55,75	4,42
1509063_h	217	182	35	44	22	83,87	20,28
Média						85,57	11,58

Tabela A.5: Taxa de detecção dos marcos. A taxa de detecção média é de 85,57% e a taxa de má detecção média é de 11,57%. (*) Os valores que apresentam uma elevada taxa de más detecções ou que a secção não é aplicável nesta análise, identificam as características excepcionais dos mapas e não foram considerados nesta análise.

Secção	NrCaract.	Manual	Autom.	Mal	Tempo	TxDetec.(%)	TxTreino (%)
080503_t	5227	678	4112	437	30min	91,64	12,97
081101_o	231	126	70	35	15min	84,85	54,55
080301_n1	438	166	215	57	10min	86,99	37,90
080504_o	2004	401	1292	311	25min	84,48	20,01
080504_n	898	316	530	52	30min	94,21	35,19
080101_f	1057	350	685	22	30min	97,92	33,11
080201_040	954	341	584	29	35min	96,96	35,74
080301_n2	226	175	47	4	25min	98,23	77,43
0809039_ef	343	157	176	10	50min	97,08	45,77
0809039_eg	300	239	59	2	45min	99,33	79,67
0809039_eh	614	317	287	10	35min	98,37	51,63
0809039_ei	737	324	410	3	40min	99,59	43,96
0809039_ej	519	279	233	7	30min	98,65	53,76
0809039_el	247	101	146	0	15min	100,00	40,89
0809039_em	108	81	27	0	20min	100,00	75,00
0204013_p	100	63	37	0	18min	100,00	63,00
0507059_d	1230	260	958	12	20min	99,02	21,14
0511030_au	779	233	532	14	30min	98,20	29,91
0704059_a	307	128	137	42	40min	86,32	41,69
0704091_c	(*)						
0802034_024	898	258	633	7	45min	99,22	28,73
0815039_q	1111	270	824	17	45min	98,47	24,30
1005022_i	1746	475	1152	119	1h15min	93,18	27,21
1012045_v	593	267	321	5	35min	99,16	45,03
1108050_p	1298	431	834	33	40min	97,46	33,20
1114042_c	1006	448	416	142	1h20min	85,88	44,53
1208020_s	858	337	430	91	45min	89,39	39,28
1209035_o	439	234	182	23	35min	94,76	53,30
1509063_h	1516	549	768	199	50min	86,87	36,21
Média						91,60	40,87

Tabela A.6: Taxa de detecção dos caracteres. A taxa de detecção média dos caracteres é de 91,60% e a percentagem média de utilização de caracteres de treino é de 40,87%.

Secção	NrContornos	Sim	Não	Mal	Tempo(s)	TxDetec.(%)	TxMáDetec. (%)
080503_t	393	322	34	37	80	81,93	9,41
081101_o							
080301_n1							
080504_o	132	85	35	12	49	64,39	9,09
080504_n	30	24	3	3	32	80,00	10,00
080101_f							
080201_040	90	64	7	19	41	71,11	21,11
080301_n2	21	16	3	2	46	76,19	9,52
0809039_ef	9	6	2	1	85	66,67	11,11
0809039_eg	5	4	0	1	84	80,00	20,00
0809039_eh	91	45	38	8	93	49,45	8,79
0809039_ei	40	29	10	1	175	72,50	2,50
0809039_ej	28	19	7	2	88	67,86	7,14
0809039_el	2	1	0	1	50	50,00	50,00
0809039_em	6	3	3	1	62	50,00	16,67
0204013_p							
0507059_d	50	38	11	1	70	76,00	2,00
0511030_au	48	36	12	0	99	75,00	0,00
0704059_a							
0704091_c							
0802034_024	54	40	11	3	116	74,07	5,56
0815039_q	79	63	14	2	62	79,75	2,53
1005022_i	73	45	25	4	74	61,64	5,48
1012045_v	53	44	8	1	44	83,02	1,89
1108050_p	95	86	9	0	44	90,53	0,00
1114042_c	62	27	33	2	118	43,55	3,23
1208020_s							
1209035_o							
1509063_h							
Média						69,68	9,80

Tabela A.7: Esta tabela apresenta o resumo da taxa de detecção do contorno dos prédios, nas secções onde o método é aplicável.

Secção	Cruzes	Círculos	Arcos	Tracej.	Marcos	Caract.	Contornos
080503_t	100,00	100,00	92,50	88,03	99,71	91,64	81,93
081101_o	94,29	97,30	100,00	93,95	100,00	84,85	
080301_n1	100,00	100,00	100,00	76,41	97,30	86,99	
080504_o	100,00	100,00	86,11	84,59	93,45	84,48	64,39
080504_n	100,00	97,78	100,00	86,71	95,62	94,21	80,00
080101_f	85,71	100,00	100,00	87,04	-	97,92	
080201_040	100,00	100,00	81,82	80,04	100,00	96,96	71,11
080301_n2	100,00	100,00	100,00	83,65	100,00	98,23	76,19
0809039_ef	94,29	100,00	100,00	87,42	99,81	97,08	66,67
0809039_eg	100,00	100,00		87,84	97,34	99,33	80,00
0809039_eh	97,14	100,00	44,83	79,65	93,46	98,37	49,45
0809039_ei	100,00	100,00	100,00	73,40	86,09	99,59	72,50
0809039_ej	94,29	100,00	100,00	86,54	95,97	98,65	67,86
0809039_el	91,43	100,00	(*)	42,27	100,00	100,00	50,00
0809039_em	97,14	100,00	(*)	87,41	100,00	100,00	50,00
0204013_p	98,41	100,00	(*)	61,41	100,00	100,00	
0507059_d	85,71	100,00	50,00	73,00	97,11	99,02	76,00
0511030_au	91,43	100,00	83,33	83,41	90,25	98,20	75,00
0704059_a	91,43	100,00	(*)	66,37	60,20	86,32	
0704091_c	65,71	100,00	100,00	58,11	-	-	
0802034_024	94,29	100,00	67,39	90,47	98,56	99,22	74,07
0815039_q	51,43	100,00	81,25	81,13	98,99	98,47	79,75
1005022_i	82,86	100,00	92,31	68,30	89,47	93,18	61,64
1012045_v	48,57	100,00	100,00	79,60	98,71	99,16	83,02
1108050_p	88,57	100,00	73,68	70,54	60,00	97,46	90,53
1114042_c	77,14	100,00	83,33	44,14	91,67	85,88	43,55
1208020_s	65,71	100,00	33,33	77,73	98,24	89,39	
1209035_o	85,71	100,00	50,00	64,28	55,75	94,76	
1509063_h	92,50	100,00	48,00	43,05	83,87	86,87	
Média	88,75	99,83	82,00	75,40	91,91	94,87	69,68

Tabela A.8: Tabela que apresenta o resumo das taxas de reconhecimento de cada módulo individualmente.