



**Universidade do Minho**  
Escola de Engenharia

Gonalo Braz Afonso

**OCR para documentos estruturados antigos**  
**Old structured documents OCR**





**Universidade do Minho**  
Escola de Engenharia

Gonçalo Braz Afonso

**OCR para documentos estruturados antigos**  
**Old structured documents OCR**

Dissertação de Mestrado  
Mestrado em Engenharia Informática

Trabalho efetuado sob a orientação de  
**José João Antunes Guimarães Dias Almeida**

# Direitos de Autor e Condições de Utilização do Trabalho por Terceiros

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

## Licença concedida aos utilizadores deste trabalho:

*[Caso o autor pretenda usar uma das licenças Creative Commons, deve escolher e deixar apenas um dos seguintes ícones e respetivo lettering e URL, eliminando o texto em itálico que se lhe segue. Contudo, é possível optar por outro tipo de licença, devendo, nesse caso, ser incluída a informação necessária adaptando devidamente esta minuta]*



**CC BY**

<https://creativecommons.org/licenses/by/4.0/> *[Esta licença permite que outros distribuam, remixem, adaptem e criem a partir do seu trabalho, mesmo para fins comerciais, desde que lhe atribuam o devido crédito pela criação original. É a licença mais flexível de todas as licenças disponíveis. É recomendada para maximizar a disseminação e uso dos materiais licenciados.]*



## CC BY-SA

<https://creativecommons.org/licenses/by-sa/4.0/> [Esta licença permite que outros remisturem, adaptem e criem a partir do seu trabalho, mesmo para fins comerciais, desde que lhe atribuam o devido crédito e que licenciem as novas criações ao abrigo de termos idênticos. Esta licença costuma ser comparada com as licenças de software livre e de código aberto «copyleft». Todos os trabalhos novos baseados no seu terão a mesma licença, portanto quaisquer trabalhos derivados também permitirão o uso comercial. Esta é a licença usada pela Wikipédia e é recomendada para materiais que seriam beneficiados com a incorporação de conteúdos da Wikipédia e de outros projetos com licenciamento semelhante.]



## CC BY-ND

<https://creativecommons.org/licenses/by-nd/4.0/> [Esta licença permite que outras pessoas usem o seu trabalho para qualquer fim, incluindo para fins comerciais. Contudo, o trabalho, na forma adaptada, não poderá ser partilhado com outras pessoas e têm que lhe ser atribuídos os devidos créditos.]



## CC BY-NC

<https://creativecommons.org/licenses/by-nc/4.0/> [Esta licença permite que outros remisturem, adaptem e criem a partir do seu trabalho para fins não comerciais, e embora os novos trabalhos tenham de lhe atribuir o devido crédito e não possam ser usados para fins comerciais, eles não têm de licenciar esses trabalhos derivados ao abrigo dos mesmos termos.]



## CC BY-NC-SA

<https://creativecommons.org/licenses/by-nc-sa/4.0/> [Esta licença permite que outros remisturem, adaptem e criem a partir do seu trabalho para fins não comerciais, desde que lhe atribuam a si o devido crédito e que licenciem as novas criações ao abrigo de termos idênticos.]



## CC BY-NC-ND

<https://creativecommons.org/licenses/by-nc-nd/4.0/> [Esta é a mais restritiva das nossas seis licenças principais, só permitindo que outros façam download dos seus trabalhos e os comparti-

*lhem desde que lhe sejam atribuídos a si os devidos créditos, mas sem que possam alterá-los de nenhuma forma ou utilizá-los para fins comerciais.]*

## **Agradecimentos**

Escreva aqui os seus agradecimentos. Não se esqueça de mencionar, caso seja esse o caso, os projetos e bolsas dos quais se beneficiou enquanto fazia a sua investigação. Pergunte ao seu orientador sobre o formato específico a ser usado. (As agências de financiamento são bastante rigorosas quanto a isso.)

# **Declaração de Integridade**

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

Universidade do Minho, Braga, janeiro 2024

Gonçalo Braz Afonso



# Resumo

A digitalização de documentos permitiu uma nova forma de salvaguardar informação para a prosperidade, evitando a sua perda para o deterioramento físico destes. De forma a posteriormente transcrever estes documentos para permitir uma consulta, procura, processamento e manipulação mais simples destes o uso de software de **OCR** é essencial. Esta tecnologia é no entanto dependente, em diferentes níveis, das características do seu alvo, nomeadamente: qualidade da imagem, complexidade da estrutura do documento, linguagem do texto. Documentos mais antigos, em especial jornais por apresentarem estruturas mais complexas, apresentam por este motivo resultados que diferem bastante do seu conteúdo original; tanto a nível do texto reconhecido, como da sua organização para os diferentes outputs disponíveis (ex.: txt simples).

A tarefa de extrair informação destes documentos, como por exemplo o isolamento e extração de artigos, torna-se numa tarefa complexa e propícia a erros. Este trabalho pretende então a criação de uma ferramenta ou um conjunto de ferramentas que permitam auxiliar o processo de extração de conteúdo de documentos, primeiramente mas não exclusivamente, mais antigos e de estruturados, com especial foco em jornais. A pipeline do projeto pretende então ser capaz de detetar e lidar com os diferentes pontos de risco nestes documentos: qualidade da imagem, erros nos resultados de **OCR**, segmentação e organização do documento, criação do output organizado.

Diferentes alternativas para **OCR** assim como métodos de tratamento destes problemas serão estudados, comparados e implementados de forma a encontrar a melhor solução para a resolução deste problema. O produto final implementado será composto por uma ferramenta utilizável num formato de um **GUI** simples ou comando de consola.

Para documentos antigos a linguagem, como mencionado, pode afetar os resultados de **OCR**. Deste modo, como objetivo secundário, propõe-se a criação de uma ferramenta que facilite a criação de um dicionário para diferentes iterações de uma linguagem para este ser posteriormente fornecido ao motor **OCR**.

**Palavras-chave** OCR, Digitalização, Documentos estruturados, Documentos antigos, Segmentação de documentos, Tratamento de imagem, Modernização de texto

# Abstract

Write abstract here (en)

**Keywords** OCR, Digitalization, Structured documents, Old documents, Document segmentation, Image treatment, Text modernization

# Conteúdo

<b>I</b>	<b>Material Introdutório</b>	<b>1</b>
<b>1</b>	<b>Introdução</b>	<b>2</b>
1.1	Enquadramento e motivação . . . . .	2
1.2	Objetivos . . . . .	3
1.3	Estrutura da dissertação . . . . .	4
<b>2</b>	<b>Estado da arte</b>	<b>6</b>
2.1	Reconhecimento ótico de caracteres . . . . .	6
2.1.1	Introdução . . . . .	6
2.1.2	Breve história e evolução . . . . .	6
2.1.3	Processo OCR . . . . .	7
2.1.4	Desafios . . . . .	9
2.1.5	Tecnologia . . . . .	10
2.2	Trabalho relacionado . . . . .	11
2.2.1	Pré Processamento para OCR . . . . .	11
2.2.2	Pós Processamento para OCR . . . . .	14
2.2.3	Identificação de imagens . . . . .	14
2.2.4	Segmentação de documentos . . . . .	14
2.2.5	Ordem de leitura . . . . .	14
<b>3</b>	<b>O problema e os seus desafios</b>	<b>15</b>
3.1	Imagens . . . . .	15

<b>II</b>	<b>Core da Dissertação</b>	<b>16</b>
<b>4</b>	<b>Contribuição</b>	<b>17</b>
4.1	Introdução . . . . .	17
4.2	Sumário . . . . .	17
<b>5</b>	<b>Aplicações</b>	<b>18</b>
5.1	Introdução . . . . .	18
5.2	Sumário . . . . .	18
<b>6</b>	<b>Conclusões e trabalho futuro</b>	<b>19</b>
6.1	Conclusões . . . . .	19
6.2	Perspetiva de trabalho futuro . . . . .	19
<b>7</b>	<b>Planeamento</b>	<b>20</b>
7.1	Atividades . . . . .	20
<b>III</b>	<b>Apêndices</b>	<b>23</b>
<b>A</b>	<b>Trabalho de apoio</b>	<b>24</b>
<b>B</b>	<b>Detalhes dos resultados</b>	<b>25</b>
<b>C</b>	<b>Listings</b>	<b>26</b>
<b>D</b>	<b>Ferramentas</b>	<b>27</b>

## Lista de Figuras

1	Legenda . . . . .	15
---	-------------------	----

## Lista de Tabelas

1	Plano de atividades. . . . .	20
---	------------------------------	----

# Acrónimos

**CNN** Convolutional Neural Network.

**dpi** dots per inch.

**EA** Estado da Arte.

**GAN** Generative Adversarial Network.

**GUI** graphic user interface.

**ML** Machine Learning.

**OCR** reconhecimento óptico de caracteres.

**RPD** Relatório de Pré-Dissertação.



# Glossário

**codec** Algoritmo de compressão com o propósito de diminuir a dimensão de uma entidade..

**ground truth** Em OCR, este é o nome dado ao conteúdo do documento original. Usualmente apenas o conteúdo textual é considerado..

# **Parte I**

## **Material Introdutório**

# Capítulo 1

## Introdução

### 1.1 Enquadramento e motivação

A digitalização tem um papel fundamental na conservação, disponibilização e proliferação de documentos físicos, não só contemporâneas, como de eras anteriores à revolução da informação. Esta tecnologia, acoplada a ferramentas de **OCR**, veio trazer uma facilidade de navegação, consulta e manipulação destes documentos que anteriormente não era possível.

A eficácia de **OCR** é no entanto dependente de vários fatores nas imagens ou ficheiros alvo: a qualidade das imagens, como a resolução, estado do documento, coloração, qualidade/tipo de escrita; a estrutura dos documentos - quanto mais complexo, mais difícil é obter a informação de forma automática mantendo a congruência original; linguagem do texto, sendo que por vezes diferentes tecnologias, como por exemplo **Tesseract**, procuram verificar a sua confiança na deteção com o vocabulário conhecido, o qual pode no entanto não coincidir com a época de produção do documento; entre outras.

Estas dependências são especialmente notórias quando se envolvem documentos mais antigos, os quais podem, além de apresentar envelhecimento causado pelo tempo e danos pelas condições de armazenamento, devido às limitações tecnológicas assim como por vezes à falta de convenções de formatação dos documentos, não dispor de uma consistência no formato e texto (estrutura, alinhamento, dimensões dos caracteres, fonte de texto consistente, etc.) usual nos documentos atuais. Estes fatores, resultam então num reconhecimento de texto não tão satisfatórios como se esperaria.

Estes documentos antigos são mais comumente, mas não exclusivamente, reconhecidos como anteriores à era da digitalização, sendo que o foco de trabalho será maioritariamente dirigido a documentos desta época, como jornais, revistas e outros, do século passado ou anteriores.

Em especial documentos com estruturas complexas, como é o caso de jornais, onde é possível a segmentação em diferentes partes com conteúdo e propósito distinto e ao mesmo tempo uma ordem de leitura complexa i.e., não segue apenas regras simples de posição do conteúdo (texto da esquerda antes

do texto da direita e cima antes de baixo) mas que exige também noção das características e relação do conteúdo.

Mesmo para ficheiros do tipo **hOCR** ou **PDF**, que já passaram por um processo de reconhecimento de texto, a complexidade da estrutura dos documentos originais ou problemas nos elementos que contém o texto (como por exemplo elementos sobrepostos ou que se intersejam) dificultam a extração e interpretação do seu conteúdo, podendo ser facilmente perdida a lógica original.

Por estas razões, seria útil uma ferramenta que permita uma deteção e tratamento destes documentos de forma automática e de uso simples, permitindo um certo nível de configuração para adaptação entre tipos de documentos com características bem definidas e distintas.

O presente documento pretende então servir como um estudo dos desafios apresentados por estes tipos de documentos perante **OCR**, assim como a procura de soluções para a melhoria dos resultados na deteção e extração de texto e assim criar uma ferramenta que torne o processo de extração de informação destes tipos de documentos mais simples e fiável.

Como trabalho complementar, é proposta a implementação de um método de modernização do conteúdo extraído, envolvendo a criação de uma ferramenta capaz de criar dicionários entre diferentes iterações de uma mesma linguagem.

## 1.2 Objetivos

O principal objetivo deste trabalho é a realização de um estudo sobre os problemas apresentados à extração de conteúdo de documentos de estrutura complexa - mantendo a sua lógica original -, assim como a implementação de uma solução para resolver ou mitigar estes desafios, aumentando a confiança na informação extraída. Em termos dos casos alvo do trabalho, será prioridade o estudo de jornais com texto máquina. Jornais por serem um particular tipo que apresenta mais dificuldades e se encontra em maior procura de soluções; e texto máquina por ser mais comum para este tipo de documento. Esta segunda restrição é menos relevante pois não é uma dificuldade do trabalho e pode ser resolvida perante a escolha da tecnologia a usar.

Especificando, os objetivos são:

- Estudo sobre os diferentes softwares de **OCR** disponíveis e as diferenças entre estes.
- Estudo as dificuldades que documentos podem apresentar no processo de reconhecimento de texto.

- Estudar o trabalho desenvolvido sobre a área de tratamento de imagem, identificação de tipo de documento, segmentação de documentos, algoritmos de cálculo da ordem de leitura, melhoramento de resultados de OCR e métricas de validação de resultado OCR.
- Estudar trabalhos com âmbito similar ou relacionado ao presente.
- Implementação de um conjunto de ferramentas dirigidas à solução dos problemas propostos.
- Implementação de uma ferramenta em formato **GUI** e comando de consola que aplique uma pipeline cujo input seria um ficheiro - imagem, pdf, hOCR -, identifique e trate de problemas deste se necessário para melhorar os resultados de OCR, e por fim devolva um output que mantenha a lógica e conteúdo do documento original.
- Secundário : ferramenta para criação de dicionário de linguagem para modernização de documentos. Ferramenta tem como input duas versões de um documento na mesma linguagem mas iterações diferentes e dá como output um dicionário entre as versões.

Estudo sobre alinhamento de documentos.

## 1.3 Estrutura da dissertação

Esta dissertação segue a seguinte estrutura:

- Capítulo 1: Breve contextualização sobre o tema proposto, as dificuldade impostas por documentos estruturados e com digitalizações ou condições físicas degradadas nos resultados **OCR** e a utilidade de uma ferramenta para o tratamento destas. Além disso foram listados os objetivos do trabalho.
- Capítulo 2: Estudo sobre o estado da arte nos tópicos relacionados ao tema da dissertação, as suas dificuldades e soluções destas; estudo de trabalho anteriormente realizado com âmbito similar ao atual.
- Capítulo 3: Listagem dos diferentes problemas que a solução irá abranger e os desafios que estes apresentam.
- Capítulo 4: Descrição da solução implementada, a sua arquitetura, componentes e características.
- Capítulo 5: Apresentação e estudo dos resultados da solução implementada.

- Capítulo 6: Reflexão sobre o trabalho realizado, os resultados e a experiência obtida, assim como uma breve exploração de caminhos para trabalho futuro do projeto.
- Capítulo 7: No último capítulo é explicado o plano de desenvolvimento da dissertação.

## Capítulo 2

### Estado da arte

Neste capítulo, será feita uma exposição do estado da arte das tecnologias relacionadas com o tema ou relevantes para o projeto, assim como trabalhos relacionados, quer no mesmo tema ou envolvente - algoritmos relevantes para o desenvolvimento -, procurando plantar uma base para o trabalho realizado e futuro, entendendo o que já foi explorado e o que está para vir em alguns casos.

#### 2.1 Reconhecimento ótico de caracteres

##### 2.1.1 Introdução

O reconhecimento ótico de caracteres é a tecnologia base do projeto proposto, estando presente em qualquer instância ou caso de estudo que será explorado, inclusive no caso em que sejam utilizados ficheiros do género **hOCR**, onde não será no software implementado, aplicado o processo de **OCR**, tal deve-se ao facto de estes serem um resultado de **OCR**.

Na sua essência e como o nome indica, software de reconhecimento ótico de caracteres permitem a deteção e transcrição de texto a partir de imagens, de forma automática e autónoma. Utilizando esta habilidade, abriu-se a possibilidade de tornar os documentos digitalizados ao longo do tempo numa fonte mais útil de informação: navegada, consultada e editada mais facilmente. Isto pois, embora a digitalização de documentos seja um grande passo para a sua preservação, a sua consulta posterior requer a adição de dados adicionais, como meta-dados, para permitir a sua indexação.

##### 2.1.2 Breve história e evolução

[Srihari et al. \[2003\]](#) e [Berchmans and Kumar \[2014\]](#) apresentam a história do reconhecimento ótico de caracteres desde a conceção do seu ideal no século XIX como uma tecnologia para auxílio de pessoas com impedimentos na leitura, até aos pontos alcançados na última década onde até escrita humana se

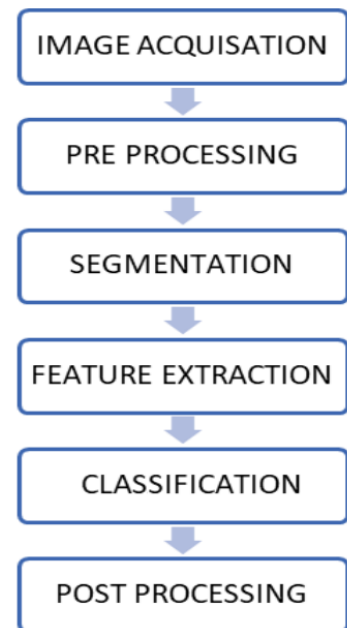
tornou num desafio, até certo ponto, conquistável. As primeiras instâncias de reconhecimento óptico realizado por máquinas deu-se no final do séc. XIX, mais especificamente em 1870 por Charles R. Carey com a criação de um scanner de retina, mas é necessário esperar até meio do século seguinte e a consequente evolução que decorreu nesta área, para a subárea de reconhecimento de caracteres começar a ver a sua comercialização com a invenção de David Shepard: GISMO. A génese desta tecnologia começou num formato bastante limitado, sendo capaz apenas de reconhecer um conjunto muito limitado de caracteres de uma fonte específica a um ritmo de 1 carácter por minuto, isto em condições de input bem controladas (papel sem ruído, apenas com o texto a ser reconhecido). Esta considerada por [Berchmans and Kumar \[2014\]](#) como a primeira geração de **OCR**. A segunda geração começa a dar os primeiros passos no processamento de escrita humana, como é exemplo o *IBM 1287* na década de 60. A terceira geração, nas décadas de 70 e 80, introduziu um maior foco no processamento da escrita humana e na capacidade de lidar com problemas na imagem original. A quarta geração tornou-se capaz de tratar documentos complexos com misturas entre texto e imagens, assim como qualidades de inputs menos favoráveis, documentos com cor e mais precisão com texto manuscrito. Atualmente com a evolução das técnicas de pré processamento, assim como os algoritmos de reconhecimento e a ascensão da inteligência artificial [[Mittal and Garg, 2020](#)], a precisão e flexibilidade dos softwares de **OCR** são capazes de, até em imagens de paisagens, segmentar e reconhecer texto localmente de forma automática e com pouco pré processamento. Além disso, embora tenha sido o foco anteriormente em software **OCR** pago e dedicado a um tipo específico de documentos, a implementação de softwares mais geral e de uso aberto tem-se tornado mais vulgar. Em algumas instâncias complexas - documento complexo e linguagem com caracteres fora do latim -, já existe tecnologia capaz de obter taxas de acerto acima dos 95% mesmo para texto escrito à mão [[Mittal and Garg, 2020](#)].

### **2.1.3 Processo OCR**

Um software **OCR** pode ter reconhecimento online ou offline [[Srihari et al., 2003](#)][[Berchmans and Kumar, 2014](#)]. O primeiro é reconhecimento em tempo real, em que usualmente o input é feito num dispositivo como um tablet digitalizador, no formato de um conjunto de coordenadas, podendo portanto ser mais preciso a custo de menor flexibilidade na entrada. O mais comum, método offline, recebe como um input por norma uma imagem com o documento finalizado. O bitmap desta imagem será utilizado o alvo do reconhecimento de caracteres. Utilizando este último método um tipo de entrada menos controlado, exige uma fase pré processamento mais minuciosa do que o reconhecimento online. Focando no reconhecimento offline, este pode ser geralmente dividido em 6 partes:



- **Aquisição de input** : imagem a ser reconhecida, incluindo algoritmos de compressão do próprio formato guardado.
- **Pré processamento** : técnicas de manipulação do input para melhorar resultado de **OCR**
- **Segmentação** : segmentação do input, a vários níveis, de modo a isolar o melhor possível os conteúdos relevantes, i.e. o texto.
- **Extração de características** : processo de reconhecimento de características dos caracteres isolados.
- **Classificação** : utilizando as características calculadas é feita a decisão sobre a sua identidade.
- **Pós processamento** : técnicas para melhoria do resultado, como por exemplo a correção de erros ortográficos. Por vezes pode alterar o documento original se este contiver erros deste tipo.



O **Pré Processamento** é um passo essencial para o aumento do acerto do reconhecimento de texto, sendo que ele pretende remover imperfeições do input como: baixo contraste das linhas, texto mal delimitado, ruído de imagem, orientação do documento ou do texto (principalmente manuscrito). Em alguns casos mais complexos, com ajuda de inteligência artificial, também é possível a reposição de partes parciais de uma imagem que foram perdidas, ou remoção de elementos como *watermarks*.

A **Segmentação** é usada para isolar o conteúdo útil do resto da imagem podendo envolver vários passos como: segmentação da página para separar texto do resto do conteúdo; segmentação de caracteres, com o intuito de os separar em caracteres individuais, algo que é especialmente difícil com escrita à mão devido à tendência em criar ligações entre caracteres ou mesmo de os unir; tratamento e normalização dos caracteres isolados - normalização do tamanho, filtração morfológica.

A **Extração de Características** (Feature Extraction) trata-se do processo de deteção e cálculo das características dos caracteres, para a criação do classificador (dependendo da arquitetura) e anotação do que distingue o carácter alvo. Este processo é possivelmente o mais aberto para variações e que, juntamente com o classificador, mais influencia o resultado. Diferentes técnicas de extração de características e **Classificação** são utilizadas e foram estudadas durante as últimas décadas: desde *template matching*

[Srihari et al., 2003] onde são usados algoritmos para cálculo de similaridade entre um template e o alvo, a segmentação de características como presença de loops ou traços verticais maiores [Srihari et al., 2003], ou distribuições de pixels [Mittal and Garg, 2020]. Para texto humano, este processo torna-se ainda mais complexo devido à necessidade de lidar com múltiplos caracteres invés de singulares. A classificação passava por um processo de comparação do valor das características calculado com diferentes templates, porém mais recentemente, o uso de estratégias no ramo de machine learning são mais comuns: redes neurais, support vector machines e k- nearest neighbor; são alguns dos modelos mais utilizados [Mittal and Garg, 2020] Berchmans and Kumar [2014]. Por vezes, o classificador utiliza conhecimento do léxico de uma linguagem para ajudar na sua classificação, sendo que documentos com linguagem desatualizada poderão sofrer nesse caso.

O **Pós Processamento** é responsável pelo tratamento do output, responsável por mitigar ou corrigir alguns erros do reconhecimento, desde correções ortográficas a posicionamento na página [Mittal and Garg, 2020].

Este trabalho irá ter como foco principal as secções de pré e pós processamento na procura de aumentar a eficácia do reconhecimento.

## 2.1.4 Desafios

Com a evolução da tecnologia, os problemas foram mudando de foco, tendo passado por um longo período em que a maior prioridade era a capacidade de reconhecimento de caracteres para além de um escopo limitado, tanto em termos de identidade como estilo, para a capacidade de tratar a imagem de forma a que o reconhecimento tenha uma maior taxa de acerto [Bieniecki et al., 2007]. Alguns dos maiores desafios atualmente para **OCR** são:

- documento original : danos no objeto; texto ilegível ou com um tipo de letra muito complexo; linguagem desatualizada; estrutura complexa; inclinação do texto; distorções da página.
- imagem : má iluminação; múltiplas páginas com orientação diferentes orientações; baixa resolução; pouco contraste; ruído.
- classificador ou extrator de features não adequado para uma dada linguagem.
- resultado : validação quando não se tem a **ground truth** disponível
-

Dentro destes, o processamento de estruturas complexas será o foco principal e o esperado maior contributo deste trabalho.

### 2.1.5 Tecnologia

Presentemente, com a proliferação permitida pela internet e a globalização, a disponibilização de ferramentas de **OCR**, anteriormente primariamente privilégio de instituições ou empresas, como bancos [Srihari et al., 2003], tornou-se trivial, acessível através de itens do dia a dia como um computador ou telemóvel de forma gratuita, ex.: Google Lens.

Alguns destes softwares que serão utilizados neste trabalho são:

- **Tesseract**
- **Keras-OCR**
- **PaddleOCR**

Os resultados deste tipo software podem ser genericamente descritos como uma lista de caixas, delimitadoras de texto, o seu conteúdo, i.e. o texto nela contido e, por norma, o nível de confiança no reconhecimento nesse texto. No caso do **Tesseract** [tes], dentro das várias formas que os resultados podem ser apresentados, a lista de caixas pode ser interpretada como uma árvore de blocos, em que cada nível corresponde a um tipo de estrutura no documento: página → bloco → parágrafo → linha → texto. Usando **PaddleOCR** [pad], os resultados são mais simples, divididos apenas pelas linhas de texto detetado. Já o **Keras-OCR** [ker] lista um conjunto de caixas em que cada contém uma palavra reconhecida. Uma outra característica que o **Tesseract** tem é a capacidade de reconhecer, com nível de acerto variável, outros elementos relevantes de um documento, como imagens ou delimitadores. Isto pode, por outro lado, causar erros na interpretação dos resultados por sobreposição ou multiplicação da quantidade de caixas.

O output deste tipo de software pode ainda ser processado para tomar diversas formas: formatos que apenas têm o conteúdo como texto simples ou markdown; formatos que mantêm informação sobre os blocos detetados, como hOCR.

A validação do output é na maioria dos casos medida a partir da comparação com a **ground truth**, o que limita a capacidade de testar e treinar (no caso de **ML** supervisionado) modelos visto que os datasets tem de ser criados de forma minuciosa e consumidora de tempo.

Além dos softwares de reconhecimento, é preciso ter atenção ao tipo dos ficheiros de entrada. Estes são usualmente imagens e, dependendo do tipo de **codec** destes, os algoritmos de compressão aplica-

dos poderão diminuir a qualidade de imagem, no caso de formatos *lossy* como **JPEG**, potencialmente diminuindo o acerto do reconhecimento do texto. [Darwiche et al. \[2015\]](#), no seu estudo demonstra que mesmo entre diferentes tipos de *lossy* **codec** o seu impacto pode variar relevantemente nos resultados de **OCR**, sendo que o formato **JPEG**, um dos mais populares, resultou nas menores taxas de sucesso.

## 2.2 Trabalho relacionado

### 2.2.1 Pré Processamento para OCR

Como foi listado na secção 2.1.4, existe uma diversa quantidade de defeitos que os documentos originais e as imagens digitalizadas destes podem ter e cuja presença pode afetar negativamente os resultados de software de **OCR**. O pré processamento pode ser considerado como uma fase de tratamento de imagem para remover estes problemas que deterioram o reconhecimento de texto. De forma a entender os diferentes métodos utilizados e a sua evolução para o presente, foram seleccionados os estudos: [\[Bi-eniecki et al., 2007\]](#),[\[Likforman-Sulem et al., 2009\]](#),[\[Souibgui and Kessentini, 2022\]](#),[\[Lat and Jawahar, 2018\]](#),[\[Dey et al., 2022\]](#),[\[Wei et al., 2018\]](#),[\[Bui et al., 2017\]](#). Entre o grande leque de diferentes algoritmos e tratamentos que podem ser aplicados nas imagens, em geral, estes podem ser segmentados nos mais comuns [\[Dey et al., 2022\]](#):

- **binarização/thresholding da imagem** : processo de normalização dos pixels para um de dois valores, mediante um determinado limiar
- **remoção de ruído** : algoritmos para retirar degradações ou sujidades da imagem através de processos como, por exemplo, suavizando a imagem calculando para cada pixel o valor médio da sua vizinhança
- **correções de texto** : alguns casos deste são texto que apresenta um ângulo de rotação, texto com inclinação, distorções locais no texto, *watermarks*
- **super-resolução** : aumentar a resolução da imagem
- **foco da imagem** : acentuação das arestas, diminuir desfocagem
- **transformações morfológicas** : operações sobre a imagem de modo a provocar maior contraste do conteúdo, ou permitir melhor distinção das características, ex.: dilatação do texto para tornar mais fácil a distinção entre regiões com e sem texto.

Para **binarização**, o objetivo principal é distinguir o texto do resto da imagem, daí a alocação dos pixels para 1 de dois valores. Este processo é distinguido principalmente entre o uso de *thresholding* global ou local (ou adaptativo), sendo que o global implica um cálculo das características estatísticas locais dentro da imagem, e é mais adequado para o tratamento de imagens com cor, ou com variações de intensidade dispersas pela imagem [Dey et al., 2022]. Alguns dos algoritmos mais comuns são comparados por Souibgui and Kessentini [2022], onde é evidente a dependência destes nas condições do documento original e da imagem. Um exemplo apresentado é numa imagem com uma mancha escura (com o texto ainda distinguível), o algoritmo de Otsu conseguiu gerar uma imagem com pouco ruído e bom contraste, mas a zona da mancha ficou completamente preta, comparado ao algoritmo de Niblack que, embora com mais ruído, recuperou algum texto dentro da mancha.

A **remoção de ruído** é possivelmente a área com mais alternativas possíveis e das que mais afeta o resultado do reconhecimento por, a nível de pixels, o ruído interferir com a composição dos caracteres ou criar acumulações de informação extra que serão mal identificadas pelos softwares, como notado por Bieniecki et al. [2007]. O ruído nas imagens pode ser de vários o que dificulta a forma de o detetar e tratar. Alguns dos outros tratamentos, como a binarização, transformações morfológicas e alguns tipos de super-resolução, também tratam desta questão mesmo não sendo o seu foco principal. Da mesma forma, alguns dos filtros utilizados podem ter outros resultados como o aumento do contraste ou eliminação de distorções. Alguns dos tratamentos mais comuns do ruído são [Dey et al., 2022][Bui et al., 2017][Bieniecki et al., 2007]:

- filtro Gaussiano
- médias não locais
- suavização com filtro de mínimos locais
- suavização com filtro Wiener

Vários destes métodos resultam tanto na acentuação das arestas do texto, como na remoção de lixo ou ruído à sua volta, recuperando o **foco da imagem**.

A **correção de texto** necessita, ao contrário dos outros processos que podem, mesmo sem uma análise prévia do estado da imagem, melhorar o reconhecimento de texto; desta análise visto que por exemplo, não se pode aplicar uma rotação na imagem sem saber o ângulo de orientação inicial desta. Bieniecki et al. [2007] faz uma apresentação convincente do efeito de uma rotação de ângulo 15° no reconhecimento do Tesseract, não tendo qualquer acerto. O método proposto passa pela computação de

uma linha que afete a margem na extremidade esquerda do texto, de modo a calcular a sua inclinação relativa à margem da imagem e assim descobrir o ângulo de rotação do texto. No espectro mais limitado da sua proposta, devido à sua localidade nos documentos, [Bieniecki et al. \[2007\]](#) discute distorções nos documentos como curvaturas resultantes da bainha de um livro. Aqui, em traços gerais, a linha de curvatura do texto é detetada, com a qual é criado um quadrilátero da área afetada onde será, de acordo com o nível de curvatura na projeção sobre a linha, realizada a correção. Em ambos os casos, os resultados demonstrados para casos de grande deformação, os algoritmos propostos conseguiram tornar a completa falha de reconhecimento para taxas de acerto dentro dos 99%.

A aplicação de **super-resolução** procura auxiliar o processo de reconhecimento ao melhorar a qualidade de imagens de baixa resolução, i.e. aumentar os seus **dpi** e tornar os caracteres mais reconhecíveis. Entre os vários algoritmos utilizados para este propósito, o uso de interpolação tendia a ser o mais comum, porém nem sempre os resultados eram satisfatórios, resultando em imagens transformadas desfocadas, ou com os defeitos originais acentuados, especialmente quando o salto era feito a partir de imagens com **dpi** baixo - 100 para baixo [\[Lat and Jawahar, 2018\]](#). No entanto, com os avanços na área das redes neuronais, em particular na categoria de imagens naturais, modelos como as **CNN** [\[Wei et al., 2018\]](#), [\[Lat and Jawahar, 2018\]](#) trouxeram uma nova forma treinar algoritmos para tratar imagens de forma adaptativa e com resultados muito melhores do que os algoritmos bem estabelecidos para este problema. Um dos pontos negativos deste tipo de redes é que a criação de datasets de treino é um processo demorado, sendo que para cada imagem de treino (degradada), é necessário uma imagem par com o resultado ideal para validação do resultado. Adicionalmente estes datasets têm de ter casos com características dispersas o suficiente para permitir uma boa generalização do modelo. Um outro modelo que tem vindo a emergir, são as **GAN** que invés de utilizarem uma única rede para gerar conteúdo que depois será validado em cada iteração do treino, utilizam duas redes que competem diretamente; a geradora que tenta transformar imagens de modo a enganar o discriminador que tenta entender se a imagem de input é a imagem original ou se foi gerada. [Souibgui and Kessentini \[2022\]](#) propõe um modelo deste género que demonstra a sua superioridade tanto dos algoritmos baseados em regras, como de modelos baseados em **CNN**.

As **transformações morfológicas** são compostas por vários métodos e interesses diferentes, muitas vezes não com o propósito de melhorar a qualidade da imagem, mas para acentuar certas características desta. Por exemplo, técnicas como a dilatação podem ser utilizadas para acentuar regiões de texto de forma a ser possível separar o texto do resto. Por outro lado técnicas de detecção de arestas, erosão ou *thinning*, diminuem o tamanho dos elementos da imagem, podendo simplificar os caracteres tornando o

seu reconhecimento, ou das suas características (como loops) mais evidentes [Dey et al., 2022].

Estes diferentes tipos de tratamento podem, na grande maioria dos casos, complementar-se mutuamente e é costume - inclusive nos estudos referenciados - a criação de pipelines de pré processamento que aplicam estes vários tratamentos de forma sequencial. No entanto, como estes diferentes tratamentos impactam diretamente os dados de input para reconhecimento, nem sempre são benéficos e têm de ser escolhidos com cuidado consoante o estado do sujeito. Bui et al. [2017] demonstra precisamente isto mostrando, por exemplo, que a aplicação de um filtro Gaussiano para a redução de ruído num caso de teste reduziu a taxa de acerto do Tesseract para menos de 1 terço comparado ao resultado sem pré processamento. Isto naturalmente dificulta a criação de pipelines automáticas de pré processamento. Nesse mesmo estudo, é proposto o uso de uma CNN que consoante um número limitado de classes que representam conjuntos de técnicas de pré processamento, decide a melhor para uma dada imagem. Esta solução resultou numa melhoria considerável, principalmente para o reconhecimento do Tesseract e, mais interessante, a tendência para certas combinações de técnicas com: binarização escolhida 90% das vezes, redução de ruído 35% e acentuação de contrastes 34%. Como mencionado anteriormente, os avanços no tratamento de imagem com uso de modelos de Deep Learning vem trazer, quando suficientemente generalizados, um método ubíquo para a realização destes vários tratamentos de forma adaptativa. Souibgui and Kessentini [2022] com a GAN proposta, demonstra resultados no tratamento de ruído, focagem, binarização e remoção de *watermarks* excelentes, mesmo tendo em conta que o foco principal do modelo era o aumento da resolução da imagem original.

### **2.2.2 Pós Processamento para OCR**

### **2.2.3 Identificação de imagens**

### **2.2.4 Segmentação de documentos**

### **2.2.5 Ordem de leitura**

## Capítulo 3

# 0 problema e os seus desafios

0 problema e os seus desafios

### 3.1 Imagens

Exemplo de inserção de uma imagem como texto exibido,



— dentro no texto, bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla  
bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla  
bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla  
bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla  
bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla  
bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla bla-bla  
— ou em formato flutuante



Figura 1: Legenda



## **Parte II**

### **Core da Dissertação**

## **Capítulo 4**

### **Contribuição**

Principais resultados e as suas evidências científicas.

#### **4.1 Introdução**

#### **4.2 Sumário**

## **Capítulo 5**

### **Aplicações**

Aplicação do resultado principal (exemplos e casos de estudo)

#### **5.1 Introdução**

#### **5.2 Sumário**

## **Capítulo 6**

### **Conclusões e trabalho futuro**

Conclusões e trabalho futuro.

#### **6.1 Conclusões**

#### **6.2 Perspetiva de trabalho futuro**

## Capítulo 7

# Planeamento

### 7.1 Atividades

Tarefa	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul
<i>Background</i> e EA	•	•	•							
Preparação do RPD		•	•	•						
Contribuição				•	•	•	•	•	•	
Escrita							•	•	•	•

Tabela 1: Plano de atividades.

## Bibliografia

Keras ocr documentação. URL <https://keras-ocr.readthedocs.io/en/latest/examples/index.html>.

Paddleocr documentação. URL <https://github.com/PaddlePaddle/PaddleOCR>.

Tesseract documentação. URL <https://tesseract-ocr.github.io>.

Deepa Berchmans and S S Kumar. Optical character recognition: An overview and an insight. In *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, pages 1361–1365, 2014. doi: 10.1109/ICCICCT.2014.6993174.

Wojciech Bieniecki, Szymon Grabowski, and Wojciech Rozenberg. Image preprocessing for improving ocr accuracy. In *2007 International Conference on Perspective Technologies and Methods in MEMS Design*, pages 75–80, 2007. doi: 10.1109/MEMSTECH.2007.4283429.

Quang Anh Bui, David Mollard, and Salvatore Tabbone. Selecting automatically pre-processing methods to improve ocr performances. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 169–174, 2017. doi: 10.1109/ICDAR.2017.36.

Mostafa Darwiche, The-Anh Pham, and Mathieu Delalandre. Comparison of jpeg’s competitors for document images. In *2015 International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 487–493, 2015. doi: 10.1109/IPTA.2015.7367194.

Raghunath Dey, Rakesh Chandra Balabantaray, Surajit Mohanty, Debabrata Singh, Marimuthu Karuppiah, and Debabrata Samanta. Approach for preprocessing in offline optical character recognition (ocr). In *2022 Interdisciplinary Research in Technology and Management (IRTM)*, pages 1–6, 2022. doi: 10.1109/IRTM54583.2022.9791698.

Ankit Lat and C. V. Jawahar. Enhancing ocr accuracy with super resolution. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3162–3167, 2018. doi: 10.1109/ICPR.2018.8545609.

- Laurence Likforman-Sulem, Jérôme Darbon, and Elisa H. Barney Smith. Pre-processing of degraded printed documents by non-local means and total variation. In *2009 10th International Conference on Document Analysis and Recognition*, pages 758–762, 2009. doi: 10.1109/ICDAR.2009.210.
- Rishabh Mittal and Anchal Garg. Text extraction using ocr: A systematic review. In *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 357–362, 2020. doi: 10.1109/ICIRCA48905.2020.9183326.
- Mohamed Ali Souibgui and Yousri Kessentini. De-gan: A conditional generative adversarial network for document enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1180–1191, 2022. doi: 10.1109/TPAMI.2020.3022406.
- Sargur N. Srihari, Ajay Shekhawat, and Stephen W. Lam. *Optical Character Recognition (OCR)*, page 1326–1333. John Wiley and Sons Ltd., GBR, 2003. ISBN 0470864125.
- Tan Chiang Wei, U. U. Sheikh, and Ab Al-Hadi Ab Rahman. Improved optical character recognition with deep neural network. In *2018 IEEE 14th International Colloquium on Signal Processing Its Applications (CSPA)*, pages 245–249, 2018. doi: 10.1109/CSPA.2018.8368720.

## **Parte III**

### **Apêndices**



## **Apêndice A**

### **Trabalho de apoio**

Resultados auxiliares.

## **Apêndice B**

### **Detalhes dos resultados**

Detalhes de resultados cuja extensão comprometeria a legibilidade do texto principal.

## **Apêndice C**

### **Listings**

Se for o caso.

## Apêndice D

# Ferramentas

(Se for o caso)

Utilizadores de [L<sup>A</sup>T<sub>E</sub>X](#) devem consultar [TUG](#) , o grupo de utilizadores [T<sub>E</sub>X](#) .



Coloque aqui informação sobre financiamento, projeto FCT, etc. em que o trabalho se enquadra. Deixe em branco caso contrário.