

Análise e conversão de tablóides de promoções

Igor dos Santos Montagner

Orientador: Prof. Dr. Roberto Marcondes Cesar Junior

21 de novembro de 2010

Agradecimentos

Agradeço primeiramente a Celina Maki Takemura e à Diretoria de Inovação do Ibope Media pelo apoio ao trabalho e pelo tempo de estágio cedido para trabalhar neste trabalho de formatura. Agradeço ao meu orientador, Roberto Marcondes Cesar Junior, pelas sugestões dadas ao trabalho e ao professor Roberto Hirata Júnior pelo incentivo que me deu durante o curso. Finalmente, agradeço à minha namorada Karina e à minha família pelo apoio e por me ouvir comentar tantas vezes sobre este trabalho.

Sumário

1	Introdução	4
1.1	Motivação	4
1.2	Objetivos	4
1.3	Organização do trabalho	5
2	Arquitetura de um sistema de leitura de tablóides	6
3	Segmentação dos tablóides	8
3.1	Segmentação do <i>background</i> e identificação de produtos	8
3.2	Segmentação dos produtos encontrados	13
4	Detecção de textos e OCR	16
4.1	Detecção de nomes e preços	16
4.2	OCR	18
4.2.1	Limitações encontradas	20
5	Análise de resultados	21
5.1	Segmentação dos tablóides	21
5.1.1	Detecção de produtos	21
5.1.2	Criação do recorte e detecção de linhas	21
5.2	Detecção de Textos e OCR	22
6	Conclusões	24
6.1	Sobre o trabalho realizado	24
6.2	Análise subjetiva	24
6.2.1	Desafios e dificuldades encontradas	24
6.2.2	Disciplinas relevantes para o trabalho	25
6.2.3	Trabalhos futuros	26

1 Introdução

Com os avanços na área de computação e a demanda crescente por informação, é evidente a necessidade de buscar, indexar e armazenar documentos e informações de forma eficiente. Documentos em formato eletrônico podem suprir melhor estas necessidades, pois possuem vantagens significativas em relação ao papel. Mesmo assim, a quantidade de documentos em papel usada atualmente é muito grande e uma tarefa muito comum em algumas áreas, como análise de mercado e pesquisas de opinião, é a conversão manual das informações contidas em papel para formatos eletrônicos que facilitem a tarefa de análise. Sendo assim, existe um esforço de pesquisa em busca de uma técnica capaz de automatizar esta conversão de forma robusta. Neste trabalho, será tratada a conversão e análise de um tipo específico de documento: tablóides de promoções distribuídos em varejo.

1.1 Motivação

Para o consumidor, o trabalho ajudaria a encontrar os produtos em promoção em cada estabelecimento e a comparar estabelecimentos para a decisão do local com melhores preços.

A leitura de tablóides de promoções possui uma característica peculiar: não existe uma ordem explícita de leitura de uma página, diferentemente de outros documentos em que existe semelhança com outros documentos do mesmo tipo, como artigos científicos, revistas e jornais (trabalhos como [5] apresentam modelos para a determinação da ordem de leitura de uma página). Em um documento que não possui ordem explícita de leitura, é natural que algumas partes sejam examinadas primeiro, enquanto outras recebem atenção mais periférica. Esta seleção de quais partes serão melhor examinadas é chamada de *atenção seletiva* e é descrita em [4] juntamente com diversos sistemas de computação que usam *atenção visual* para analisar imagens e selecionar regiões de interesse. Outros trabalhos, como [1, 2, 3], também descrevem abordagens para análise de imagens baseada em *atenção visual*. Neste trabalho, os conceitos apresentados em [4] são utilizados para a construção de um sistema para a análise de tablóides.

1.2 Objetivos

O trabalho de leitura dos tablóides foi dividido em duas partes: *segmentação e detecção de textos e OCR(reconhecimento ótico de caracteres)*. A etapa de *segmentação* corresponde a, dada uma página de tablóide, identificar os

produtos nela presentes e fazer um recorte que contenha a imagem do produto, seu preço e nome. A *etapa de detecção de textos e OCR* compreende a localização do nome do produto e do preço dentro de um recorte obtido na segmentação e o uso de um programa de OCR para a obtenção da informação textual contida na imagem. Os objetivos do trabalho são construir um sistema que:

- produza uma listagem com os preços de cada produto encontrado nas páginas de um tablóide;
- torne viável a análise de tablóides de diferentes estabelecimentos comerciais.

A figura abaixo ilustra o resultado desejado.



→ Coca Cola normal ou PET 600ml ... 1,79
 → Sorvete Nestlé vários sabores - 2L ... 11, 80
 ...
 ...

1.3 Organização do trabalho

Na seção 2 é explicada a arquitetura de um sistema para análise e leitura de tablóides que seja robusto e expansível a vários tipos de tablóides. A seção 3 descreve a etapa de segmentação de uma página de um tablóide. A seção 4 descreve a etapa de detecção de textos e uso de um programa de OCR. Na seção 5 são apresentados os resultados obtidos e na seção 6.1 são explicitadas as conclusões finais do trabalho, assim como uma avaliação do cumprimento dos objetivos propostos. Na seção 6.2 é apresentada uma análise subjetiva do curso do Bacharelado em Ciência da Computação.

2 Arquitetura de um sistema de leitura de tablóides

Um sistema que faça análise de tablóides deve ser robusto e expansível, pois precisa lidar com a constante modificação dos formatos de tablóides de promoções. Nesta seção será descrita a arquitetura de um sistema de leitura e análise de tablóides para realizar a leitura de vários tipos de tablóides.

Apesar da constante modificação do formato, visual e cores dos tablóides, existem alguns elementos padrão nesta classe de documentos que permitem que seja criado um algoritmo de análise de em que cada etapa possa ser adaptada para um tipo específico de tablóide. Fotos, nomes e preços dos produtos estão presentes em todos os tipos, juntamente com detalhes do *background* da página. Abaixo está descrito um algoritmo de análise que utiliza-se destes elementos comuns para possibilitar a análise de vários formatos de tablóides. Veja, na figura 2, a sequência dos passos.

1. Segmentação do *background*; define quais partes da imagem serão ignoradas no restante do processo;
2. Detecção dos produtos presentes;
3. Definição de uma vizinhança, chamada neste trabalho de *recorte*, em torno cada produto. Dentro do recorte de um produto devem estar seu nome e preço;
4. Detecção das regiões que contém o nome e o preço do produto analisado dentro do recorte de cada produto.

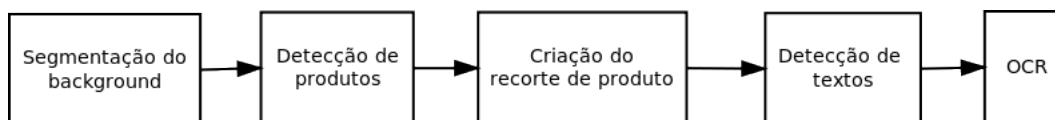


Figura 1: Sequência de passos do método proposto

Este algoritmo é um esqueleto em que cada etapa deve ser adaptada para o tipo de tablóide desejado. Esta arquitetura é definida em [6] como um *Template Method*. No sistema desenvolvido, a classe *Analizador* representa o algoritmo acima e cada um de seus métodos é um passo do algoritmo que deve, obrigatoriamente, ser sobrescrito. No sistema descrito neste trabalho, para dar suporte à análise de um novo tipo de tablóide deve-se criar uma

subclasse de *Analizador* e sobrescrever os métodos correspondentes a cada um dos passos acima. Veja na figura 2 um diagrama UML da classe *Analizador* e suas possíveis especializações para cada tipo de tablóide.

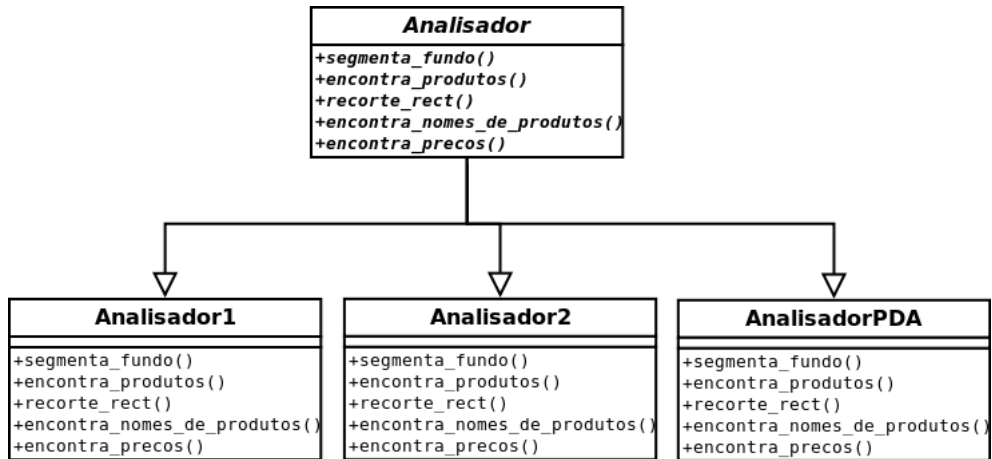


Figura 2: Arquitetura do sistema que possibilita extensão dos tipos de panfletos analisados

Os resultados deste algoritmo (recorte do produto, detecção da região do nome e preço) são usados pelo sistema para passar um programa de OCR nas regiões de texto e gerar um arquivo texto contendo os nomes de cada produto presente na página, juntamente com os preços correspondentes.

Neste trabalho serão descritas as etapas para a criação de uma especialização de *Analizador*, chamada *AnalizadorPDA*, que é capaz de analisar tablóides recolhidos em um supermercado entre os meses de maio e junho de 2010.

3 Segmentação dos tablôides

Para uma segmentação da página estar correta, é necessário que todos os produtos, juntamente com seu nome e preço, apareçam em pelo menos uma divisão. As divisões identificadas não precisam formar uma partição da imagem. Esta definição inclui os *Grupos de produtos*, que são um conjunto de vários produtos muito próximos um do outro e cujos nomes e preços estão associados ao grupo todo e não a um produto em específico. Veja a figura 3 para um exemplo ilustrativo.



Figura 3: (a) e (b) representam divisões válidas e (c) é um *Grupo de produtos*

As seguintes etapas são realizadas para obter uma segmentação correta: segmentação do *background*, detecção dos produtos da página e identificação de divisões entre produtos.

3.1 Segmentação do *background* e identificação de produtos

A primeira etapa para a segmentação do tablôide é a segmentação do *background* e identificação dos produtos presentes em cada página. A abordagem

descrita nesta seção tem uma restrição importante: *em qualquer imagem analisada, um leitor humano deveria considerar os produtos da imagem como maiores focos de atenção*. Em outras palavras, quando uma pessoa comum lê o tablóide pela primeira vez, seu olhar se dirige primeiramente para os produtos e depois para os outros elementos da página, como preços, nomes de produtos e detalhes do plano de fundo.

Em alguns trabalhos, por exemplo [1, 2, 3, 4], o contraste, seja entre pixels individuais ou entre regiões da imagem, é citado como um componente importante para identificar quais áreas da imagem são possíveis focos de atenção do leitor. Em [1], esta característica é usada como único parâmetro para este fim, como explicado abaixo.

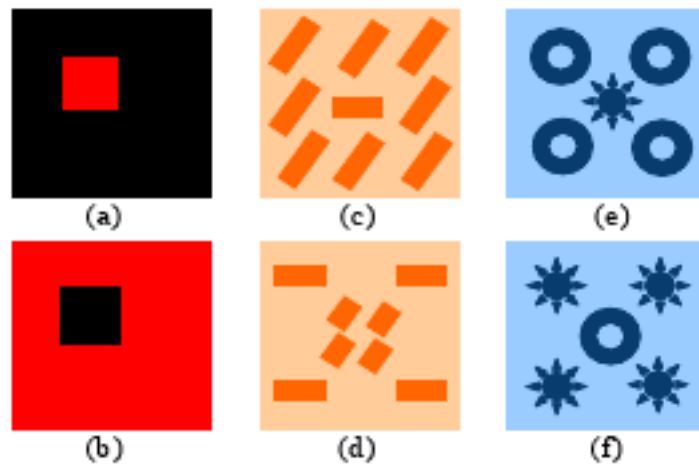


Figura 4: Esta figura, retirada de [1], demonstra que o contraste parece ter mais importância sobre outras características como formato, orientação ou cor para a identificação de regiões de interesse

Em [1], uma imagem de tamanho $M \times N$ pode ser tratada como um *campo perceptivo* composto por $M \times N$ *unidades de percepção*, se cada *unidade de percepção* contiver um pixel. O contraste C_{ij} ($i \in [0, M], j \in [0, N]$) da *unidade de percepção* (i, j) é definido a seguir:

$$C_{ij} = \sum_{q \in \theta} d(p_{ij}, q) \quad (1)$$

onde p_{ij} e q são a quantidade de estímulo recebida pelas *unidades de percepção*, θ é o tamanho da vizinhança analisada e d é a distância entre p_{ij} e q . Neste trabalho, é usada a Distância Gaussiana [3], definida pela equação 2,

onde d_e é a distância euclidiana entre duas cores no espaço RGB e σ é o desvio padrão da gaussiana usada na distância.

$$d(p, q) = 1 - e^{-d_e(p,q)/2\sigma^2} \quad (2)$$

Em [1], a mesma distância é usada, porém com cores no espaço LUV. Outros tipos de estímulo podem ser usados, dependendo do resultado desejado.

Normalizando para $[0, 255]$, os contrastes C_{ij} formam um *mapa de saliências*. É por meio da análise deste mapa que é feita a segmentação do *background* e a detecção dos produtos, que segundo a restrição apresentada no começo da seção, devem possuir os maiores valores no *mapa de saliências*.

Para diminuir o ruído e fazer uma análise multiescala na imagem, é usada uma sequência de imagens em que, dado um elemento i , o elemento $i + 1$ possui a metade das dimensões da imagem anterior. O nível 0 é formado pela imagem nas dimensões originais. Esta sequência é chamada de pirâmide de imagens, sendo usada para fazer uma análise que leve em conta diversas escalas usando a mesma operação em cada uma das imagens geradas. Após este passo, é criada uma função que combina os resultados obtidos em um resultado final. Neste trabalho, a pirâmide contém 5 elementos. Para cada elemento $i \in [0, 4]$, um mapa de saliências MS_i é construído usando o método acima, com $\theta = 3$ e $\sigma = 15$, e o mapa de saliências final MS_f é dado pela seguinte equação:

$$MS_{45ij} = \frac{MS_{4ij} + 5MS_{5kl}}{6}, k = i/2, l = j/2 \quad (3)$$

$$MS_{345ij} = \frac{MS_{3ij} + 5MS_{45kl}}{6}, k = i/2, l = j/2 \quad (4)$$

$$MS_{2345ij} = \frac{MS_{2ij} + 5MS_{345kl}}{6}, k = i/2, l = j/2 \quad (5)$$

$$MS_{fij} = \frac{MS_{1ij} + 5MS_{2345kl}}{6}, k = i/2, l = j/2 \quad (6)$$

Após a criação de MS_f , é aplicado um limiar com valor l_{fundo} para a obtenção de uma imagem binária que representa as regiões da imagem que não fazem parte do *background* do tablóide. Um pós-processamento é feito para eliminar ruído pontual e linhas verticais e horizontais.

Neste trabalho, supõe-se que o leitor tenha algum conhecimento sobre morfologia matemática. Caso contrário, a referência [7] pode ser consultada para maiores informações sobre o assunto. A primeira operação aplicada é uma abertura com um elemento estruturante quadrado 3×3 , para eliminar

pontos pequenos que possam ser muito salientes, mas pequenos demais para representar algum produto. Após isto, duas operações semelhantes são executadas para eliminar linhas verticais e horizontais que, apesar de possuírem alto contraste, fazem parte do *background*. As operações são duas aberturas, uma com elemento estruturante quadrado de 17×1 e uma com tamanho 1×17 .

Com a segmentação do *background* concluída, o próximo passo é classificar cada região em produto ou não. Nos tablóides analisados, o tamanho das regiões pode ser usado para a diferenciação dos produtos com o resto dos elementos da página. A região r_i de tamanho $w_i \times h_i$ é um produto se obedece as seguintes equações, onde $E(w)$ é a média das larguras e $E(h)$ é a média das alturas das regiões.

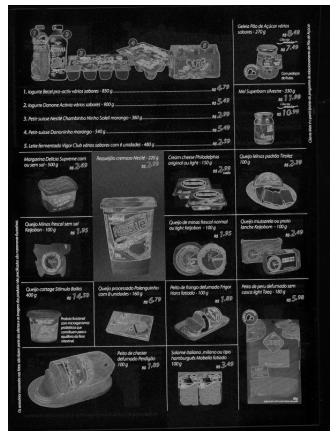
$$w_i/E(w) > \theta_w \tag{7}$$

$$h_i/E(h) > \theta_h \tag{8}$$

Os resultados são expostos na figura 5.



(a)



(b)



(c)



(d)



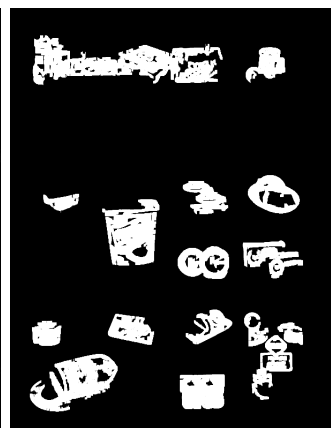
(e)



(f)



(g)



(h)

Figura 5: Em ordem, (b), (c), (d), (e) e (f) são os mapas de saliências intermediários obtidos de (a) e (g) é o mapa final. A figura (h) é a classificação de produtos usando $\theta_w = \theta_h = 2$ e $l_{fundo} = 65$

3.2 Segmentação dos produtos encontrados

Após a identificação dos produtos é feita a segmentação propriamente dita. O algoritmo apresentado nesta seção utiliza as linhas divisórias de produtos encontradas nos tablóides analisados (vide figura 6) para obter o recorte de cada um dos produtos. Devido ao fato que as linhas possuem alto contraste em relação ao fundo, o mapa de saliências será usado. A detecção foi dividida em encontrar as linhas verticais e as linhas horizontais. Só a detecção das linhas verticais será descrita com detalhes. A detecção de linhas horizontais é análoga.

Para detectar as linhas são necessários três parâmetros: a largura máxima max_thick e o comprimento mínimo min_size das linhas a serem realçadas e a mínima distância (min_dist) para que dois segmentos de reta colineares sejam considerados duas linhas separadas. Estes parâmetros evitam que segmentos com comprimento menor que min_size e com largura maior que max_thick sejam considerados linhas e que dois segmentos colineares com distância menor que min_dist sejam considerados duas linhas separadas. Estes parâmetros são usados pela operação morfológica descrita na equação 9, onde L é a imagem com as linhas realçadas, MS_f é o mapa de saliências final exposto na seção 3.1 e A, B, C, D são elementos estruturantes de tamanhos $max_thick \times 1$, 3×3 , $1 \times min_size$ e $1 \times min_dist$. Após esta operação, um limiar é feito para realçar somente as linhas mais salientes.

$$L = (((((MS_f \bullet A - MS_f) \oplus B) \circ C) \bullet D) \quad (9)$$

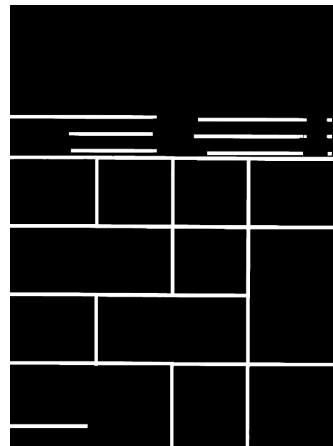
A operação 9 é feita no mapa de saliências MS_f pois as linhas possuem bastante contraste com o fundo e, em MS_f , aparecem realçadas independentemente da cor da linha na imagem original.

A figura 6 mostra alguns resultados. Note que devido ao pouco contraste de alguns quadrados amarelos, não é possível detectar completamente as linhas divisórias destes produtos.

O recorte de um produto p é a caixa envolvente do componente conexo que contém a máscara de p (vinda das imagens da seção 3.1). Se a máscara de p estiver na divisão de dois ou mais componentes conexas, a união destes componentes é usada. Na figura 7 estão imagens contendo a imagem original de entrada e alguns recortes obtidos usando este método.



(a)



(b)



(c)



(d)

Figura 6: Resultados do algoritmo de detecção de linhas com limiar igual a 30. Note a incapacidade do método proposto em lidar com produtos com fundo amarelado.

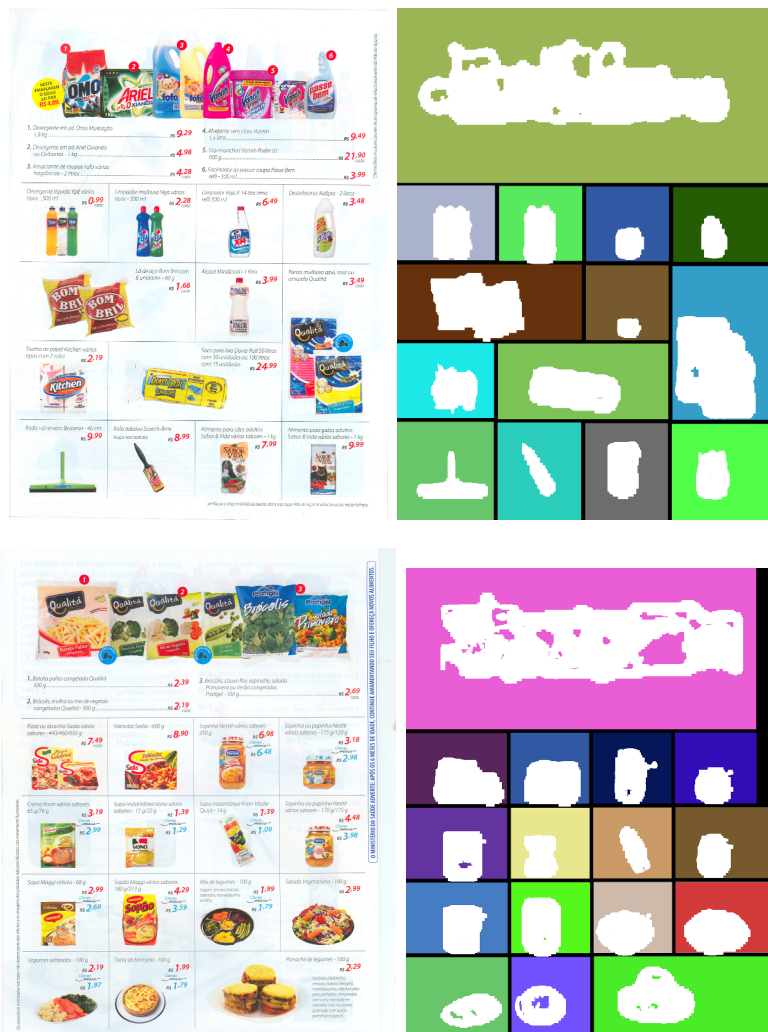


Figura 7: Na primeira coluna está a imagem original e na segunda todos os produtos detectados e seus recortes.

4 Detecção de textos e OCR

Após a segmentação dos produtos, é necessário identificar o nome e o preço de cada produto. Para isto, é necessário identificar quais regiões da imagem contém algum texto e quais são fotos de produtos ou detalhes da página. Com as regiões contendo os nomes de produtos e preços identificadas em cada recorte, estes resultados são passados para um programa de OCR fazer a identificação dos caracteres. Na seção 4.1 é descrita a abordagem usada para a detecção das regiões da imagem que contém texto e na seção 4.2 é descrita a utilização de um programa de OCR para a criação do arquivo de texto contendo os nomes e preços dos produtos.

4.1 Detecção de nomes e preços

Para a classificação de textos e preços foi utilizada uma abordagem baseada em classificação supervisionada. Nesta abordagem, o objetivo é criar um classificador capaz de identificar um conjunto de objetos em uma de N classes. Para isto, uma amostra já classificada dos objetos (chamada de conjunto de treinamento) é dada e, a partir da escolha de características relevantes, é criado um classificador que busca generalizar as informações obtidas do conjunto de treinamento. Definir quais são as características relevantes é um passo importante do processo de criação de um classificador, assim como definir qual é o conjunto de treinamento. Durante esta generalização algumas informações do conjunto de treinamento são perdidas, sendo comum o classificador criado não obter 100% de acerto no conjunto de treinamento. Para maiores informações sobre classificação supervisionada pode-se consultar [9].

O classificador usado neste trabalho foi o *k-Nearest Neighbors* (KNN, ou k-vizinhos mais próximos). Neste classificador, as características extraídas são tratadas como um vetor no espaço R^n e a classificação de um ponto $p \in R^n$ (que representa um objeto a ser classificado) é feita escolhendo a classe mais comum entre os k vizinhos mais próximos de p (vide figura 8). A escolha do classificador foi feita devido à sua simplicidade e facilidade de implementação.

Dado o recorte de um produto p , as regiões que serão passadas para o classificador são obtidas com o seguinte tratamento: primeiro o mapa de MS_f é calculado usando $\theta = 3$. Em seguida é aplicado um limiar com valor 75 e por último é feito fechamento com elemento estruturante 3×3 . Vide a figura 9 para uma ilustração do processo.

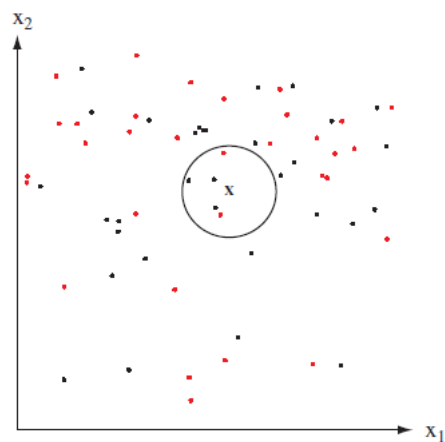


Figura 8: Ilustração do processo de classificação com $k = 5$. A classificação de \mathbf{x} como preto é feita segundo a classe mais comum entre os 5 vizinhos mais próximos

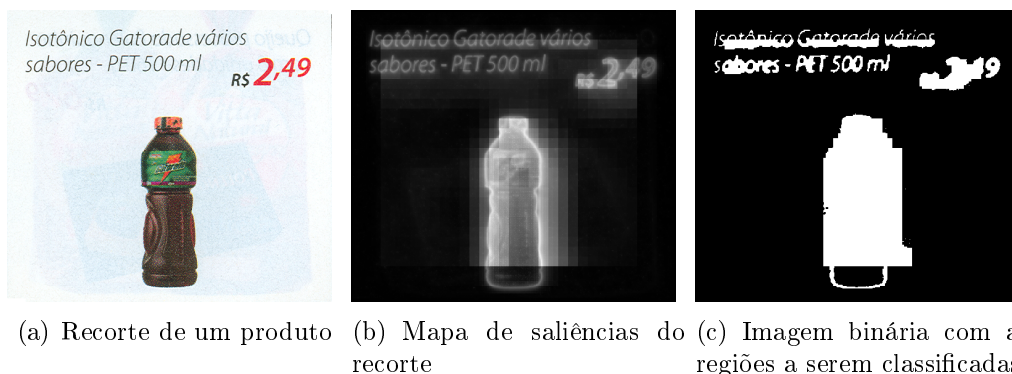


Figura 9: Sequência de imagens ilustrando o processamento feito para a obtenção das regiões a serem classificadas

Foram definidas 3 classes: *nome*, *preço* e *qualquer outra coisa* e extraídas as seguintes características: aspecto, altura, largura, média e desvio padrão da região no mapa de saliências. O treinamento consistiu na extração das características de 192 regiões retiradas de 10 exemplos de recortes fornecidos. Na criação de um AnalisadorPDA, este conjunto de treinamento é carregado e usado nos métodos *encontra_nomes_de_produtos* e *encontra_precos*, que fazem a detecção de regiões contendo nomes de produtos e preços, respectivamente.

Após a classificação das regiões que contém nomes de produtos (ou preços), é realizada uma operação de fechamento de tamanho 15×15 de modo a juntar as regiões que contém letras ou palavras em frases e formar uma única região que contenha o nome do produto ou preço. Também é feita uma dilatação de 25×25 para criar uma borda em torno de cada região de texto. Um exemplo de recorte e as etapas deste processo pode ser visto na figura 10.



(a) Recorte de um produto (b) Regiões do recorte a serem classificadas



(c) Nome de produto está realçado em vermelho e o preço está realçado em azul

Figura 10: Recorte de um produto e o resultado final da detecção de nomes de produtos e preços

4.2 OCR

Após a detecção de nomes de produtos e preços ser concluída, é usado um programa de *OCR* para a leitura dos caracteres presentes na imagem. O programa usado foi o *Tesseract OCR*[8], versão 3.0. O algoritmo descrito nesta seção é repetido para cada produto encontrado e, no final, as informações obtidas são compiladas em um arquivo de texto. É feito um pré-processamento na imagem de modo a diminuir o ruído e melhorar o reconhecimento dos caracteres.

Uma versão em níveis de cinza do recorte é usada nesta etapa. A primeira operação aplicada é um limiar adaptativo, em que o valor $T_I(x, y)$ é calculado para cada pixel segundo a equação a seguir, onde b é o tamanho da vizinhança a ser considerada, $I(x, y)$ é intensidade do recorte no ponto (x, y) , G_b é uma gaussiana com vizinhança de tamanho b e $param1$ é um parâmetro escolhido arbitrariamente.

$$T_I(x, y) = \left(\sum_{k=1}^b \sum_{l=1}^b G_b(k, l) * I(x - k, y - l) \right) - param1 \quad (10)$$

Após a aplicação do limiar, é feito um borrimento gaussiano, de modo a diminuir o ruído na imagem. Esta etapa melhorou consideravelmente a detecção dos caracteres pelo *Tesseract*. O borrimento gaussiano $B(x, y)$ é dado pela expressão 11, onde $I(x, y)$ é intensidade da imagem no ponto (x, y) , σ é o desvio padrão da gaussiana e $b \approx 3\sigma$. Nesta etapa foi usado $\sigma = 5$.

$$B(x, y) = \sum_{k=-b/2}^{b/2} \sum_{l=-b/2}^{b/2} G(k, l) * I(x + k - 1, y + l - 1) \quad (11)$$

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (12)$$

Após este processamento, o programa *tesseract* é chamado via linha de comando com as opções abaixo, sendo que *texto.tif* contém uma região com nome de produto ou preço e o resultado será armazenado em *resultado_ocr.txt*. Não foi feita nenhuma ação especial dependente da classe (nome de de produto ou preço) do texto a ser reconhecido. Veja na tabela 1 alguns resultados.

`$tesseract texto.tif resultado_ocr -l por`

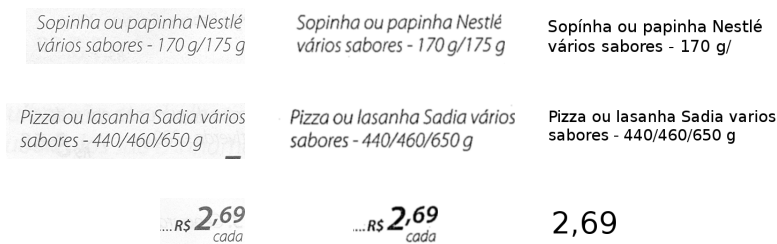


Tabela 1: Na primeira coluna está a imagem original, na segunda a imagem pré-processada e na terceira o resultado obtido pelo *OCR*.

4.2.1 Limitações encontradas

A versão utilizada no trabalho foi a 3.0 do *Tesseract*, porém até 30 de setembro a versão mais atual era a 2.04. Ao utilizar a versão 2.04 algumas limitações foram encontradas. A principal delas foi a qualidade do texto devolvido. Sem o pré-processamento para a redução de ruídos, a detecção gerava texto praticamente ilegível e mesmo após o pré-processamento, a quantidade de caracteres não alfa-numéricos retornados não era desprezível. Também era notável a incapacidade de lidar com imagens contendo texto com fontes de tamanhos diferentes e isto prejudicou bastante o reconhecimento dos preços. Estas limitações foram melhoradas na versão 3.0 e os resultados obtidos melhoraram significativamente, porém o reconhecimento de preços ainda ficou inferior ao de nomes de produtos.

É importante ressaltar que é necessário especificar a língua em que está escrito o texto para o *Tesseract*. Usar a língua correta melhora consideravelmente os resultados da leitura.

5 Análise de resultados

Nesta seção, são analisados e avaliados os resultados obtidos por cada etapa do trabalho. Uma análise neste formato permite detectar quais partes do trabalho poderiam ser melhoradas e quais podem se tornar gargalos do sistema. O conjunto de testes consistiu em 10 páginas de tablóides com um total de 146 produtos.

5.1 Segmentação dos tablóides

A etapa de segmentação será avaliada em duas partes, uma que avaliará somente a detecção dos produtos e outra que tratará do recorte identificado para cada produto.

5.1.1 Detecção de produtos

Os resultados mostrados na tabela 2 foram obtidos ao rodar o programa no conjunto de testes. A coluna Junção/Divisão representa produtos que foram detectados como um só produto ou que foram divididos em dois ou mais produtos.

Falsos Negativos	Falsos Positivos	Junção/Divisão	Acerto(%)
13	12	7	91,1%

Tabela 2: Resultados da detecção de produtos

Os casos de produtos não detectados ocorreram principalmente com aqueles como os da figura 11, que chamam pouca atenção em relação ao fundo da página ou contém grande porção na cor branca. Produtos de limpeza e higiene foram os que tiveram os piores resultados devido ao seu visual conter cores claras e com pouco contraste.

5.1.2 Criação do recorte e detecção de linhas

A principal limitação da detecção do recorte de um produto é a incapacidade de lidar com os produtos com fundo amarelo. Devido a falhas na detecção das linhas divisórias nestes produtos, o recorte feito engloba os produtos vizinhos também. Se o interesse for somente a extração dos nomes e preços dos produtos, é possível fazer a classificação dos nomes de produtos e preços, porém perde-se uma informação valiosa: a associação do nome e preço de um produto com sua foto.



Figura 11: Figuras com pouco contraste que falham na detecção de produtos

De todos os produtos com fundo amarelo, nenhum deles obteve um recorte que envolvesse apenas 1 produto. Na coluna *Erros (fundo branco)* também são considerados erros nos recortes de produtos com fundo branco que fazem vizinhança com produtos com fundo amarelo. No conjunto de testes foram obtidos os resultados apresentados na tabela 3.

Acertos	Erros (fundo amarelo)	Erros (fundo branco)
61%	100%	33%

Tabela 3: Resultados do recorte de produtos

5.2 Detecção de Textos e OCR

O classificador produziu os resultados apresentados na tabela 4 no conjunto de treinamento. Uma classificação de nome de produto foi considerada correta se pelo menos 80% do nome do produto está na região e uma classificação de preço foi considerada correta se todos os números do preço estão presentes na região encontrada. As colunas *Erros nomes* e *Erros preços* contém o número de elementos que não são texto e foram incorretamente classificados como nome de produto e preço, respectivamente. Dentre 146 produtos do conjunto de testes, deveriam ser detectados 185 nomes de produtos e preços.

Nomes detectados(%)	Preços detectados(%)	Erros nomes	Erros preços
89,7%	78,9%	229	71

Tabela 4: Resultados da detecção de textos

Apesar do alto número de regiões classificadas incorretamente, foi possível filtrar as regiões corretas das incorretas pelo resultado do OCR.

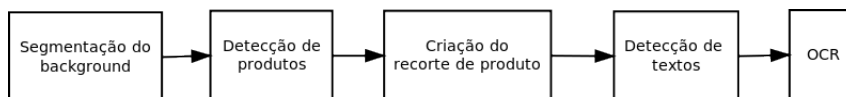
Os resultados obtidos pelo *Tesseract* não foram medidos quantitativamente, porém é possível afirmar que, enquanto a leitura de nomes de produtos produziu resultados legíveis, embora não 100% corretos, a leitura de preços produziu um número bem maior de resultados ilegíveis ou sem sentido.

6 Conclusões

6.1 Sobre o trabalho realizado

A análise e conversão das informações de documentos em papel para formatos eletrônicos é uma tarefa comum em diversas áreas, entre elas análise de mercado e pesquisas de opinião, e existe um esforço de pesquisa para automatizar esta conversão de maneira robusta. Foi apresentado neste trabalho um método para analisar tablóides promocionais.

O método proposto, ilustrado na figura abaixo, permite o reaproveitamento de técnicas usadas para vários formatos de tablóides e minimiza o esforço necessário para suportar um novo formato. Além disto, os elementos considerados (nome de produto, preço, *background*) estão presentes em todos os tablóides promocionais, o que torna possível analisar diversos tipos de formatos.



Os resultados das etapas de detecção de produtos, criação do recorte e detecção de textos foram apresentados na seção 5 e indicam que, apesar de algumas limitações encontradas, o método proposto é viável. Ainda existem muitas melhoras a serem feitas, porém já é possível analisar um tipo específico de tablóide promocional.

Portanto, as técnicas de análise de imagens descritas neste trabalho podem servir como base para a construção de outros sistemas de análise de documentos baseados em *atenção visual* e demonstram um dos possíveis modos para que um sistema possa suportar vários formatos diferentes de documentos de modo robusto e minimizando o esforço necessário para a implementação de um novo formato.

6.2 Análise subjetiva

Esta seção contém uma análise subjetiva do trabalho realizado e sua relação com o curso de Ciência da Computação.

6.2.1 Desafios e dificuldades encontradas

O principal desafio do trabalho foi a definição do método de análise dos tablóides e a extensão deste método para poder aceitar vários formatos de

tablóides. Foram quase 6 meses com diversas tentativas para as detecções dos produtos e segmentação da página e mais 2 meses de trabalhos e testes na detecção de textos e uso de OCR. Este processo foi um pouco cansativo em vários momentos, porém a possibilidade de testar várias técnicas foi valiosa, pois além de permitir escolher a melhor implementação ainda aumentou o aprendizado com o trabalho.

O algoritmo definido no trabalho só foi implementado para um tipo de tablóide e, infelizmente, foi trocado por novo formato. Esta alta rotatividade impede que seja possível trabalhar durante muito tempo com um mesmo formato e a grande diversidade de formatos existentes dificulta o trabalho de análise. Antes de definir junto com meu orientador o algoritmo apresentado na seção 2, o objetivo era produzir um analisador que tivesse suporte para vários tipos de tablóides. Entretanto produzir vários analisadores, um para cada tipo, pode ser mais eficiente. Trocar a abordagem ao problema foi um passo importante e diminuiu a frustração na busca por um algoritmo que pudesse ler qualquer tipo de tablóide.

Um ponto importante que diferencia o aprendizado obtido neste trabalho do obtido durante a graduação é a questão de usar uma técnica visando um objetivo final contra usar uma técnica para analisar seu resultado. Durante o curso, obtive mais conhecimento do segundo tipo, pois os trabalhos e exercícios cobrados pediam, normalmente, para implementar um algoritmo ou técnica e analisar seu resultado. Poucas vezes o objetivo foi “encontre um problema e pesquise como resolvê-lo”, que é o que este trabalho propôs e que possibilitou um aprendizado que envolve o processo de pesquisa para a solução de um problema dado em alto nível. Acredito que os aprendizados são complementares e não é possível ter um aprendizado completo sem os dois lados. Este trabalho foi proveitoso pois permitiu complementar meus conhecimentos, usando os algoritmos e técnicas estudadas durante o curso para resolver um problema “aberto” e que não possui uma solução ideal já conhecida.

6.2.2 Disciplinas relevantes para o trabalho

As disciplinas que foram mais relevantes para o trabalho são as seguintes:

1. **MAC 417 Visão Computacional** - Esta disciplina mostrou diversas técnicas de processamento de imagens e foi possível aplicar várias destas técnicas neste trabalho;
2. **MAC 460 Aprendizagem Computacional** - A parte de classificação de nomes de produtos e preços foi feita com os conhecimentos obtidos nesta disciplina;

3. **MAC 211 Laboratório de programação** - O primeiro contato com visão computacional ocorreu nesta disciplina, em que o projeto era desenvolver um jogo em que a interação com o usuário fosse feita usando uma webcam;
4. **MAC 323 Estruturas de dados e MAC 122 Introdução ao desenvolvimento de algoritmos** - A quantidade de prática no desenvolvimento de algoritmos nestas disciplinas proporcionou uma flexibilidade grande para poder implementar algoritmos e testar ideias rapidamente.

6.2.3 Trabalhos futuros

Existem diversas possibilidades para a continuação dos trabalhos neste projeto, incluindo implementar o suporte a tablóides de outros formatos, melhorar a detecção de linhas na presença de quadros coloridos na imagem, melhorar a classificação de nomes de produtos e preços e testar outros tipos de classificadores.

Referências

- [1] Y. Ma, H. Zhang. *Contrast-based image attention analysis by using fuzzy growing*. Proceedings of the Eleventh ACM international Conference on Multimedia Berkeley, CA, USA. 2003.
- [2] L. Itti, C. Koch. *A saliency-based search mechanism for overt and covert shifts of visual attention* Vision Research, Volume 40, Issues 10-12, Pages 1489-1506, ISSN 0042-6989, DOI: 10.1016/S0042-6989(99)00163-7. 2000.
- [3] H. Liu, S. Jiang, Q. Huang, C. Xu, W. Gao *Region-based visual attention analysis with its application in image browsing on small displays*. In Proceedings of the 15th international Conference on Multimedia (Augsburg, Germany, September 25 - 29, 2007). 2007.
- [4] S. Frintrop, E. Rome, H. I. Christensen. *Computational visual attention systems and their cognitive foundations: A survey*. ACM Trans. Appl. Percept. 7, 1 (Jan. 2010), 1-39. 2010.
- [5] T. M. Breuel. *High performance document layout analysis*. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.13.13032003>
- [6] E. Gamma, R. Helm, R. Johnson, J. Vlissides. *Design Patterns: Elements of Reusable Objected-Oriented Software*. Addison-Weskey Professional; 1 edition, November 10th, 1994
- [7] E.R. Dougherty, R.A. Lotufo. *Hands-on Morphological Image Processing*. SPIE press, 2003
- [8] *Tesseract OCR* <http://code.google.com/p/tesseract-ocr/>
- [9] R. O. Duda, P. E. Hart, D. Stork. *Pattern Classification*. Wiley-Interscience; 2nd edition, October 2000.