

UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

Otávio Oliveira Deon

**AUTOMATIZANDO A EXPORTAÇÃO DE QUESTÕES DE
PROVAS DA OLIMPÍADA BRASILEIRA DE INFORMÁTICA
POR MEIO DE FERRAMENTAS DE EXTRAÇÃO DE
TEXTO E VISÃO COMPUTACIONAL**

Santa Maria, RS
2018

Otávio Oliveira Deon

**AUTOMATIZANDO A EXPORTAÇÃO DE QUESTÕES DE PROVAS DA OLIMPÍADA
BRASILEIRA DE INFORMÁTICA POR MEIO DE FERRAMENTAS DE EXTRAÇÃO
DE TEXTO E VISÃO COMPUTACIONAL**

Trabalho de Conclusão de Curso apresentado
ao Curso de Ciência da Computação da Uni-
versidade Federal de Santa Maria (UFSM, RS),
como requisito parcial para a obtenção do grau
de **Bacharel em Ciência da Computação**

Orientadora: Prof^ª. Dr^ª. Andrea Schwertner Charão

446
Santa Maria, RS
2018

Otávio Oliveira Deon

**AUTOMATIZANDO A EXPORTAÇÃO DE QUESTÕES DE PROVAS DA OLIMPÍADA
BRASILEIRA DE INFORMÁTICA POR MEIO DE FERRAMENTAS DE EXTRAÇÃO
DE TEXTO E VISÃO COMPUTACIONAL**

Trabalho de Conclusão de Curso apresentado
ao Curso de Ciência da Computação da Uni-
versidade Federal de Santa Maria (UFSM, RS),
como requisito parcial para a obtenção do grau
de **Bacharel em Ciência da Computação**

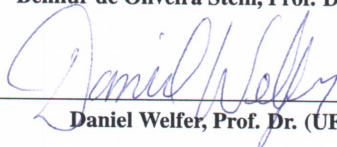
Apresentação em 03 de Dezembro de 2018:



Andrea Schwertner Charão, Dr^a.
(Presidente/Orientadora)



Benhur de Oliveira Stein, Prof. Dr. (UFSM)



Daniel Welfer, Prof. Dr. (UFSM)

Santa Maria, RS

2018

AGRADECIMENTOS

Agradeço aos meus familiares e amigos, que sempre me apoiaram de diversas formas e me ajudaram a crescer pessoal e academicamente.

Agradeço também minha professora e orientadora Andrea Charão, que foi parte importante deste trabalho e me ajudou sempre que possível, demonstrando disponibilidade e interesse. Devo agradecimentos também aos professores Benhur Stein e Daniel Welfer, membros da banca. Através de suas críticas e sugestões na sessão de prévia, contribuíram para o progresso do trabalho.

RESUMO

AUTOMATIZANDO A EXPORTAÇÃO DE QUESTÕES DE PROVAS DA OLIMPÍADA BRASILEIRA DE INFORMÁTICA POR MEIO DE FERRAMENTAS DE EXTRAÇÃO DE TEXTO E VISÃO COMPUTACIONAL

AUTOR: OTÁVIO OLIVEIRA DEON

ORIENTADORA: ANDREA SCHWERTNER CHARÃO

A extração de conteúdo de imagens e documentos digitais pode ser feita automaticamente com auxílio de ferramentas computacionais existentes. Entretanto, nem sempre basta aplicar uma dessas ferramentas sobre o documento para obter o conteúdo desejado. A estrutura do objeto em questão pode exigir uma customização das ferramentas para que a extração seja feita corretamente. Este trabalho tem o objetivo de estudar e utilizar ferramentas de visão computacional e extração de texto para extrair automaticamente questões de provas da Olimpíada Brasileira de Informática de maneira organizada.

Palavras-chave: Extração de texto. Olimpíada Brasileira de Informática.

ABSTRACT

AUTOMATING THE EXPORT OF BRAZILIAN INFORMATICS OLYMPIAD EXAMS' QUESTIONS WITH TEXT EXTRACTION AND COMPUTER VISION TOOLS

**AUTHOR: OTÁVIO OLIVEIRA DEON
ADVISOR: ANDREA SCHWERTNER CHARÃO**

Content extraction from digital images and documents can be achieved automatically with existing digital tools. However, the document's structure may require some customization of those tools in order to correctly extract the data. This work aims to study and apply both computer vision and text extraction tools to automatically extract questions from Brazilian Informatics Olympiad exams in an organized way.

Keywords: Text extraction. Brazilian Informatics Olympiad.

LISTA DE FIGURAS

Figura 5.1 – Questão da prova da OBI de 2004 na modalidade Iniciação	22
Figura 5.2 – Recorte de texto extraído com a ferramenta pdftotext	23
Figura 5.3 – Recorte de texto extraído com problemas	23
Figura 5.4 – Recorte de texto extraído com a ferramenta pdftotext e a <i>flag</i> -layout	24
Figura 5.5 – Recorte de página a ser testada com pdftotext e <i>flag</i> -layout	24
Figura 5.6 – Resultado problemático de extração com a <i>flag</i> -layout	24
Figura 5.7 – Recorte de texto extraído com a ferramenta pdftotext e a <i>flag</i> -raw	25
Figura 5.8 – Recorte de extração de texto com a ferramenta pdfminer	26
Figura 5.9 – Recorte de extração de texto com a ferramenta Tesseract	27
Figura 5.10 – Recorte de questão	27
Figura 5.11 – Extração de texto sobre imagem com 300dpi	27
Figura 5.12 – Extração de texto sobre imagem com 900dpi	28
Figura 5.13 – Questão da prova de 2015 (1ª fase, 2º nível)	30
Figura 5.14 – <i>Workflow</i> da biblioteca OCRopus	31
Figura 5.15 – Imagem de entrada para o OCRopus	32
Figura 5.16 – Imagem de entrada para o OCRopus	33
Figura 5.17 – Erro na segmentação de texto	33
Figura 5.18 – Recorte da interface da ferramenta LAREX após segmentação automática ..	34
Figura 5.19 – Interface do programa	37
Figura 6.1 – Segmentação de prova de 2008 apresentada para o usuário	38
Figura 6.2 – Segmentação de prova de 2004 apresentada para o usuário	39
Figura 6.3 – Recorte de segmentação de prova de 2005 apresentada para o usuário	40
Figura A.1 – Questão da prova de 2016 (1ª fase, 1º nível)	45
Figura A.2 – Extração de texto de questão com a ferramenta pdftotext e <i>flag</i> -layout	46
Figura A.3 – Questão da prova de 2015 (1ª fase, 2º nível)	47
Figura A.4 – Extração de texto de questão com a ferramenta pdftotext e <i>flag</i> -raw	47
Figura A.5 – Extração de texto de questão com a ferramenta pdfminer	47
Figura A.6 – Recorte de questão da prova de 2005 (1º nível, fase única)	48
Figura A.7 – Extração da questão "Florista" com a ferramenta pdftotext e <i>flag</i> -layout	48
Figura A.8 – Extração de texto da questão "Florista" com a ferramenta pdftotext e <i>flag</i> -raw	48

LISTA DE TABELAS

Tabela 5.1 – Comparação entre ferramentas de extração de texto	28
Tabela 5.2 – Resultados de extrações sobre páginas inteiras da amostra	29
Tabela 5.3 – Aplicação da pdftotext -raw sobre todas provas do <i>dataset</i>	30

LISTA DE APÊNDICES

APÊNDICE A – EXEMPLOS DO USO DE FERRAMENTAS DE EXTRAÇÃO DE TEXTO SOBRE PROVAS DA OBI	45
---	----

SUMÁRIO

1 INTRODUÇÃO	10
1.1 OBJETIVOS	11
1.1.1 Objetivo Geral.....	11
1.1.2 Objetivos Específicos	11
1.2 JUSTIFICATIVA.....	11
2 TRABALHOS RELACIONADOS	13
3 METODOLOGIA	15
4 REFERENCIAL TEÓRICO	17
4.1 VISÃO COMPUTACIONAL	17
4.2 RECONHECIMENTO ÓTICO DE CARACTERES	18
5 DESENVOLVIMENTO	20
5.1 MATERIAL E FERRAMENTAS	20
5.2 TESTES DE EXTRAÇÃO DE TEXTO DE DOCUMENTOS PDF	22
5.3 SEGMENTAÇÃO DE PÁGINAS	30
5.4 LAREX	33
5.5 SOLUÇÃO DESENVOLVIDA	35
6 RESULTADOS E DISCUSSÕES	38
7 CONCLUSÃO	41
REFERÊNCIAS	42
APÊNDICES	44

1 INTRODUÇÃO

A Olimpíada Brasileira de Informática, sigla OBI, é uma competição organizada pela Sociedade Brasileira de Computação (SBC) que visa aumentar o interesse de jovens brasileiros pela ciência da computação, através da resolução de provas com problemas de raciocínio lógico.

A OBI acontece desde 1999 e é dividida em duas modalidades: Iniciação e Programação. Na modalidade Iniciação, os competidores concorrem solucionando problemas de lógica utilizando apenas lápis e papel. Na modalidade Programação, deve-se solucionar problemas através de programas e algoritmos criados na hora, com uso de computadores.

O Clube da Computação, programa de extensão da Universidade Federal de Santa Maria (UFSM), possui um projeto que oferece treinamento e uma sede local para a Modalidade Iniciação da OBI. O treinamento é feito com o uso de provas antigas da competição, que são disponibilizadas em formato PDF.

A utilização das provas para fins de estudo pode ser beneficiada com uma organização sistemática das questões. Para isso, é necessário extraí-las das provas individualmente, possibilitando uma apresentação organizada das questões.

A extração automática das questões pode ser uma atividade desafiadora, já que elas não são dispostas nas páginas das provas de maneira uniforme. As questões populam as páginas de maneiras variadas, por exemplo, com alternância entre um *layout* em 1 ou 2 colunas. Além disso, diferentes questões podem conter seus textos muito próximos uns aos outros, assim como podem conter figuras necessárias à resolução. Essas características exigem uma leitura computacional apropriada para que a extração seja feita adequadamente.

A área de estudo do trabalho é ampla e se divide em vários segmentos. A conferência internacional ICDAR (INTERNATIONAL CONFERENCE ON DOCUMENT ANALYSIS AND RECOGNITION, 1991) reúne pesquisadores da área de análise de documentos para apresentar trabalhos sobre processamento de imagens, reconhecimento de texto, aprendizado de máquina para análise de documentos, e outros tópicos específicos. A conferência ainda organiza uma série de competições que avaliam métodos e algoritmos relacionados à análise de documentos e imagens. As categorias das competições incluem reconhecimento de manuscritos, reconstrução de documentos, análise de manuscritos históricos, análise forense, etc.

Atualmente, existem diversas ferramentas para extração automática de texto de imagens

e documentos PDF, como Online OCR¹. Além disso, bibliotecas de visão computacional, como a OpenCV², permitem a análise e extração de informação de imagens de maneira mais aprofundada.

Levando em conta as particularidades da estruturação textual das folhas de provas da OBI, a utilização de ferramentas desse tipo, em conjunto, poderá viabilizar a extração correta das questões da competição.

A obtenção automática das questões de forma estruturada e inteligente resultará em uma organização útil, podendo ser estendida também às próximas edições da competição. As extrações poderão ser armazenadas em um banco de dados para buscas futuras, por exemplo.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Este trabalho teve o objetivo de extrair de forma automática as questões das provas da modalidade Iniciação da OBI, desde o início até o presente ano, com a ajuda de ferramentas de extração de texto e visão computacional. O conteúdo extraído pode ser aproveitado da maneira que o usuário desejar - guardar as questões em um banco de dados, de forma organizada, pode ser útil para gerar novas provas, por exemplo.

1.1.2 Objetivos Específicos

- Pesquisar e estudar ferramentas que possam ser úteis na leitura e análise das provas;
- Adequar as ferramentas existentes para otimizar o uso sobre as provas;
- Aplicar o método encontrado para extrair corretamente as questões, viabilizando uma apresentação organizada das mesmas.

1.2 JUSTIFICATIVA

A Olimpíada Brasileira de Informática disponibiliza *online* as provas da modalidade Iniciação de todas edições, mas não é possível acessar cada questão individualmente. A organização e a análise de cada questão da competição é útil para fins de estudo, por exemplo.

¹ <https://www.onlineocr.net/pt/>

² <https://opencv.org/>

Com ajuda de Jogos de Lógica (MARTINS, 2011), livro escrito especialmente para estudos para a OBI, é facilitado o preparatório para a realização das provas da competição. O livro, disponível gratuitamente, descreve os tipos de problemas que constituem as provas, explicando técnicas e métodos de resolução que são muito úteis na etapa de estudos. O leitor do livro acaba também aprimorando habilidades de interpretação e dedução lógica. Porém, além do livro e do material disponível no site oficial da competição, não se observa atualmente outros meios de preparação para a OBI.

Os problemas apresentados nas provas da competição contêm diferentes características e podem ser classificados em diferentes tipos. Uma questão pode ter atributos de ordenação, agrupamento ou cálculo, por exemplo. Durante os estudos prévios à realização de uma prova, o aluno pode ter resultados mais satisfatórios se levar em conta essas particularidades. É possível, assim, treinar apenas questões de um tipo específico que esteja proporcionando mais dificuldades. Organizando melhor os estudos, o aluno poderá acertar mais questões na prova.

Tendo em vista a possibilidade de extrair questões e arranjá-las de maneira sistemática, será oferecida, para os competidores ou outros interessados, a oportunidade de aprimorar o método de estudo. Será possível obter questões baseando-se em seu tipo, nível, ano em que foi publicada, etc. Também cria-se a oportunidade de gerar novas provas, reunindo questões de forma arbitrária. Essas possibilidades oferecerão novos recursos para quem se interessa em se aprofundar na olimpíada.

Entretanto, a tarefa de extrair automaticamente as questões das provas não é trivial, considerando a estrutura das páginas que muitas vezes exige atenção para distinguir corretamente as questões, seus títulos, enunciados, alternativas, figuras e até múltiplas questões que se referem a um só enunciado.

Com a ajuda de programas e métodos da área de visão computacional, é possível percorrer as páginas das provas aplicando métodos morfológicos para segmentar suas regiões, para então realizar a extração das questões de forma correta e automatizada.

2 TRABALHOS RELACIONADOS

Em (S.SHIYAMALA, 2017) é feita uma revisão de técnicas relacionadas à detecção e extração de texto em imagens e vídeos. O trabalho apresenta, em detalhes, tendências e possíveis direções de pesquisas da área. São descritos três principais tipos de métodos de detecção de texto.

O método baseado em região parte do princípio que há pouca variação de cor no texto, e essa cor é distinta do fundo da imagem. O texto é obtido com o *threshold* da imagem entre a cor do texto e do fundo. Esse método pode ser dividido em duas abordagens:

- baseado em componentes conectados: pequenos componentes são agrupados em pedaços maiores até que todas regiões da imagem sejam identificadas. A abordagem localiza o texto rapidamente, mas falha com planos de fundo complexos.
- baseado em contornos: é focado em altos contrastes entre o texto e o plano de fundo, e é efetivo em localizar texto tanto em documentos como em imagens (interiores e exteriores). Não trabalha bem com fontes de grande tamanho.

O método baseado em textura se beneficia das propriedades texturais que distinguem o texto do fundo. Para isso, são usadas técnicas baseadas na Transformada rápida de Fourier, máquina de vetores de suporte (1999), filtros de Gabor (1988) etc. Nesse caso, o texto é detectado mesmo com planos de fundo complexos. Porém, há grande complexidade computacional na classificação de textura.

Já o método baseado em morfologia extrai características (traços) importantes do texto. Essas características são invariantes mesmo com mudanças na translação, rotação ou escala da imagem. Portanto, o método funciona bem com alterações significativas na imagem.

O trabalho de (CATTONI et al., 1998) descreve e compara informações sobre o estado de pesquisa em imagens de documentos. Algoritmos propostos na literatura são apresentados, com foco em estimativa de ângulo de inclinação e decomposição de páginas. Diferentes abordagens de segmentação de texto, segmentação de páginas e classificação de blocos são descritas e comparadas. Por exemplo, diversos algoritmos que utilizam componentes conectados são comparados em relação às suas suposições, limitações e características. Alguns processam adequadamente imagens com ruídos no plano de fundo; outros suportam diferentes orientações do texto.

Em (ZIRARI et al., 2013), é proposto um método de separação de componentes textuais e não-textuais em documentos. A abordagem utilizada pelos autores baseia-se na modelagem da imagem através de grafos, e na aplicação de regras de estruturação do *layout*. Os autores defendem que a representação da imagem com grafos permite a inclusão de informações espaciais no modelo. A primeira etapa do método extrai os componentes conectados da imagem. Para isso, é utilizado um grafo não-orientado cujos nós representam os pixels da imagem. As arestas com pesos representam as relações de conectividade e são balanceadas pela soma das intensidades dos pixels. No início do método, cada pixel da imagem é considerado como sua própria região. As regiões são incrementalmente fundidas - as arestas são percorridas em ordem crescente de peso, e caso um peso seja menor que a variação interna dos nós conectados pela aresta, as regiões são fundidas. Após obter o conjunto de componentes conectados, eles são classificados em elementos gráficos (figuras, tabelas, linhas) ou textuais. Para isso, os componentes são filtrados com base no alinhamento de caracteres de tamanho similar.

A conferência ICDAR organiza periodicamente competições de análise/extração de texto em imagens e documentos. Em (GATOS; STAMATOPOULOS; LOULLOUDIS, 2011) é descrita uma competição de métodos de segmentação de texto escrito à mão. Os métodos são testados para segmentação de linhas de texto e segmentação de palavras. O dataset usado para a avaliação dos métodos conteve 200 imagens. A métrica de performance das soluções desenvolvidas correspondeu à média entre taxa de detecções e acurácia do reconhecimento. Os melhores resultados de segmentação de linhas de texto e palavras foram, respectivamente, 99,53% e 94,77%.

3 METODOLOGIA

O trabalho consistiu de etapas de coleta de materiais, pesquisa e desenvolvimento.

Primeiramente, foi necessário obter as provas da modalidade Iniciação da OBI, que estão disponíveis na página oficial da competição e serviram como base de extração das questões. Visando velocidade e praticidade, essa obtenção foi feita através de um *script* que quando executado uma vez, retornou todas as provas disponíveis. Essa tarefa foi possível devido à disponibilização de maneira padronizada das provas.

Em seguida, foi realizada a pesquisa e o estudo de ferramentas que auxiliaram no desenvolvimento do trabalho. A biblioteca OpenCV e alguns métodos de extração de texto já eram conhecidos, mas procurou-se buscar outras ferramentas que pudessem ser úteis, priorizando aquelas eficientes em detectar e segmentar estruturas textuais dentro de um documento. Os métodos de OpenCV utilizados no trabalho acabaram sendo apenas os relativos à interface gráfica. No programa desenvolvido, páginas de provas e as regiões de texto segmentadas são exibidas para o usuário com ajuda da biblioteca. O tratamento dos cliques do mouse também é feito com OpenCV.

Com as provas e as ferramentas disponíveis, foram testadas diferentes maneiras de extração das questões. Foram desenvolvidos algoritmos em conjunto com as ferramentas, buscando os melhores resultados possíveis. Os métodos de extração foram aplicados sobre uma amostra das provas disponíveis. A amostra consistiu de provas de diversos anos, sendo possível realizar os testes englobando diferentes tipos de questões.

Entre os testes realizados, constam diferentes abordagens de extração. As primeiras ferramentas apenas obtêm o texto presente nos documentos PDF. Outras aplicam o reconhecimento óptico de caracteres sobre as imagens das provas. Para a segmentação e análise de *layout* das páginas, é utilizada uma abordagem *bottom-up*. É realizado o processo de dilatação das imagens, em que os pixels relativos ao texto são expandidos e agrupados em componentes conectados. Os componentes são considerados como regiões na página, evidenciando diferentes funções (enunciados, alternativas, etc). Essa estratégia é frequentemente utilizada em trabalhos de análise e extração de texto em imagens.

Os resultados de extração foram comparados com informações das questões previamente definidas manualmente. Foi verificado se os métodos foram capazes de distinguir corretamente diferentes questões, enunciados e alternativas, assim medindo a eficácia dos métodos de extra-

ção analisados.

4 REFERENCIAL TEÓRICO

4.1 VISÃO COMPUTACIONAL

Técnicas de processamento digital de imagens se referem à manipulação de uma imagem representada, no computador, por *pixels* (*picture elements*). Cada *pixel* tem um valor de intensidade. Normalmente se usa 1 byte por valor - por exemplo, o valor 0 representa a cor preta e o valor 255 representa a cor branca. As diferenças entre os valores dentro da matriz nos permite identificar o conteúdo das imagens.

Visão computacional é a área da computação que estuda e desenvolve tecnologias ligadas ao reconhecimento e classificação de estruturas presentes em imagens e vídeos. Através dessa área, criam-se métodos de aquisição, processamento e análise de imagens digitais, tornando possível a automação de tarefas que antes eram realizadas apenas com ajuda do olho humano.

O escopo da aplicação de tecnologias desenvolvidas com visão computacional é amplo. Na medicina, por exemplo, extrai-se informação de imagens para realizar diagnósticos sobre pacientes. Em indústrias, processos como controle de qualidade e cálculo de posição e orientação para máquinas e componentes robóticos são auxiliados pela visão computacional. A área também tem importância crítica quando se trata de veículos autônomos, pois realiza tarefas como reprodução do ambiente real no campo digital e detecção de eventos e obstáculos.

Dependendo da aplicação, antes de um método ser aplicado, é necessário realizar um pré-processamento sobre a imagem. Ela é transformada, com o objetivo de melhorar a acurácia das etapas posteriores. Os principais métodos de pré-processamento incluem a remoção de ruído das imagens (eliminação de elementos como manchas em pedaços de papel, que podem adicionar ou ocultar informações no resultado final), a correção de orientação, como a inclinação de documentos escaneados, e o realce de contraste, que facilita a detecção de elementos na imagem.

As principais funções na manipulação de imagens são a detecção de linhas, bordas, formas e texturas, filtragem de regiões de interesse, reconhecimento de padrões, tratamentos de cor, etc. Ao se trabalhar com documentos de texto, diversas funções serão empregadas para alcançar o objetivo final.

O principal objetivo de um módulo de análise de *layout* de um documento é identificar as

diversas regiões físicas no documento, assim como suas características (NAMBOODIRI; JAIN, 2007). A análise pode ser feita seguindo uma abordagem *top-down*, *bottom-up*, ou híbrida. Na *top-down*, a análise parte da imagem inteira e a divide repetidamente em regiões menores. Essas regiões, dependendo do documento, podem ser figuras, palavras, componentes conectados, etc. A abordagem *bottom-up* começa com os elementos primitivos (pixels, por exemplo) e os agrupa em regiões maiores, como palavras, linhas, parágrafos, etc. A abordagem híbrida mistura essas estratégias.

O algoritmo *X-Y Cut* (NAGY; SETH; VISWANATHAN, 1992) é do tipo *top-down*. Ele divide a imagem em seções baseando-se nos seus perfis de projeção. As regiões da segmentação são projetadas alternadamente nos eixos horizontal e vertical. O algoritmo particiona o documento até que as regiões sejam consideradas atômicas.

O algoritmo *Docstrum* (O'GORMAN, 1993) é um exemplo que usa a abordagem *bottom-up*. Ele detecta uma série de componentes conectados como primitivos. Os k-vizinhos mais próximos são identificados e linhas de texto são formadas. Para verificar a distância entre caracteres e entre palavras, são usados histogramas dos componentes, assim formando blocos de texto.

4.2 RECONHECIMENTO ÓTICO DE CARACTERES

O termo OCR (*Optical Character Recognition*) se refere ao reconhecimento e conversão de texto presente em imagens para texto codificado em um computador. As imagens em questão podem corresponder a fotos contendo textos de qualquer natureza, documentos escaneados, etc. Portanto, o alcance da aplicação de OCR é amplo - a técnica pode ser usada para reconhecer placas de automóveis, guardar informações de recibos, possibilitar a edição de conteúdo de documentos impressos, etc.

Programas que realizam OCR costumam pré-processar a entrada para aumentar as chances de reconhecer os caracteres corretamente. A entrada pode passar por várias etapas, como a correção da orientação do texto (deixando-o totalmente no sentido horizontal), remoção de linhas de tabelas, binarização (conversão de imagem colorida ou em tons de cinza para preto e branco - com apenas 2 cores, facilita-se a distinção entre texto e plano de fundo, além da detecção de bordas), e análise de *layout* (identificação de parágrafos, colunas e outros elementos).

Há duas principais maneiras de realizar o processo de reconhecimento dos caracteres. A primeira, correspondência de matrizes, converte os caracteres em padrões dentro de uma matriz,

e os compara com índices de caracteres já conhecidos. Não funciona bem quando são analisadas diferentes fontes de texto, por exemplo.

A segunda forma, detecção de traços, é mais complexa e versátil. Ela se baseia em identificar linhas, curvas e particularidades morfológicas dos caracteres, em vez de analisá-los "como um todo". Por exemplo, a letra "A" é vista como duas linhas inclinadas que se encontram no topo, com uma linha horizontal ligando-as no meio. Essa estratégia permite o reconhecimento mesmo sobre diferentes fontes ou estilos de escrita. O algoritmo *k-nearest neighbors* (DESARATHY, 1991) é frequentemente usado. Ele usa uma base de dados e classifica esses dados em diferentes "classes", para prever a classificação de uma nova entrada com base em suas semelhanças. Alguns programas ainda aplicam redes neurais para o reconhecimento de caracteres seguindo essa abordagem.

5 DESENVOLVIMENTO

5.1 MATERIAL E FERRAMENTAS

A primeira etapa do trabalho consistiu na obtenção de todas provas da modalidade Iniciação da Olimpíada Brasileira de Informática até o presente ano. As provas são disponibilizadas na página da OBI³ de maneira organizada, facilitando a obtenção. Para agilizar o processo, evitando trabalho manual, foi feito um *script* na linguagem Python. Os links para as provas são gerados de forma padronizada, possibilitando o acesso por um algoritmo. Quando executado, o *script* realiza um laço de repetição que a cada iteração, incrementando no link o ano, nível e fase, acessa a página *web* que contém a prova e automaticamente faz o *download* dela para o computador. Não foi possível obter uma prova, a primeira da modalidade Iniciação, de 2002. É informado na página da OBI que neste ano foi realizada a prova da modalidade, em uma fase e um nível, mas o link para o arquivo não existe. Totalizou-se então 52 provas obtidas.

Com o montante disponível localmente, foi necessário montar uma amostra para a realização dos subseqüentes testes. O processo de filtragem de todas as provas disponíveis para um conjunto de interesse foi feito manualmente, devido à necessidade de reunir na amostra provas com questões de diferentes tipos e *layouts*. A limitação de experimentações de um novo método de extração de texto sobre questões similares poderia resultar em resultados ruins, quando o método fosse aplicado a questões de moldes diferentes. Portanto, as 52 provas foram analisadas a fim de selecionar um conjunto heterogêneo de questões para compor a amostra - a variação de formatos foi priorizada. Notou-se a existência de diferentes "gerações" de provas. Por exemplo, nas provas de 2003 e 2004 da modalidade Iniciação da competição observaram-se *layouts* mais simples do que no resto. Suas questões são estruturadas de maneira uniforme, sempre uma abaixo da outra, não contendo páginas com mais de uma coluna de texto. Além disso, suas questões têm seus enunciados precedidos apenas pelo número da questão - já as questões do resto das provas têm o enunciado precedido pela palavra "Questão" e o número relativo à mesma. Nota-se ainda que na prova de 2003 consta, abaixo de cada questão, a solução da mesma com explicações. Evitou-se selecionar mais de uma prova aplicada em um certo ano, pois eventualmente repetem-se questões de níveis diferentes num mesmo ano. Provas com imagens incluídas nas questões também foram "priorizadas", pois apesar das ferramentas não

³ <https://olimpiada.ic.unicamp.br/>

extraírem essas imagens, previu-se que elas bagunçariam o conteúdo obtido. Por fim, foram selecionadas 9 provas para compor a amostra, abrangendo edições entre 2003 e 2016.

Dentre as tecnologias conhecidas para o trabalho, a biblioteca OpenCV (*Open Source Computer Vision Library*) pertence à área de Visão Computacional e pode ser aplicada sobre imagens e vídeos, suportando programas escritos nas linguagens C++, Python e Java. Foi desenvolvida originalmente pela Intel, escrita em C/C++, é multi-plataforma e *open source*, podendo ser utilizada para fins acadêmicos e comerciais.

Áreas de aplicação da OpenCV incluem identificação de objetos, sistemas de reconhecimento facial, reconhecimento de movimentos, reconstrução 3D, realidade virtual, etc.

A utilização da OpenCV tem grande potencial no processamento das páginas compostas por texto e figuras, como é proposto no trabalho.

Tesseract⁴ é considerado um dos mais eficazes entre os *softwares* livres de OCR, sendo desenvolvido originalmente pela *Hewlett Packard*, e desde 2006 pela *Google*. Ele é capaz de reconhecer textos em mais de 100 linguagens e suporta diversos formatos de saída. A versão mais recente do Tesseract até o momento (4.0) possui o funcionamento de OCR baseado em uma rede neural LSTM⁵, focado em reconhecimento de linhas.

As versões de OpenCV e Tesseract instaladas no computador em que foi realizado o trabalho são, respectivamente, 3.4.2 e 3.04.01.

Foram pesquisadas ferramentas de extração de texto para serem experimentadas sobre a amostra. Essa pesquisa resultou em várias opções com a funcionalidade pretendida. As opções não executáveis em terminal, ou que não pudessem ser executadas através de automatização, foram filtradas. Essa restrição foi acolhida devido ao extenso número de provas e testes que viriam a ser realizados - o uso de serviços *online*, por exemplo, que necessitam do *upload* manual dos arquivos a serem analisados, resultaria em um trabalho custoso e limitado.

Foram selecionadas 3 ferramentas para um primeiro teste de extração de texto da amostra. Para cada ferramenta foi escrito um *script* na linguagem Python com o intuito de automatizar a extração, incluindo eventuais customizações de execução (ajustes de parâmetros). Além disso, considerando que todas as provas contêm a capa e nenhuma questão na primeira página, um *script* foi responsável por removê-la das provas da amostra a fim de facilitar a análise dos resultados. Os resultados das execuções são apresentados com imagens, comparando as questões originais na prova e as extrações obtidas.

⁴ <https://github.com/tesseract-ocr/>

⁵ <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

5.2 TESTES DE EXTRAÇÃO DE TEXTO DE DOCUMENTOS PDF

A primeira ferramenta testada foi pdftotext⁶. Deve-se observar que ela não realiza o processo de OCR para extrair o texto do documento - em vez disso, ela apenas obtém o texto embutido no PDF. A ferramenta foi executada, sobre as páginas da amostra, em 3 modos (com diferentes parâmetros). Primeiramente, experimentou-se a execução "pura", sem configurações. Os documentos em formato PDF resultaram em arquivos de texto. Um resultado (figura 5.2), exemplificado com a questão "Clara e Luiz"(figura 5.1), mostra-se satisfatório - o título da questão, o enunciado e as alternativas foram extraídos em ordem correta, de maneira compreensível. Outro ponto positivo da execução sem parâmetros é que páginas com *layout* formado com 2 colunas foram, em grande parte, extraídas corretamente, em ordem de leitura. No contexto do caso exemplificado, porém, deve-se observar que a questão utilizada é relativamente simples. O uso dessa ferramenta com essa configuração mostrou-se problemático em outras situações. Por exemplo, em algumas questões a descrição das alternativas apareceu abaixo das mesmas, como mostra a figura 5.3.

Clara e Luiz

13. Clara e Luiz estudam na mesma escola e cursaram as mesmas seis matérias durante o ano, mas apenas em uma das matérias – História do Brasil – tiraram a mesma nota. Em cada uma das matérias as notas variaram de 60 a 100.

Qual das seguintes afirmações permite que se deduza que a média das notas que Clara tirou nas outras cinco matérias foi maior que a média das notas de Luiz nessas mesmas cinco matérias?

- (A) A menor nota de Clara foi História do Brasil, mas a menor nota de Luiz foi Matemática.
- (B) A maior nota de Luiz foi maior do que a maior nota de Clara.
- (C) Clara teve notas maiores do que Luiz em três matérias.
- (D) A menor nota de Clara foi igual à maior nota de Luiz.
- (E) A menor nota de Luiz e a maior nota de Clara foram na mesma matéria.

14. Sempre que Luiz está ao ar livre e está fazendo sol, Luiz usa seus óculos escuros. Sempre que Luiz está ao ar livre e não está fazendo sol, Luiz carrega seus óculos escuros no bolso. Algumas vezes está fazendo sol quando Luiz não está ao ar livre.

Se as afirmações acima são verdadeiras, e Luiz não está usando seus óculos escuros, qual das afirmações abaixo deve também ser verdadeira:

- (A) Luiz está carregando seus óculos escuros no bolso.
- (B) Luiz não está ao ar livre.
- (C) Luiz não está ao ar livre e não está fazendo sol.
- (D) Luiz não está ao ar livre e/ou não está fazendo sol.
- (E) Luiz está ao ar livre e/ou não está fazendo sol.

Figura 5.1: Questão da prova da OBI de 2004 na modalidade Iniciação

A pdftotext ainda foi testada com a *flag* -layout, em que a extração é feita de modo

⁶ <https://linux.die.net/man/1/pdftotext>

Clara e Luiz

13. Clara e Luiz estudam na mesma escola e cursaram as mesmas seis matérias durante o ano, mas apenas em uma das matérias – História do Brasil – tiraram a mesma nota. Em cada uma das matérias as notas variaram de 60 a 100.

Qual das seguintes afirmações permite que se deduza que a média das notas que Clara tirou nas outras cinco matérias foi maior que a média das notas de Luiz nessas mesmas cinco matérias?

(A) A menor nota de Clara foi História do Brasil, mas a menor nota de Luiz foi Matemática.
 (B) A maior nota de Luiz foi maior do que a maior nota de Clara.
 (C) Clara teve notas maiores do que Luiz em três matérias.
 (D) A menor nota de Clara foi igual à maior nota de Luiz.
 (E) A menor nota de Luiz e a maior nota de Clara foram na mesma matéria.

14. Sempre que Luiz está ao ar livre e está fazendo sol, Luiz usa seus óculos escuros. Sempre que Luiz está ao ar livre e não está fazendo sol, Luiz carrega seus óculos escuros no bolso. Algumas vezes está Se as afirmações acima são verdadeiras, e Luiz não está usando seus óculos escuros, qual das afirmações abaixo deve também ser verdadeira:

(A) Luiz está carregando seus óculos escuros no bolso.
 (B) Luiz não está ao ar livre.
 (C) Luiz não está ao ar livre e não está fazendo sol.
 (D) Luiz não está ao ar livre e/ou não está fazendo sol.
 (E) Luiz está ao ar livre e/ou não está fazendo sol.

Figura 5.2: Recorte de texto extraído com a ferramenta pdftotext

Questão 16. Uma amostra que contenha o vírus Chicungunha e o vírus Dengue mas não contenha o vírus Zika adquirirá que sequência de cores, a primeira após o teste X ser aplicado, e a segunda após o teste Z ser aplicado?

(A)
 (B)
 (C)
 (D)
 (E)

laranja, roxo
 verde, roxo
 laranja, vermelho
 laranja, laranja
 verde, verde

Figura 5.3: Recorte de texto extraído com problemas

que o resultado, também no formato .txt, mantenha o máximo possível da estrutura espacial das páginas. Portanto, o arquivo de saída é bastante semelhante ao documento original, como mostrado na figura 5.4. Porém, isso pode gerar dificuldades no posterior tratamento das questões, já que nem todos resultados se mostraram satisfatórios, como observado nas figuras 5.5 e 5.6. Os enunciados de 2 questões, contendo palavras muito próximas, tornam-se emaranhados, dificultando a leitura.

- Clara e Luiz
13. Clara e Luiz estudam na mesma escola e cursaram as mesmas seis matérias durante o ano, mas apenas em uma das matérias – História do Brasil – tiraram a mesma nota. Em cada uma das matérias as notas variaram de 60 a 100. Qual das seguintes afirmações permite que se deduza que a média das notas que Clara tirou nas outras cinco matérias foi maior que a média das notas de Luiz nessas mesmas cinco matérias?
- (A) A menor nota de Clara foi História do Brasil, mas a menor nota de Luiz foi Matemática.
 (B) A maior nota de Luiz foi maior do que a maior nota de Clara.
 (C) Clara teve notas maiores do que Luiz em três matérias.
 (D) A menor nota de Clara foi igual à maior nota de Luiz.
 (E) A menor nota de Luiz e a maior nota de Clara foram na mesma matéria.
14. Sempre que Luiz está ao ar livre e está fazendo sol, Luiz usa seus óculos escuros. Sempre que Luiz está ao ar livre e não está fazendo sol, Luiz carrega seus óculos escuros no bolso. Algumas vezes está fazendo sol quando Luiz não está ao ar livre. Se as afirmações acima são verdadeiras, e Luiz não está usando seus óculos escuros, qual das afirmações abaixo deve também ser verdadeira:
- (A) Luiz está carregando seus óculos escuros no bolso.
 (B) Luiz não está ao ar livre.
 (C) Luiz não está ao ar livre e não está fazendo sol.
 (D) Luiz não está ao ar livre e/ou não está fazendo sol.
 (E) Luiz está ao ar livre e/ou não está fazendo sol.

Figura 5.4: Recorte de texto extraído com a ferramenta pdftotext e a *flag* -layout

- Questão 10.** Qual das seguintes opções abaixo contém uma ordem de saída da casa que poderia ser verdadeira em um dia em que os suspeitos entram na casa na ordem Mel, Juca, Kiko, Lia?
- (A) Juca, Kiko, Lia, Mel
 (B) Juca, Kiko, Mel, Lia
 (C) Kiko, Juca, Lia, Mel
 (D) Lia, Mel, Juca, Kiko
 (E) Mel, Kiko, Lia, Juca
- (D) Kiko e Mel
 (E) Lia e Mel
- Questão 13.** Em um dia em que Kiko entra na casa em segundo e Mel entra em terceiro, qual das seguintes afirmações é necessariamente verdadeira?
- (A) Juca é o primeiro a sair da casa.
 (B) Juca é o terceiro a sair da casa.
 (C) Kiko é o primeiro a sair da casa.
 (D) Lia é a terceira a sair da casa.
 (E) Mel é a segunda a sair da casa.

Figura 5.5: Recorte de página a ser testada com pdftotext e *flag* -layout

- Questão 10. Qual das seguintes opções abaixo contém uma ordem de saída da casa que poderia ser verdadeira em um dia em que os suspeitos entram na casa na ordem Mel, Juca, Kiko, Lia?
- (A) Juca, Kiko, Lia, Mel
 (B) Juca, Kiko, Mel, Lia
 (C) Kiko, Juca, Lia, Mel
 (D) Lia, Mel, Juca, Kiko
 (E) Mel, Kiko, Lia, Juca
- Questão 13. Em um dia em que Kiko entra na casa em segundo e Mel entra em terceiro, qual das seguintes afirmações é necessariamente verdadeira?
- (A) Juca é o primeiro a sair da casa.
 (B) Juca é o terceiro a sair da casa.
 (C) Kiko é o primeiro a sair da casa.
 (D) Lia é a terceira a sair da casa.
 (E) Mel é a segunda a sair da casa.

Figura 5.6: Resultado problemático de extração com a *flag* -layout

O último teste com a ferramenta em questão foi feito com a *flag* -raw. A descrição desse parâmetro é que o resultado da extração mantém a ordem de fluxo de conteúdo. Os testes mostraram que além de detectar 2 colunas das páginas e mostrá-las em ordem correta, essa execução não exibiu erros frequentes como nas outras execuções. Os enunciados e alternativas

foram extraídos corretamente, na grande maioria dos casos. Um exemplo é apresentado na figura 5.7.

```

Clara e Luiz
13. Clara e Luiz estudam na mesma escola e cursaram as mesmas seis matérias durante o ano,
mas apenas em uma das matérias – História do Brasil – tiraram a mesma nota. Em cada uma
das matérias as notas variaram de 60 a 100.
Qual das seguintes afirmações permite que se deduza que a média das notas que Clara tirou
nas outras cinco matérias foi maior que a média das notas de Luiz nessas mesmas cinco maté-
rias?
(A) A menor nota de Clara foi História do Brasil, mas a menor nota de Luiz foi Matemática.
(B) A maior nota de Luiz foi maior do que a maior nota de Clara.
(C) Clara teve notas maiores do que Luiz em três matérias.
(D) A menor nota de Clara foi igual à maior nota de Luiz.
(E) A menor nota de Luiz e a maior nota de Clara foram na mesma matéria.
14. Sempre que Luiz está ao ar livre e está fazendo sol, Luiz usa seus óculos escuros. Sempre que
Luiz está ao ar livre e não está fazendo sol, Luiz carrega seus óculos escuros no bolso. Algu-
mas vezes está fazendo sol quando Luiz não está ao ar livre.
Se as afirmações acima são verdadeiras, e Luiz não está usando seus óculos escuros, qual das
afirmações abaixo deve também ser verdadeira:
(A) Luiz está carregando seus óculos escuros no bolso.
(B) Luiz não está ao ar livre.
(C) Luiz não está ao ar livre e não está fazendo sol.
(D) Luiz não está ao ar livre e/ou não está fazendo sol.
(E) Luiz está ao ar livre e/ou não está fazendo sol.

```

Figura 5.7: Recorte de texto extraído com a ferramenta pdftotext e a *flag -raw*

A segunda ferramenta testada, *pdfminer*⁷, também possibilita configurações na execução, por meio de parâmetros variáveis. Alguns deles, relevantes para detecção e extração de texto, são os seguintes:

- **char_margin**: dois pedaços de texto cuja distância é menor que a variável *char_margin* são considerados contínuos, e portanto são agrupados em um só pedaço
- **line_margin**: duas linhas cuja distância é menor que a variável *line_margin* são agrupadas como uma "caixa de texto"(área retangular que contém porções de texto
- **word_margin**: relativo à distinção de diferentes palavras
- **boxes_flow**: especifica o quanto as posições horizontal e vertical do texto importam para determinar a ordem do texto. O valor deve estar entre -1.0 (apenas a posição horizontal importa) e +1.0 (apenas a posição horizontal importa). O valor padrão é 0.5.

Foram realizados testes com mudanças nesses parâmetros, mas logo observou-se que os valores padrão dos parâmetros produziram resultados melhores. É possível concluir pelo recorte em 5.8 o resultado negativo do uso da ferramenta, mesmo sobre um trecho relativamente "simples" de se extrair. Observa-se que os números das questões precedem ambas. Esse e outros problemas ocorreram frequentemente no uso da ferramenta, causando extrações bagunçadas.

⁷ <http://www.unixuser.org/euske/python/pdfminer/index.html>

Clara e Luiz

13.

14.

Clara e Luiz estudam na mesma escola e cursaram as mesmas seis matérias durante o ano, mas apenas em uma das matérias – História do Brasil – tiraram a mesma nota. Em cada uma das matérias as notas variaram de 60 a 100.

Qual das seguintes afirmações permite que se deduza que a média das notas que Clara tirou nas outras cinco matérias foi maior que a média das notas de Luiz nessas mesmas cinco matérias?

(A) A menor nota de Clara foi História do Brasil, mas a menor nota de Luiz foi Matemática.
 (B) A maior nota de Luiz foi maior do que a maior nota de Clara.
 (C) Clara teve notas maiores do que Luiz em três matérias.
 (D) A menor nota de Clara foi igual à maior nota de Luiz.
 (E) A menor nota de Luiz e a maior nota de Clara foram na mesma matéria.

Sempre que Luiz está ao ar livre e está fazendo sol, Luiz usa seus óculos escuros. Sempre que Luiz está ao ar livre e não está fazendo sol, Luiz carrega seus óculos escuros no bolso. Algumas vezes está fazendo sol quando Luiz não está ao ar livre.

Se as afirmações acima são verdadeiras, e Luiz não está usando seus óculos escuros, qual das afirmações abaixo deve também ser verdadeira:

(A) Luiz está carregando seus óculos escuros no bolso.
 (B) Luiz não está ao ar livre.
 (C) Luiz não está ao ar livre e não está fazendo sol.
 (D) Luiz não está ao ar livre e/ou não está fazendo sol.
 (E) Luiz está ao ar livre e/ou não está fazendo sol.

Figura 5.8: Recorte de extração de texto com a ferramenta pdfminer

Em seguida a ferramenta Tesseract (TESSERACT OCR, 2005) foi executada sobre a amostra. Diferente das anteriores, essa ferramenta utiliza *Optical Character Recognition* (OCR), tecnologia que reconhece caracteres a partir de imagens. Nos outros casos, a extração foi realizada com base no texto "embutido" nos documentos PDF. Com o Tesseract, foi necessário transformar os arquivos PDF para imagens no formato .tif, e então o texto foi reconhecido pelo programa.

Os resultados nem sempre foram bons, como exemplificado na figura 5.9 - nesse caso, os valores das alternativas aparecem separados de suas letras.

Para abranger o conhecimento e as limitações da ferramenta, ela foi executada sobre recortes de questões com diferentes estruturas textuais. O recorte visto na figura 5.10, que contém as alternativas separadas em duas colunas, foi testado de duas maneiras - na primeira, o PDF original da prova foi convertido para imagem em 300dpi (*dots per inch*), ou pontos por polegada. Na segunda, foi convertido em 900dpi. O objetivo foi verificar o quanto diferentes resoluções influenciam na extração.

Clara e Luiz

13. Clara e Luiz estudam na mesma escola e cursaram as mesmas seis matérias durante o ano, mas apenas em uma das matérias 7 História do Brasil 7 tiraram a mesma nota. Em cada uma das matérias as notas variaram de 60 a 100.

Qual das seguintes afirmações permite que se deduza que a média das notas que Clara tirou nas outras cinco matérias foi maior que a média das notas de Luiz nessas mesmas cinco matérias?

- (A)
- (B)
- (C)
- (D)
- (E)

A menor nota de Clara foi História do Brasil, mas a menor nota de Luiz foi Matemática. A maior nota de Luiz foi maior do que a maior nota de Clara.

Clara teve notas maiores do que Luiz em três matérias.

A menor nota de Clara foi igual à maior nota de Luiz.

A menor nota de Luiz e a maior nota de Clara foram na mesma matéria.

14. Sempre que Luiz esta ao ar livre e esta fazendo sol, Luiz usa seus óculos escuros. Sempre que Luiz esta ao ar livre e não esta fazendo sol, Luiz carrega seus óculos escuros no bolso. Algumas vezes esta fazendo sol quando Luiz não esta ao ar livre.

Se as afirmações acima são verdadeiras, e Luiz não esta usando seus óculos escuros, qual das afirmações abaixo deve também ser verdadeira:

- (A)
- (B)
- (C)
- (D)
- (E)

Figura 5.9: Recorte de extração de texto com a ferramenta Tesseract

Questão 19. Os dois testes NÃO conseguirão distinguir duas amostras contendo:

Amostra 1

- (A) Chicungunha, Dengue e Zika
- (B) Chicungunha e Dengue, mas não Zika
- (C) Chicungunha e Zika, mas não Dengue
- (D) Chicungunha, mas não Dengue nem Zika
- (E) Dengue, mas não Chicungunha nem Zika

Amostra 2

Chicungunha e Dengue, mas não Zika
 Dengue e Zika, mas não Chicungunha
 Dengue e Zika, mas não Chicungunha
 nem Dengue, nem Chicungunha e nem Zika
 nem Dengue, nem Chicungunha e nem Zika

Figura 5.10: Recorte de questão

Questão 19. Os dois testes NÃO conseguirão distinguir duas amostras contendo:

Amostra 1

Chicungunha, Dengue e Zika
 Chicungunha e Dengue, mas não Zika
 Chicungunha e Zika, mas não Dengue
 Chicungunha, mas não Dengue nem Zika
 Dengue, mas não Chicungunha nem Zika

Amostra 2

Chicungunha e Dengue, mas não Zika
 Dengue e Zika, mas não Chicungunha
 Dengue e Zika, mas não Chicungunha
 nem Dengue, nem Chicungunha e nem Zika
 nem Dengue, nem Chicungunha e nem Zika

Figura 5.11: Extração de texto sobre imagem com 300dpi

Questão 19. Os dois testes NÃO conseguirão distinguir duas amostras contendo:

Amostra 1 Amostra 2
 (A) Chicungunha, Dengue e Zika Chicungunha e Dengue, mas não Zika
 (B) Chicungunha e Dengue, mas não Zika Dengue e Zika, mas não Chicungunha
 (C) Chicungunha e Zika, mas não Dengue Dengue e Zika, mas não Chicungunha
 (D) Chicungunha, mas não Dengue nern Zika nern Dengue, nern Chicungunha e nem Zika
 (E) Dengue, mas não Chicungunha nern Zika nern Dengue, nern Chicungunha e nem Zika

Figura 5.12: Extração de texto sobre imagem com 900dpi

A diferença de resolução influenciou a extração, conforme as figuras 5.11 e 5.12. No primeiro caso, as colunas foram separadas como se a segunda viesse após a primeira na leitura (corrompendo o conteúdo das alternativas). No segundo caso, as colunas foram dispostas como texto corrido - respeitando a ordem das informações, porém impossibilitando a separação visual das colunas. Além disso, a palavra "nem" foi reconhecida erroneamente como "nern". Essas inconsistências indicaram a necessidade de diferentes abordagens de análise e extração sobre as imagens.

Como as ferramentas se propõem apenas a extrair o texto dos documentos, nenhuma figura presente em questões constou nos resultados. A questão da figura 5.13, por exemplo, foi extraída com espaços no lugar dos símbolos. Outra particularidade é que algumas figuras contêm legenda em formato de texto. Ele foi extraído mas acabou ficando sem sentido. Essas particularidades evidenciaram a necessidade de aplicar tratamentos de imagens nas questões antes de tentar extraí-las por completo.

	pdftotext	pdfminer	Tesseract
Realiza OCR	Não	Não	Sim
Permite uso de parâmetros na execução	Sim	Sim	Sim
Extraí figuras	Não	Não	Não

Tabela 5.1: Comparação entre ferramentas de extração de texto

A avaliação de performance é um problema recorrente em análise de documentos e imagens, e apesar de existirem diferentes bases de dados para comparação de métodos, não há um consenso a respeito de sua cobertura, representatividade ou formato de *groundtruth* (informação "base" para comparação de resultados) (HÉROUX et al., 2007).

A análise dos resultados, devido à dificuldade da determinação do *groundtruth* e ao fato de que a verificação da corretude dos processos aplicados sobre as provas nem sempre é binária, se assemelha mais a uma avaliação de performance do que a um *benchmarking*. Os resultados forneceram informações sobre a capacidade e o modo de funcionamento das ferramentas.

Ainda assim, os resultados aproximados dos testes iniciais das ferramentas sobre a amostra foram quantificados e são observados na tabela 5.2. Deve ser observado que as execuções tiveram como entrada as páginas inteiras das provas. Foram feitos mais testes com a ferramenta Tesseract, em cima de recortes de questões específicas. Esses testes mostraram resultados melhores, pois não houve a mistura de linhas de colunas diferentes, por exemplo.

A medida usada para comparação das ferramentas foi o número de linhas extraídas com erro. Uma linha é considerada errada se foi extraída misturada com outra (por exemplo, quando duas linhas de colunas diferentes são extraídas como uma só), em ordem errada de leitura ou com caracteres errados (por exemplo, acentuação não detectada). Erros consequentes das figuras e tabelas presentes nas questões não foram contabilizados.

Entre as 9 provas da amostra, o número total de páginas é 72. Ao realizar as extrações, foram ignoradas as páginas que não contêm questões (ou seja, as capas e folhas de respostas). Portanto, foram analisadas 56 páginas no total. O número total de linhas das questões (contando apenas os títulos das questões, enunciados e alternativas) correspondeu a 2700.

As execuções da ferramenta pdftotext sem parâmetros somaram 841 erros. Com o parâmetro `-raw`, ocorreram apenas 6 erros - essa execução foi capaz de detectar e extrair questões em páginas com 2 colunas na ordem correta de leitura. E como ela não aplica OCR sobre as provas, os caracteres quase sempre foram extraídos corretamente. A ferramenta pdfminer contabilizou 1059 erros. Essa quantia significativa é explicada pelo fato de que em certas provas, a ferramenta errou todas extrações de acentos. Finalmente, a ferramenta Tesseract retornou 623 erros.

Ferramenta	Linhas com erro	Porcentagem de erros
pdftotext	841	31.14%
pdftotext -raw	6	0.22%
pdfminer	1059	39.22%
Tesseract OCR	623	23.07%

Tabela 5.2: Resultados de extrações sobre páginas inteiras da amostra

A ferramenta pdftotext (com a *flag* `-raw`) apresentou uma taxa muito baixa de erros. Portanto, realizaram-se testes com o resto das provas disponíveis. Foram analisadas as 43 provas que haviam ficado de fora da amostra. Elas consistiram de 11669 linhas, das quais 82 foram extraídas com erro. A porcentagem de erros na extração, em relação à amostra, aumentou para 0.70%. Juntando as duas etapas dos testes (tabela 5.3), a ferramenta pdftotext com a *flag* `-raw` analisou 14369 linhas de 52 provas, extraíndo 88 linhas com erro e totalizando aproximada-

mente 0.61% de erros.

Páginas analisadas	Linhas analisadas	Linhas com erro	Porcentagem de erros
228	14369	88	0.61%

Tabela 5.3: Aplicação da pdftotext -raw sobre todas provas do *dataset*

Essa opção se mostrou bastante eficiente para extrair o texto das questões. Porém, o conteúdo presente em figuras e tabelas não constou corretamente nos resultados.

O formato dos resultados não coopera completamente para a automatização da identificação e extração de cada questão. Com base nos testes realizados até então, concluiu-se que é essencial aplicar tratamentos nas páginas das provas, para depois capturar o texto em si. Esses tratamentos devem atentar à estrutura e forma em que o conteúdo é disposto nas provas. Portanto, deve-se trabalhar com a segmentação das páginas.

Questão 1. Até hoje os índios Turiacu mantêm vivas suas tradições. Todo final de mês, numa cerimônia presidida pelo pajé, eles enviam uma mensagem com sinais de fumaça para informar às tribos vizinhas o número de crianças nascidas naquele mês, como uma indicação de que eles continuarão fortes no futuro.

A mensagem é composta sempre de cinco símbolos. O primeiro símbolo é sempre um sinal de fumaça escura que serve para indicar o início da mensagem (●). O sinal horizontal (⇔) sempre representa o valor 0, em qualquer posição que apareça. O sinal vertical (⋮) representa o valor 1 se aparece na segunda posição (após o sinal de fumaça escura), 2 se aparece na terceira posição, 4 se aparece na quarta posição, e 8 se aparece na quinta posição. O número de crianças indicado na mensagem é a soma dos valores dos sinais. Assim, por exemplo, a mensagem (● ⋮ ⇔ ⋮ ⇔) representa $1 + 0 + 4 + 0 = 5$. Já a mensagem (● ⇔ ⇔ ⇔ ⋮) representa $0 + 0 + 0 + 8 = 8$. Qual o maior valor possível de ser enviado em uma mensagem?

- (A) 8
- (B) 9
- (C) 15
- (D) 16
- (E) 31



Figura 5.13: Questão da prova de 2015 (1ª fase, 2º nível)

5.3 SEGMENTAÇÃO DE PÁGINAS

A análise de *layout* é uma etapa muito importante na análise de documentos. Erros cometidos nesse estágio propagam nas etapas subsequentes de OCR e podem impactar negativamente no sucesso da aplicação como um todo (ANTONACOPOULOS; KARATZAS; BRIDSON, 2006).

Durante o processo de pesquisa de ferramentas para auxiliar no trabalho, foi encontrada uma opção interessante no que se refere à análise da estrutura dos documentos. A ferramenta OCRopus⁸ corresponde a um conjunto de programas de manipulação de imagens, executáveis

⁸ <https://github.com/tmbdev/ocropy>

por linha de comando. Suas funções incluem reconhecimento de texto, transformação de imagem, e análise de *layout*.

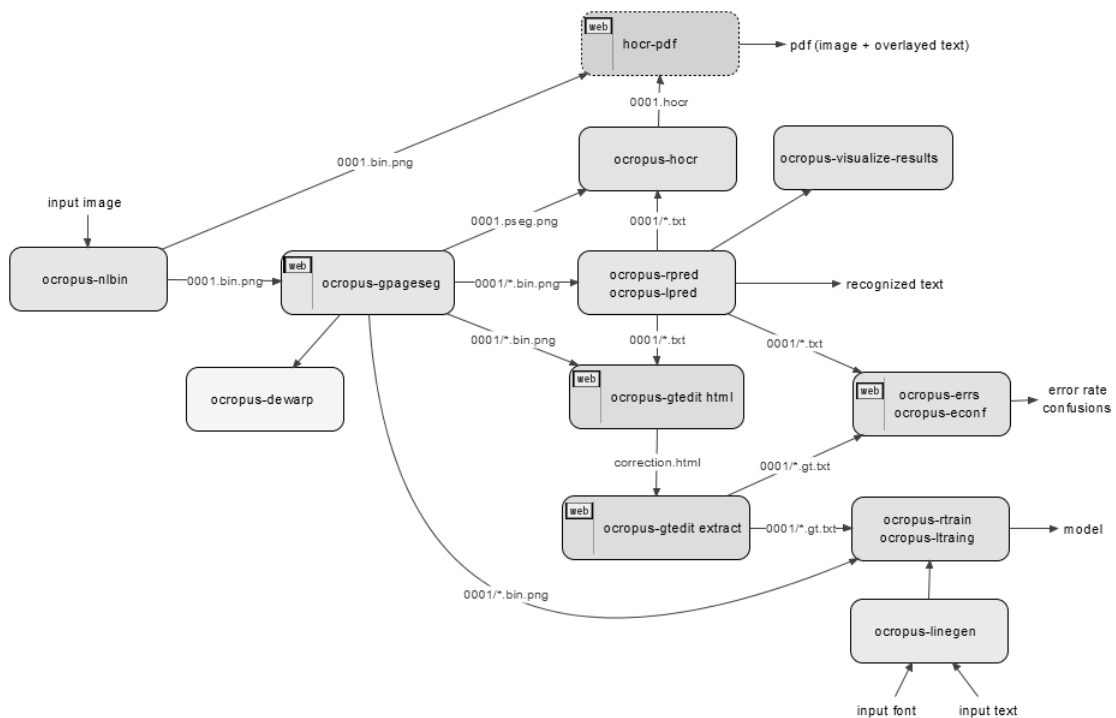


Figura 5.14: *Workflow* da biblioteca OCRopus

Em 5.14 observa-se as possibilidades de uso da biblioteca. Alguns comandos devem ser realizados em uma determinada ordem para funcionarem corretamente. Por exemplo, pode-se fazer a binarização de uma imagem, e sobre o resultado a segmentação, seguida do reconhecimento de caracteres.

No trabalho, foi priorizada a função de segmentação, que extrai individualmente as linhas de texto da página com o comando `ocropus-gpageseg`. Para cada execução sobre uma imagem, cria-se um diretório contendo imagens de cada linha reconhecida.

A questão da figura 5.15 foi utilizada como entrada em um teste da ferramenta. Como resultado, novas imagens foram geradas, cada uma representando uma linha de texto detectada. Até o fim do enunciado, os resultados foram corretos, porém, apenas a alternativa "E" foi detectada. Da alternativa "A" até a "D", somente a letra da alternativa foi retornada - a ferramenta não detectou os números 0, 1, 2 e 3.

Questão 2. Camila muda constantemente suas senhas. Suas senhas são sempre números inteiros maiores do que zero, e ela definiu as seguintes regras para suas senhas:

- a senha deve ter sempre o menor valor possível
- a senha deve ser maior do que qualquer outra senha já utilizada
- a senha não pode ser divisível por nenhuma senha já utilizada

A primeira senha de Camila tem o valor 5. Quantos números pares podem ser usados como senha por Camila, durante toda a sua vida?

- (A) 0
- (B) 1
- (C) 2
- (D) 3
- (E) infinitos

Figura 5.15: Imagem de entrada para o OCRopus

A ferramenta ainda possibilita o uso do parâmetro `--maxcolseps`", em que o usuário especifica o número máximo de separadores de colunas (espaços em branco no meio da página, por exemplo) que dividem o texto. Por padrão o valor é 3. Os resultados da execução com esse parâmetro valendo 0 e 1 sobre o recorte da figura 5.16 mostram a influência dele. Com a opção `--maxcolseps 1`", as alternativas ficaram divididas em diferentes imagens para cada letra e valor da alternativa. E a ordem em que essas imagens foram geradas e dispostas como resultado não foi ideal, pois mostrou primeiro todas as letras das alternativas e depois todos valores das alternativas, dificultando o iminente tratamento delas. Já com o parâmetro tendo valor 0, o programa não considerou o espaço em branco entre letra da alternativa e texto como divisor de colunas. O resultado, portanto, foi melhor, unindo toda a linha em uma só imagem. Porém, o programa também mostrou resultados equivocados, como se observa na figura 5.17. Mesmo variando o parâmetro, duas linhas de colunas diferentes do texto são detectadas como uma só linha. Execuções sobre outras questões produziram muitas extrações com erros, desmotivando o uso da ferramenta.

Questão 16. Qual das seguintes opções é uma lista completa, correta e na ordem de substâncias adicionadas nos produtos A e B?

- (A) Produto A: Q, P, L, O, J
Produto B: M, N, R, Q
- (B) Produto A: M, R, L, O
Produto B: J, N, K, P, Q
- (C) Produto A: M, J, L, K, P
Produto B: Q, N, R, O
- (D) Produto A: P, J, L, K, Q
Produto B: M, N, R, O
- (E) Produto A: M, Q, L, O
Produto B: J, R, N, K, P

Figura 5.16: Imagem de entrada para o OCRopus

Dez crianças do bairro: C, D, E, F, G, H, J, L, M e N. **Questão 12.** Qual das seguintes crianças deve ser ne-

Figura 5.17: Erro na segmentação de texto

5.4 LAREX

O próximo passo do trabalho foi explorar a ferramenta LAREX (REUL; SPRINGMANN; PUPPE, 2017). O nome da ferramenta é um acrônimo para *Layout Analysis and Region EXtraction*. Seu objetivo é ajudar usuários na segmentação e classificação de regiões em imagens. O procedimento de segmentação usa uma abordagem de componentes conectados, destacando diferentes blocos de texto/imagens de uma página, e em seguida permite que o usuário corrija manualmente os eventuais erros de segmentação. A ferramenta deve ser executada em um navegador web. Para este trabalho, as imagens carregadas na ferramenta foram conversões das páginas dos arquivos PDF originais para o formato .tif, em 300dpi.

A primeira etapa realizada pela ferramenta corresponde ao pré-processamento da página - ela é binarizada e em seguida, caso o usuário tenha especificado uma região de interesse para a segmentação, é selecionada apenas essa região para ser processada nas etapas seguintes.

Na etapa seguinte são detectadas as imagens (figuras) presentes na página. É presumido que as figuras são compactas ou contêm bordas à sua volta. É realizada uma operação de dilatação, em que os pixels pretos das figuras são expandidos, formando componentes conectados. Os componentes conectados que forem maiores que um certo *threshold* de área de imagem são considerados como uma imagem.

Para lidar com o texto presente na página, é utilizada uma abordagem *bottom-up*. Outra operação de dilatação é aplicada, com valores maiores. Os pixels do texto se misturam, for-

mando blocos. As regiões detectadas são classificadas em um tipo semântico, que pode ser um parágrafo, marginália ou número de página. Para isso, são levadas em conta algumas condições, como o tamanho da região, número de ocorrências do tipo de região na página, e localização da região na página. Por exemplo, para uma região ser considerada como referente ao número da página, ela deve estar posicionada próxima às bordas da página. Parâmetros como o tamanho mínimo que uma região precisa ter para ser considerada um parágrafo, por exemplo, são customizáveis pelo usuário.

A ferramenta realiza a segmentação automática e a exhibe para o usuário como na figura 5.18, possibilitando edições manuais, como por exemplo, separar uma região em duas. As regiões são representadas por polígonos que envolvem o texto. Como esses polígonos contêm muitos pontos, a interação do usuário ao editá-los (aumentando, diminuindo ou mudando o formato) acaba sendo trabalhosa.

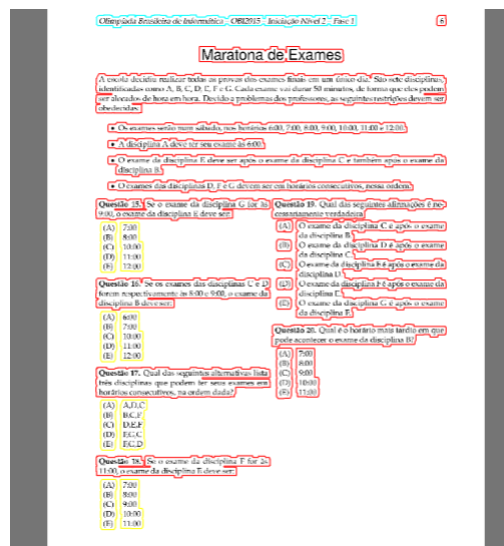


Figura 5.18: Recorte da interface da ferramenta LAREX após segmentação automática

Quando o usuário estiver satisfeito com a segmentação, o próximo passo é salvar o resultado em um arquivo XML, no formato PageXML (PLETSCHACHER; ANTONACOPOULOS, 2010). Esse formato, propício para análise de imagens, guarda informações de páginas, incluindo estrutura do *layout* e seu conteúdo. Ele é utilizado em avaliações de métodos submetidos em competições do ICDAR, por exemplo. As tags mais relevantes dos arquivos gerados pela LAREX são *TextRegion* (que possui o atributo "tipo", referindo-se ao tipo semântico da região) e *ImageRegion*. Ambas possuem como filha a tag *Coords*, que guarda todos os pontos do polígono que envolve a região.

5.5 SOLUÇÃO DESENVOLVIDA

Para possibilitar a extração das questões, foi desenvolvido um programa com a linguagem de programação Python. Utilizou-se a biblioteca OpenCV um *wrapper* do Tesseract para a linguagem. O programa manipula os resultados oriundos da LAREX (arquivos no formato PageXML) e apresenta as provas, através de uma interface gráfica, para o usuário completar as extrações.

O objetivo é que o usuário não precise processar as provas manualmente na LAREX, mas o programa desenvolvido também requer a interação do usuário para garantir que as questões sejam extraídas corretamente. Caso a execução da LAREX tenha detectado as questões erroneamente, o usuário pode corrigir os erros. Porém, essa interação é mais simples do que a proposta pelo LAREX. Os blocos (regiões de texto ou imagens) são tratados como retângulos em vez de polígonos, facilitando as eventuais manipulações. Além disso, antes de apresentar as páginas a serem editadas (opcionalmente) pelo usuário, o programa processa os blocos descritos no arquivo XML e realiza algumas operações, como ignorar o conteúdo do cabeçalho das páginas. Mostrou-se frequente, no uso da LAREX, a detecção das letras de alternativas e seus valores em blocos distintos. O programa se encarrega de juntar esses blocos automaticamente, com base no tamanho desses blocos e posição dentro da página, para que o usuário tenha menos trabalho ao editar.

A execução pede como argumento o caminho para um diretório contendo os arquivos de imagem da prova e seus respectivos arquivos descritivos em XML. Ambos arquivos devem possuir o mesmo nome. Cada imagem (página da prova) é processada com o arquivo XML vinculado. A representação das regiões de texto é transformada de polígonos para retângulos, e nos casos em que as letras de alternativas foram detectadas separadamente na execução da LAREX, elas são unidas antes da página com os blocos destacados ser apresentada para o usuário realizar as correções.

Os tipos semânticos de texto classificados pela LAREX (parágrafo, marginalia e número de página) foram ignorados pelo programa. Essas classificações não foram úteis no contexto das provas da OBI. A classificação semântica das regiões nas provas deveria ter outros valores para possibilitar o armazenamento em um banco de dados específico, por exemplo. Deve-se levar em consideração o fato de que podem existir diversas questões relativas ao enunciado de um único problema.

O usuário pode marcar os blocos com tipos semânticos. Esses tipos podem ser:

1. título do problema
2. enunciado do problema
3. regras do problema
4. enunciado da questão
5. alternativas
6. figura

Para realizar a marcação, basta selecionar o bloco com o botão direito do mouse e clicar no número referente ao tipo semântico.

O usuário conta ainda com as funcionalidades de criar blocos, unir blocos distintos, deletar blocos e dividir um bloco em dois. As ações são realizadas clicando com o botão direito do mouse. Para unir blocos, basta clicar em um bloco e arrastar o mouse até o outro. Também é possível reverter a página para o estado original. O usuário escolhe o formato em que as questões são extraídas, podendo ser como arquivos de imagem (formato .tif) ou texto (formato .txt). A saída em formato de imagem é útil principalmente quando uma questão contém figuras, sendo essa a única maneira de obter a questão sem perder informações essenciais. Para extrair as questões como texto, o programa chama um comando do Tesseract. Após um dos tipos de extração ser realizado, o programa abre automaticamente a próxima página para o usuário continuar o processo.

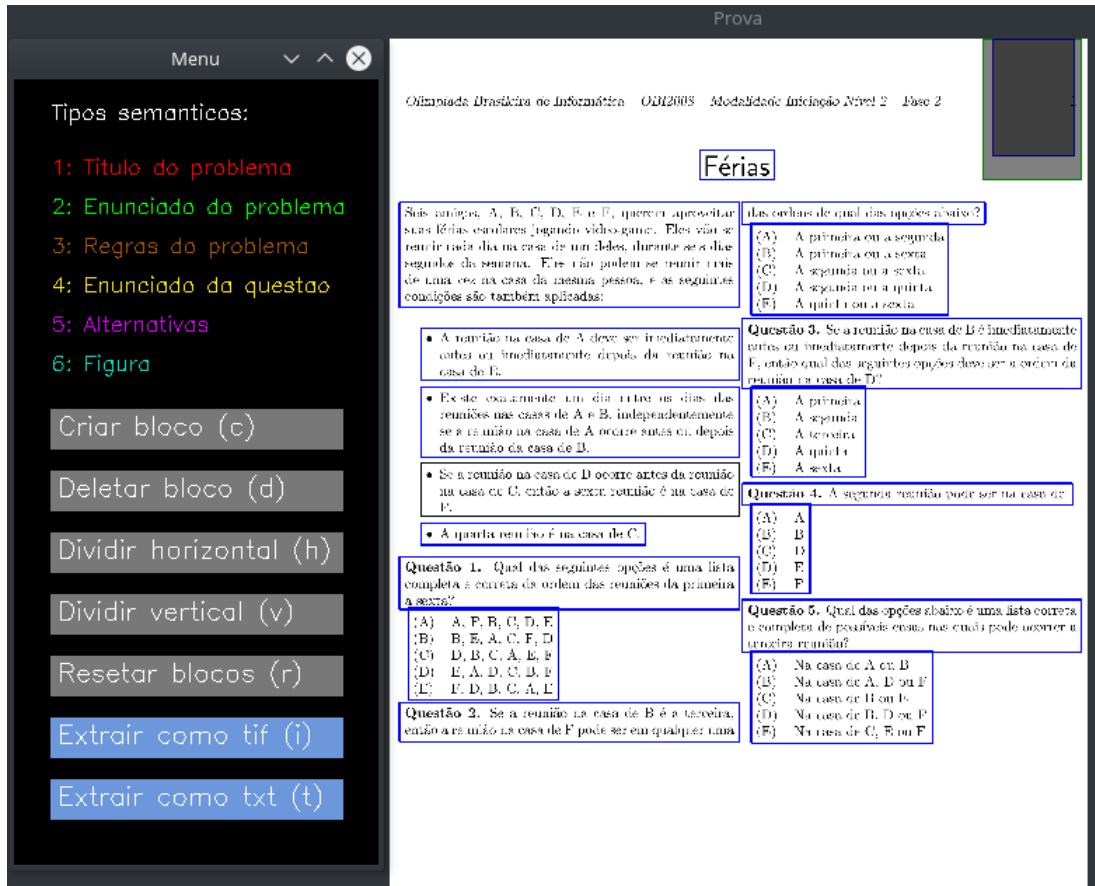


Figura 5.19: Interface do programa

A interface gráfica do programa (figura 5.19) foi criada apenas com elementos do OpenCV e apresenta duas janelas. A janela principal (menu) mostra botões que respondem às funcionalidades descritas anteriormente. Também mostra os possíveis tipos semânticos e os números relacionados, que quando pressionados no teclado, marcam os blocos. Outra janela apresenta a página a ser editada e ter as questões extraídas. Com o fim de permitir um uso prático da ferramenta, ainda é possível realizar as ações do programa utilizando atalhos do teclado, que são especificados na janela principal.

6 RESULTADOS E DISCUSSÕES

Há várias dificuldades relacionadas à avaliação de performance em quase todas áreas de pesquisa de visão computacional (S.SHIYAMALA, 2017). Não foi identificado um método específico de avaliação para medir a performance e os resultados da solução desenvolvida, especialmente considerando a etapa de interação do usuário. Além disso, as questões extraídas não necessariamente têm um formato que possa ser considerado certo ou errado. Por exemplo, pode-se extrair uma questão contendo a pergunta e as alternativas no mesmo bloco, ou em blocos separados - ambas maneiras podem ser válidas, dependendo da intenção do usuário após a extração. Uma comparação de cunho quantitativo da ferramenta com outras abordagens existentes seria confusa.

Ainda assim, é possível analisar alguns resultados do uso do programa. A figura 6.1 mostra uma página de prova após o programa processar os resultados da LAREX e realizar os ajustes padrões, antes do usuário editar e extrair. Nota-se que os blocos detectados correspondem corretamente ao título, partes do enunciado, perguntas e alternativas.

Olimpíada Brasileira de Informática - OBI2008 - Modalidade Inicial - Nível 2 - Fase 2

Máquina de Salgadinhos

Numa loja existe uma máquina que vende salgadinhos. Para comprar um salgadinho você escolhe uma moeda e aperta o botão correspondente ao sabor desejado; a máquina então coloca o salgadinho que você escolheu num compartimento para que você possa pegá-lo. Nessa máquina existem seis botões na vertical, um para cada um dos seguintes sabores: Pizza, Presunto, Requeijão, Churrasco, Cebola e Queijo. O problema é que os salgadinhos ficam colocados incrementalmente e não se sabe mais qual botão corresponde a que sabor. São conhecidas as seguintes fatos:

- Cada botão só pode estar associado a somente um sabor.
- Os botões estão dispostos na vertical, um em cima do outro.
- Um e somente um dos botões está quebrado; esse seja, corresponde ao sabor do salgadinho com o rótulo do botão.
- O botão que dá o sabor de Cebola está mais alto que o botão que dá o sabor de Queijo mas baixo que o botão que dá o sabor de Requeijão.
- O botão que dá o sabor de Churrasco está imediatamente acima ou imediatamente abaixo do botão que dá o sabor de Requeijão.

Questão 21. Qual das seguintes opções é uma lista completa e correta de sabores associados aos botões, de mais alto para mais baixo?

(A) Cebola, Presunto, Churrasco, Requeijão, Queijo, Pizza
 (B) Pizza, Presunto, Requeijão, Churrasco, Cebola, Queijo
 (C) Presunto, Churrasco, Requeijão, Pizza, Cebola, Queijo
 (D) Requeijão, Cebola, Churrasco, Queijo, Presunto, Pizza
 (E) Presunto, Pizza, Requeijão, Churrasco, Cebola, Queijo

Questão 22. Qual é o máximo número de sabores diferentes que podem ser dados ao pressionar o botão rotulado com Cebola?

(A) 1
 (B) 2
 (C) 3
 (D) 4
 (E) 5

Questão 23. Se o sabor de Cebola é dado pelo botão rotulado com Requeijão, qual dos seguintes botões está dando exatamente o sabor rotulado?

(A) Presunto
 (B) Requeijão
 (C) Churrasco
 (D) Pizza
 (E) Queijo

Questão 24. Se o sabor de Presunto é dado pelo botão que está mais alto que o botão que dá o sabor Churrasco, qual das seguintes opções é falsa?

(A) Presunto vem do botão rotulado Presunto.
 (B) Cebola vem do botão rotulado Pizza.
 (C) Pizza vem do botão rotulado Presunto.
 (D) Requeijão vem do botão rotulado Presunto.
 (E) Queijo vem do botão rotulado Pizza.

Questão 25. Se o sabor de Pizza não está no botão rotulado Pizza, qual das seguintes opções pode ser verdadeira?

(A) Requeijão é dado pelo botão rotulado Queijo.
 (B) Cebola é dado pelo botão rotulado Requeijão.
 (C) Presunto é dado pelo botão rotulado Pizza.
 (D) Presunto é dado pelo botão rotulado Requeijão.
 (E) Cebola é dado pelo botão rotulado Presunto.

Figura 6.1: Segmentação de prova de 2008 apresentada para o usuário

Já o casos mostrados nas figuras 6.2 e 6.3 não foram ideais. Na primeira, o título foi destacado em 2 partes, dois blocos se intersectam erroneamente e a última questão tem os valores das alternativas separados, além do número da página ser incluído em um bloco. E na segunda, as alternativas de uma questão estão separadas, além de parte de outra questão não estar inclusa em nenhum bloco.

Maionese

Um grande número de convidados ficou intoxicado após a festa de casamento de João e Joana. A suspeita logo recaiu sobre a salada de maionese servida na comemoração. Para tratar com mais eficiência os convidados afetados, amostras de maionese devem ser testadas para determinar a presença das toxinas R, S e T. Serão utilizados dois testes, X e Z.

Sabe-se que:

- uma amostra mantém a cor adquirida em um teste a menos que outro teste altere a cor da amostra;
- o teste X faz a amostra adquirir cor verde se esta contém R ou S, ou ambas, e faz a amostra adquirir cor laranja se esta não contém nem R nem S;
- o teste Z faz a amostra adquirir cor rosa se esta contém T; se a amostra não contém T, permanece com a mesma cor que tinha antes do teste Z.

5. Uma amostra que contenha R e S mas não contenha T adquirirá que sequência de cores, a primeira após o teste X ser aplicado, e a segunda após o teste Z ser aplicado?

(A) Verde, verde
 (B) Verde, rosa
 (C) Laranja, amarelo
 (D) Laranja, laranja
 (E) Laranja, rosa

6. Uma amostra que permaneça amarela quando submetida ao teste Z e que adquira cor verde quando submetida ao teste X pode ser uma amostra contendo

(A) R, S e T.
 (B) S e T, mas não R.
 (C) T, mas não R nem S.
 (D) S, mas não R nem T.
 (E) nem R, nem S, nem T.

7. Os dois testes NÃO conseguirão distinguir duas amostras contendo:

Amostra 1	Amostra 2
(A) R, S e T	R e S, mas não T
(B) R e S, mas não T	S e T, mas não R
(C) R e T, mas não S	S e T, mas não R
(D) R, mas não S nem T	Nem S, nem R e nem T
(E) S, mas não R nem T	Nem S, nem R e nem T

Figura 6.2: Segmentação de prova de 2004 apresentada para o usuário

Questão 1. Qual das seguintes listas seguintes é uma atribuição válida de funcionários aos projetos?

	Projeto 1	Projeto 2	Projeto 3
(A)	Denise e Felipe	Felipe e Bia	Alfonso Clara
(B)	Denise e Bia	Licardo e Felipe	Alfonso Clara
(C)	Eduardo e Clara	Denise e Bia	Alfonso Felipe
(D)	Felipe e Clara	Alfonso e Bia	Denise e Felipe
(E)	Alfonso e Felipe	Denise e Eduardo	Bia e Clara

Questão 2. Qual das seguintes é uma lista completa e correta dos funcionários que o engenheiro chefe pode escolher para trabalhar no mesmo projeto que Clara?

(A) Bia
(B) Denise
(C) Eduardo
(D) Bia e Felipe
(E) Denise e Eduardo

Questão 3. Se Eduardo trabalhar no Projeto 2, qual das seguintes afirmações é necessariamente verdadeira?

(A) Bia trabalhará no Projeto 1
(B) Clara trabalhará no Projeto 2

Questão 4. O engenheiro chefe NÃO PODE fazer as seguintes atribuições:

(A) Alfonso trabalhar no Projeto 1 e Bia trabalhar no Projeto 2
(B) Alfonso trabalhar no Projeto 2 e Clara trabalhar no Projeto 3
(C) Denise trabalhar no Projeto 1 e Eduardo trabalhar no Projeto 3
(D) Felipe trabalhar no Projeto 1 e Clara trabalhar no Projeto 3
(E) Felipe trabalhar no Projeto 1 e Denise trabalhar no Projeto 2

Questão 5. Qual das seguintes afirmações é verdadeira?

Figura 6.3: Recorte de segmentação de prova de 2005 apresentada para o usuário

7 CONCLUSÃO

Tendo como objetivo automatizar a extração de questões de provas da Olimpíada Brasileira de Informática, o trabalho se propôs a pesquisar e estudar ferramentas que ajudassem na análise e extração de texto das provas da competição. Entende-se que a apresentação das questões de forma isolada pode ser proveitosa para fins de estudo, por exemplo. Foram avaliados prós e contras das diferentes abordagens, descobrindo-se algumas limitações e desafios a serem solucionados.

Uma amostra de provas foi obtida para que os testes das ferramentas fossem realizados. Em alguns casos houve um número grande de erros de extração de conteúdo das provas. Um problema para a extração das questões é o fato de que os *layouts* das provas não têm sempre o mesmo formato. A disposição de texto em duas colunas, por exemplo, foi um empecilho para algumas ferramentas. Em outros casos os resultados foram melhores, extraindo corretamente os enunciados e alternativas das questões. As execuções da ferramenta pdftotext com o parâmetro -raw sobre as 52 provas do dataset resultaram em apenas 0.61% de erros. Porém, essa ferramenta não extraiu figuras e outras informações gráficas das questões.

Considerando as informações obtidas, concluiu-se que não há uma ferramenta que atenda perfeitamente ao objetivo desejado. A maneira de extrair as questões individualmente foi desenvolver um programa que processa informações resultantes da ferramenta LAREX. Essa ferramenta atua sobre imagens (as páginas das provas), realizando uma análise de *layout* das mesmas e apresenta o resultado (regiões de texto e imagens detectadas) no formato PageXML.

O programa desenvolvido, que analisa as imagens das páginas das provas e os respectivos arquivos descritivos em XML, processa essas informações para apresentar ao usuário os blocos de conteúdo detectados. A interface gráfica do programa é simples, feita com funções da biblioteca OpenCV. O usuário então interage com o programa, corrigindo os eventuais erros, e tem a opção de obter os resultados como imagens ou em formato de texto.

Unindo o processamento automático e a interação do usuário com o programa, tornou-se possível extrair as questões de maneira prática, viabilizando o uso dos resultados para quaisquer fins desejados.

REFERÊNCIAS

- ANTONACOPOULOS, A.; KARATZAS, D.; BRIDSON, D. Ground truth for layout analysis performance evaluation. In: INTERNATIONAL WORKSHOP ON DOCUMENT ANALYSIS SYSTEMS. **Anais...** [S.l.: s.n.], 2006. p.302–311.
- CATTONI, R. et al. Geometric layout analysis techniques for document image understanding: a review. **ITC-irst Technical Report**, [S.l.], v.9703, n.09, 1998.
- CHAPELLE, O.; HAFFNER, P.; VAPNIK, V. N. Support vector machines for histogram-based image classification. **IEEE transactions on Neural Networks**, [S.l.], v.10, n.5, p.1055–1064, 1999.
- DAUGMAN, J. G. Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression. **IEEE Transactions on acoustics, speech, and signal processing**, [S.l.], v.36, n.7, p.1169–1179, 1988.
- DESARATHY, B. **Nearest Neighbor Pattern Classification Techniques**. [S.l.]: Los Alamitos: Institute of Electrical and Electronics Engineers (IEEE) Computer Society Press, 1991.
- GATOS, B.; STAMATOPOULOS, N.; LOULLOUDIS, G. ICDAR2009 handwriting segmentation contest. **International Journal on Document Analysis and Recognition (IJ DAR)**, [S.l.], v.14, n.1, p.25–33, 2011.
- HÉROUX, P. et al. Automatic ground-truth generation for document image analysis and understanding. In: DOCUMENT ANALYSIS AND RECOGNITION, 2007. ICDAR 2007. NINTH INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2007. v.1, p.476–480.
- INTERNATIONAL Conference on Document Analysis and Recognition. 1991.
- MARTINS, W. S. **Jogos de Lógica - divirta-se e prepare-se para a Olimpíada Brasileira de Informática**. [S.l.]: Editora Vieira, 2011.
- NAGY, G.; SETH, S.; VISWANATHAN, M. A prototype document image analysis system for technical journals. **Computer**, [S.l.], v.25, n.7, p.10–22, 1992.
- NAMBOODIRI, A. M.; JAIN, A. K. Document structure and layout analysis. In: **Digital Document Processing**. [S.l.]: Springer, 2007. p.29–48.

O’GORMAN, L. The document spectrum for page layout analysis. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, [S.l.], v.15, n.11, p.1162–1173, 1993.

PLETSCHACHER, S.; ANTONACOPOULOS, A. The PAGE (page analysis and ground-truth elements) format framework. In: PATTERN RECOGNITION (ICPR), 2010 20TH INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2010. p.257–260.

REUL, C.; SPRINGMANN, U.; PUPPE, F. LAREX: a semi-automatic open-source tool for layout analysis and region extraction on early printed books. In: INTERNATIONAL CONFERENCE ON DIGITAL ACCESS TO TEXTUAL CULTURAL HERITAGE, 2. **Proceedings...** [S.l.: s.n.], 2017. p.137–142.

S.SHIYAMALA, D. S. Review on Text Detection, Extraction and Recognition from Images and Videos. **International Journal of Innovative Research in Computer and Communication Engineering**, [S.l.], v.5, n.7, 2017.

TESSERACT OCR. Acessado: Agosto 2018, <https://github.com/tesseract-ocr/>.

ZIRARI, F. et al. A document image segmentation system using analysis of connected components. In: DOCUMENT ANALYSIS AND RECOGNITION (ICDAR), 2013 12TH INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2013. p.753–757.

APÊNDICES

APÊNDICE A – Exemplos do uso de ferramentas de extração de texto sobre provas da OBI

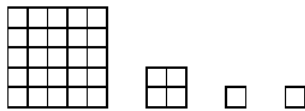
O objetivo deste apêndice é mostrar algumas particularidades de questões da OBI ao longo dos anos as consequências da aplicação de ferramentas de extração de texto sem nenhum tratamento adicional.

Um problema explícito está na presença de imagens/ilustrações nas questões. Como se observa na questão da figura A.1, uma informação essencial para sua resolução não consta no objeto extraído (figura A.2).

Azulejos

São dados N azulejos de dimensões $10\text{cm} \times 10\text{cm}$. Com eles, você deve montar um conjunto de quadrados (com espessura de um azulejo) de modo a utilizar TODOS os azulejos dados. Inicialmente você deve montar o maior quadrado possível com os azulejos dados; então, com os azulejos que sobraram, você deve montar o maior quadrado possível, e assim sucessivamente.

Por exemplo, se forem dados 31 azulejos, o conjunto montado terá quatro quadrados, conforme ilustra a figura abaixo



Conjunto com quatro quadrados, montado a partir de 31 azulejos

Questão 7. Qual o número de quadrados do conjunto montado se forem dados 75 azulejos?

- (A) 2
- (B) 3
- (C) 4
- (D) 5
- (E) 6

Questão 8. Qual o número de quadrados do conjunto montado se forem dados 120 azulejos?

- (A) 2
- (B) 3
- (C) 4
- (D) 5
- (E) 6

Questão 9. Qual o número de quadrados do conjunto montado se forem dados 148 azulejos?

- (A) 2
- (B) 3
- (C) 4
- (D) 5
- (E) 6

Figura A.1: Questão da prova de 2016 (1ª fase, 1º nível)

A questão na figura A.3 também apresenta problemas relacionados a símbolos. As ferramentas pdftotext (com a *flag* -raw) e pdfminer trataram os símbolos redondos de maneiras diferentes. A primeira mostra, no lugar dos símbolos, um retângulo que representa um caractere não encontrado. A última mostra apenas o código CID (*Character Identifier*) do caractere, que é usado para lidar com fontes de linguagens orientais, como japonês ou coreano. Além disso, os valores das alternativas aparecem antes mesmo da pergunta da questão.

A questão "Florista", da prova de 2015, contém em seu enunciado números formatados de maneira circular. As extrações reconhecem os números, mas exibem problemas. Na figura

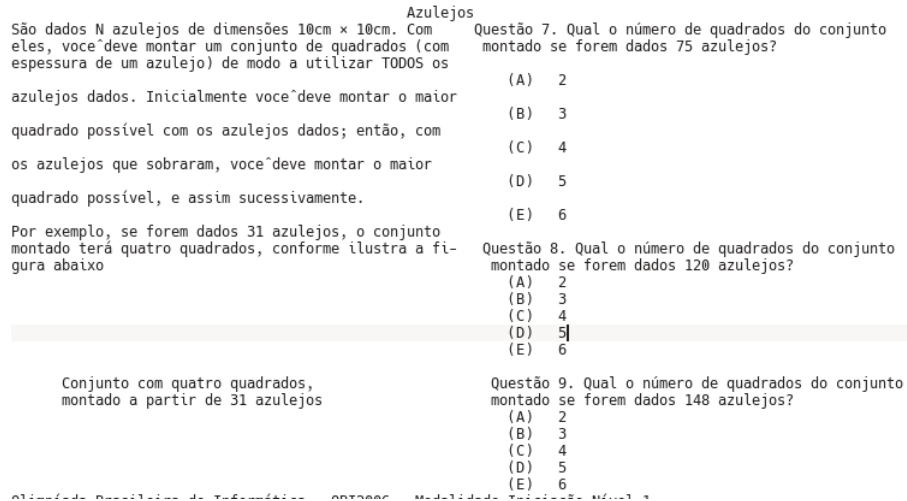


Figura A.2: Extração de texto de questão com a ferramenta pdftotext e *flag*-layout

A.8, a informação é negligenciada pois os números aparecem um abaixo do outro. Já na figura A.7, o layout circular dos números é respeitado, exceto por um pequeno desvio. Porém, partes do enunciado quase se misturam com as questões adjacentes.

Questão 8. Um robô furador pode ser programado usando os comandos gráficos \rightarrow e \circ , que realizam as seguintes operações :

- \rightarrow : move-se 1cm para a frente
- \circ : faz um furo na posição corrente

Assim, podemos programar que o robô faça dois furos a uma distância de 1cm um do outro com os comandos $\circ \rightarrow \circ$. Além disso, podemos programar repetições, utilizando números e parênteses. Por exemplo

- $4 \rightarrow$: repete quatro vezes a ação "move-se 1cm para a frente" (ou seja, o robô move-se 4cm para a frente)
- $4 \circ$: repete quatro vezes a ação "faz um furo na posição corrente" (como o robô não se move, faz um único furo na posição corrente)
- $4 (\rightarrow \rightarrow)$: repete quatro vezes a ação "move-se 1cm para a frente, move-se 1cm para a frente" (ou seja, o robô move-se 8cm para a frente)

Qual o comando para o robô fazer quatro furos em linha reta, cada furo distante 1cm do furo seguinte?

- (A) $4 (\rightarrow \rightarrow \circ)$
 (B) $4 \rightarrow 4 \circ$
 (C) $4 (\rightarrow \circ)$
 (D) $4 \circ 4 \rightarrow$
 (E) $4 \circ \rightarrow$

Figura A.3: Questão da prova de 2015 (1ª fase, 2º nível)

```

Questão 8. Um robô furador pode ser programado usando os comandos gráficos  $\rightarrow$  e  $\square$ , que realizam
as seguintes operações :
•  $\rightarrow$  : move-se 1cm para a frente
•  $\square$  : faz um furo na posição corrente
Assim, podemos programar que o robô faça dois furos a uma distância de 1cm um do outro com os
comandos  $\square \rightarrow \square$ . Além disso, podemos programar repetições, utilizando números e parênteses. Por
exemplo
•  $4 \rightarrow$  : repete quatro vezes a ação "move-se 1cm para a frente" (ou seja, o robô move-se 4cm para
a frente)
•  $4 \square$  : repete quatro vezes a ação "faz um furo na posição corrente" (como o robô não se move,
faz um único furo na posição corrente)
•  $4 (\rightarrow \rightarrow)$  : repete quatro vezes a ação "move-se 1cm para a frente, move-se 1cm para a frente"
(ou seja, o robô move-se 8cm para a frente)
Qual o comando para o robô fazer quatro furos em linha reta, cada furo distante 1cm do furo seguinte?
(A)  $4 (\rightarrow \rightarrow \square)$ 
(B)  $4 \rightarrow 4 \square$ 
(C)  $4 (\rightarrow \square)$ 
(D)  $4 \square 4 \rightarrow$ 
(E)  $4 \square \rightarrow$ 

```

Figura A.4: Extração de texto de questão com a ferramenta pdftotext e *flag* -raw

```

Questão 8. Um robô furador pode ser programado usando os comandos gráficos  $\rightarrow$  e (cid:7), que realizam
as seguintes operações :
•  $\rightarrow$  : move-se 1cm para a frente
• (cid:7) : faz um furo na posição corrente
Assim, podemos programar que o robô faça dois furos a uma distância de 1cm um do outro com os
comandos (cid:7)  $\rightarrow$  (cid:7). Além disso, podemos programar repetições, utilizando números e parênteses. Por
exemplo
•  $4 \rightarrow$  : repete quatro vezes a ação "move-se 1cm para a frente" (ou seja, o robô move-se 4cm para
a frente)
•  $4$  (cid:7) : repete quatro vezes a ação "faz um furo na posição corrente" (como o robô não se move,
faz um único furo na posição corrente)
•  $4 (\rightarrow \rightarrow)$  : repete quatro vezes a ação "move-se 1cm para a frente, move-se 1cm para a frente"
(ou seja, o robô move-se 8cm para a frente)
4 ( $\rightarrow \rightarrow$  (cid:7))
4  $\rightarrow$  4 (cid:7)
4 ( $\rightarrow$  (cid:7))
4 (cid:7) 4  $\rightarrow$ 
4 (cid:7)  $\rightarrow$ 
Qual o comando para o robô fazer quatro furos em linha reta, cada furo distante 1cm do furo seguinte?
(A)
(B)
(C)
(D)
(E)

```

Figura A.5: Extração de texto de questão com a ferramenta pdfminer

Florista

Uma florista está arranjando oito flores (A, B, C, F, G, J, K e L) em oito vasos colocados em formato de círculo, como mostrado abaixo:



Sabe-se o seguinte sobre o arranjo de flores:

- A, B e C são lírios; F e G são margaridas; J, K e L são rosas;
- apenas uma flor deve ser colocada em cada vaso;

Questão 9. Se L for colocada no vaso 8, e K for colocada em vaso vizinho ao vaso de L, qual das seguintes afirmações é necessariamente verdadeira?

- (A) A é colocada no vaso vizinho ao vaso de B
- (B) B é colocada no vaso vizinho ao vaso de G
- (C) G está no vaso diametralmente oposto ao vaso de J
- (D) J está no vaso diametralmente oposto ao vaso de A
- (E) L está no vaso diametralmente oposto ao vaso de A

Questão 10. Qual das seguintes flores está necessariamente em vaso vizinho ao vaso de A?

Figura A.6: Recorte de questão da prova de 2005 (1º nível, fase única)

Florista	
<p>Uma florista está arranjando oito flores (A, B, C, F, G, J, K e L) em oito vasos colocados em formato de círculo, como mostrado abaixo:</p> <p>Sabe-se o seguinte sobre o arranjo de flores:</p> <ul style="list-style-type: none"> • A, B e C são lírios; F e G são margaridas; J, K e L são rosas; • apenas uma flor deve ser colocada em cada vaso; 	<p>Questão 9. Se L for colocada no vaso 8, e K for colocada em vaso vizinho ao vaso de L, qual das seguintes afirmações é necessariamente verdadeira?</p> <ul style="list-style-type: none"> (A) A é colocada no vaso vizinho ao vaso de B (B) B é colocada no vaso vizinho ao vaso de G (C) G está no vaso diametralmente oposto ao vaso de J (D) J está no vaso diametralmente oposto ao vaso de A (E) L está no vaso diametralmente oposto ao vaso de A <p>Questão 10. Qual das seguintes flores está necessariamente em vaso vizinho ao vaso de A?</p>

Figura A.7: Extração da questão "Florista" com a ferramenta pdftotext e *flag* -layout

```

Uma florista está arranjando oito flores (A, B,
C, F, G, J, K e L) em oito vasos colocados em
formato de círculo, como mostrado abaixo:
1
5
4
3
2
7
6
8
Sabe-se o seguinte sobre o arranjo de flores:
• A, B e C são lírios; F e G são margaridas; J,
K e L são rosas;
• apenas uma flor deve ser colocada em cada
vaso;
• lírios devem colocados em vasos vizinhos;

```

Figura A.8: Extração de texto da questão "Florista" com a ferramenta pdftotext e *flag* -raw