UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ

***CAMPUS* LUIZ MENEGHEL**

RICARDO DE OLIVEIRA ALMEIDA

RECONHECIMENTO AUTOMÁTICO DE BLOCOS PARA AUXILIAR A INDEXAÇÃO EM SOLUÇÕES GED

Bandeirantes
2014

Ricardo de Oliveira Almeida

RECONHECIMENTO AUTOMÁTICO DE BLOCOS PARA AUXILIAR A INDEXAÇÃO EM SOLUÇÕES GED

Trabalho de Conclusão de Curso submetido à Universidade Estadual do Norte do Paraná, como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Ederson Marcos Sgarbi

Bandeirantes
2014

Ricardo de Oliveira Almeida

RECONHECIMENTO AUTOMÁTICO DE BLOCOS PARA AUXILIAR A INDEXAÇÃO EM SOLUÇÕES GED

Trabalho de Conclusão de Curso submetido à Universidade Estadual do Norte do Paraná, como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

COMISSÃO EXAMINADORA

_____
Prof. Dr. Ederson Marcos Sgarbi
UENP – *Campus* Luiz Meneghel

_____
Prof. Me. Christian J. de Castro Bussman
UENP – *Campus* Luiz Meneghel

_____
Prof. Me. Bruno Miguel N. de Souza
UENP – *Campus* Luiz Meneghel

Bandeirantes, 17 de Novembro de 2014

LISTA DE ABREVIATURAS E SIGLAS

CMYK	*Cyan, Magenta, Yellow, Black*
Dpi	*Dots Per Inch*
GED	Gerência Eletrônica de Documetos
HSI	*Hue Saturation Intensity*
NC	Nível de Cinza
OCR	*Optical Caracter Recognition*
PDI	Processamento Digital de Imagens
RGB	*Red, Green, Blue*

LISTA DE FIGURAS

SUMÁRIO

RESUMO

O presente trabalho visa o desenvolvimento de uma técnica para reconhecimento automático de documentos, para sistemas de gerência eletrônica de documentos (GED). Sistemas GED atualmente, não possuem um sistema para o reconhecimento automático de documentos, o que torna o armazenamento e a recuperação de documentos muito lenta. A técnica abordada neste trabalho consiste em segmentar a imagem em blocos de informações utilizando a morfologia matemática binária. A partir destes blocos extrair a informação textual de cada um utilizando o motor OCR Tesseract. Então é possível gerar arquivos de textos, que serão úteis para o processo de indexação de documentos. Os testes foram realizados em uma base de imagens contendo 90 documentos públicos, os documentos testados foram: Portarias, Atos Executivos e Ordens de Serviços. Os testes apresentaram resultados promissores, o reconhecimento dos caracteres foi bem sucedido e não houve perda de informações ao realizar o recorte nos blocos de imagens.

Palavras-Chave: 1. Gerência Eletrônica de Documentos, 2. OCR, 3. GED, 4. Morfologia Matemática.

ABSTRACT

This study aims to develop a technique for automatic recognition of documents to electronic documents management systems (EDMS). EDM systems currently do not have a system for the automatic recognition of documents, which makes the storage and retrieval of documents very slow. The technique discussed in this work is to segment the image into blocks of information using binary mathematical morphology. From these blocks extract textual information of each using the Tesseract OCR engine. So you can generate text files, which will be useful for document indexing process. The tests were performed in an image database containing 90 public documents, the documents were tested: Ordinances, Acts Executives and Service Orders. The tests showed promising results, the character recognition was successful and there was no loss of information when making the cut in image blocks.

Keywords: *1. Electronic Document Management, 2. OCR, 3. GED, 4. Mathematical Morphology.*

1. INTRODUÇÃO

O armazenamento de documentos é essencial para todo e qualquer ser humano, quase todas as decisões tomadas geram algum documento, ao se anotar um número de telefone em um papel qualquer, automaticamente gerou-se um documento, ao comprar uma passagem de ônibus um novo documento foi gerado.

Cunha (apud ARAUJO e COELHO, 2009, p. 3) diz que, documento é:

> Informação registrada, produzida ou recebida no início, condução ou conclusão de uma atividade individual ou organizacional, e que compreende conteúdo, contexto e estrutura para fazer prova dessa atividade. [...] Informações registradas, independentes do suporte, produzida ou recebida no decorrer das atividades de uma instituição ou pessoa, dotada de organicidade, que possui elementos constitutivos suficientes para servir de prova dessas atividades.

Com a invenção de novas tecnologias, foram criadas soluções para Gerência Eletrônica de Documentos (GED), afim de otimizar o gerenciamento de documentos. Com a ferramenta GED o armazenamento e a recuperação de documentos se tornam mais usuais, do que se fosse feita manualmente.

Para se trabalhar eletronicamente com um documento, deve haver a conversão do mesmo para o formato digital, processo denominado digitalização. Após a digitalização do documento, ele se torna uma imagem digital. Uma imagem digital não passa de uma matriz de *pixels*, onde cada *pixel* armazena informações da imagem, tornando-a assim manipulável.

A digitalização do documento não garante qualidade à imagem, ou seja após digitalizado há a possibilidade da imagem conter imperfeições. Para tratar tais imperfeições existem técnicas de processamento de imagens, que têm por objetivo deixar a imagem mais limpa.

Após um documento ser convertido em imagem, e a mesma ser processada, ela não será editável, no entanto é possível ter acesso às informações contidas nela utilizando uma ferramenta *OCR Optical Caracter Recognition*. A *OCR* tem a função de reconhecer caracteres contidos no documento digitalizado.

Para fins de indexação e organização, o reconhecimento de *layout* é muito importante para uma aplicação GED, com ele, o usuário não precisa se preocupar em como irá extrair informações da imagem, ou se está extraindo informações a mais, já que será realizada a detecção do *layout* automaticamente, assim gerando blocos de informações contidos no documento serão onde a *OCR* fará o trabalho, de extrair toda a informação textual contida nos blocos, para um texto editável.

1.1. O Problema

Empresas de vários setores trabalham com uma grande carga de documentos diariamente, existem vários fatores que podem gerar problemas a partir deste enorme fluxo de documentos, tais como, armazenamento, preservação, lentidão na busca por documentos.

Sistemas GED atualmente não possuem um sistema de reconhecimento automático de *layout*, o que torna o processo de inserção e indexação do documento mais lento e trabalhoso.

O uso de técnicas de processamento de imagem faz com que a imagem fique mais clara, possibilitando maior desempenho ao reconhecimento dos caracteres pelo OCR. Com o reconhecimento automático de *layout*, o usuário não tem de se preocupar com o nome, local ou tipo de documento, já que esse reconhecimento visa a extração de informações necessárias para a indexação automática do documento.

1.2. Justificativa

Um problema encontrado em empresas e ou instituições públicas é a grande quantidade de documentos impressos, o que necessita de amplo local para o armazenamento, as buscas são mais demoradas, e os documentos não possuem garantias de longa durabilidade.

Hoje os sistemas de gerenciamento eletrônico de documentos *GED*, visa tornar o armazenamento e as consultas mais ágeis. Soluções *GED* utilizam aplicações *OCR*, para extrair caracteres de textos das imagens, o problema é que

OCRs open source reconhecem somente caracteres alfanuméricos, o que deixa inviável a extração de informações de tabelas, textos com figuras.

Apesar da extração da informação textual do documento, não há um reconhecimento do *layout* do documento, o que torna o processo de indexação demorado e trabalhoso, pois a busca por informação se faz por meio de pesquisa *full text*, ou seja há uma pesquisa em todo o conteúdo do mesmo.

Com um reconhecimento automático de *layout*, o sistema fica transparente ao usuário desde aquisição da imagem, até a indexação do documento, o que torna tudo automático e rápido.

1.3. Objetivos

1.3.1. Objetivo Geral

Desenvolver um protótipo que utilize técnicas de processamento de imagens para o reconhecimento automático do *layout* de um documento, para fins de indexação.

1.3.2. Objetivos Específicos

- Utilizar de recursos de processamento de imagens para melhoramento da imagem;
- Definir áreas estratégicas na imagem para que assim ocorra o reconhecimento automático do *layout* do documento e;
- Estudar e implementar a OCR Tesseract na aplicação.

1.4. Organização do Trabalho

O trabalho está organizado da seguinte maneira: O capítulo 2 apresenta a Fundamentação teórica que se divide em: Aquisição, Processamento de Imagens, Segmentação, Morfologia Matemática, Rotulação de Componentes Conexos, Técnica de *Bounding Box*, OCR e Corretor Ortográfico. O capítulo 3 aborda o

método proposto para o desenvolvimento desta pesquisa. O capítulo 4 expõe os resultados experimentais. E por fim o capítulo 5 apresenta a conclusão desta pesquisa científica e possíveis trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

O intuito deste capítulo é abordar a pesquisa bibliográfica deste estudo. Utilizando referências de autores com conhecimento relevante em relação à área de processamento de imagem, gerência eletrônica de documentos, layout de documentos e OCR.

Segundo Marques Filho & Vieira Neto (1999), uma imagem digital pode ser considerada uma matriz, onde linhas e colunas indicam um ponto na imagem, esse ponto é chamado de pixel abreviatura de "*picture element*", cada pixel carrega informações sobre seu RGB, que são os valores que determinam sua respectiva cor.

A partir de Gonzalez & Woods (2008), entende-se que os procedimentos de processamento de imagens digitais podem ser estruturados por etapas, porém não há necessidade de que toda a estrutura participe do processamento, já que o intuito da estruturação é para fins de organização. Ao decorrer deste capítulo entraremos mais a fundo sobre a estrutura do processamento de imagens.

O processamento de imagens é dividido em algumas etapas: Aquisição, Pré-Processamento, Segmentação e Pós-Processamento, mais adiante será explicado cada uma das etapas.

2.1. Aquisição

A etapa de aquisição é onde a imagem é formada, sem nenhuma alteração, os instrumentos responsáveis pela aquisição de imagens podem gerar o sinal analógico da imagem, ou realizar a digitalização da imagem, como o caso de câmeras digitais, ou *scanners*(Alves, 2006)

Para Marques Filho & Vieira Neto (1999), aquisição de uma imagem é a conversão do cenário tridimensional para uma imagem analógica. Neste processo a imagem tem sua dimensionalidade reduzida, deixando de ser tridimensional, assim se tornando bidimensional.

2.2. Pré-Processamento

O processamento da imagem visa preparar a imagem para a análise digital, salienta Alves(2006). Onde a imagem passa por um pré-processamento buscando a correção de problemas advindos da aquisição da imagem. Enquanto a segmentação subdivide a imagem em partes e objetos constituintes, ou seja, a separação de plano de fundo e objeto. Abaixo serão brevemente apresentados alguns métodos utilizados nesta etapa do processamento da imagem.

2.2.1.Filtros

Imagens apresentam áreas com respostas variadas ao eletromagnetismo, áreas estas representadas pela tonalidade. As variações de intensidade por unidade de distância de uma imagem caracterizam a frequência espacial (IBGE, 2000). Facon (2005), diz que há duas classes de filtragem, a filtragem não linear e a linear.

2.2.1.1. Filtro de Média

Segundo IBGE (2000), esse filtro fornece uma suavização, através da substituição do nível de cinza (NC) do pixel pela média aritmética dos *pixels* da máscara. Facon(2005) diz que, o tamanho da máscara varia de acordo com o tamanho da imagem e a presença de ruídos nela. Uma vizinhança variável pode ser utilizada com cada pixel caso seja necessário. A Figura 1 mostra o resultado da imagem após a aplicação do filtro de média.

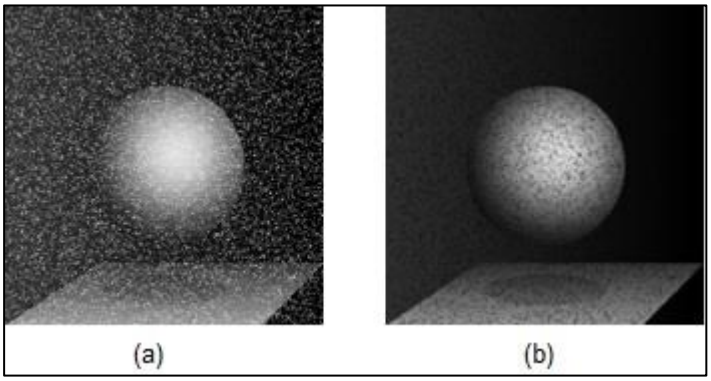


Figura 1: (a) imagem original com ruídos e (b) imagem após aplicado o filtro de Média.

2.2.1.2. Filtro de Mediana

Apesar de não ser um filtro linear como o de Média, o filtro de Mediana caracteriza uma suavização da imagem, variando notavelmente o NC dos *pixels* vizinhos, sendo utilizado também para eliminar ruídos, dentre outros problemas da imagem (BATISTA, 2005 *apud* SOUZA; CORREIA, 2007). A Figura 2 apresenta a aplicação do filtro de mediana em uma imagem ruidosa.

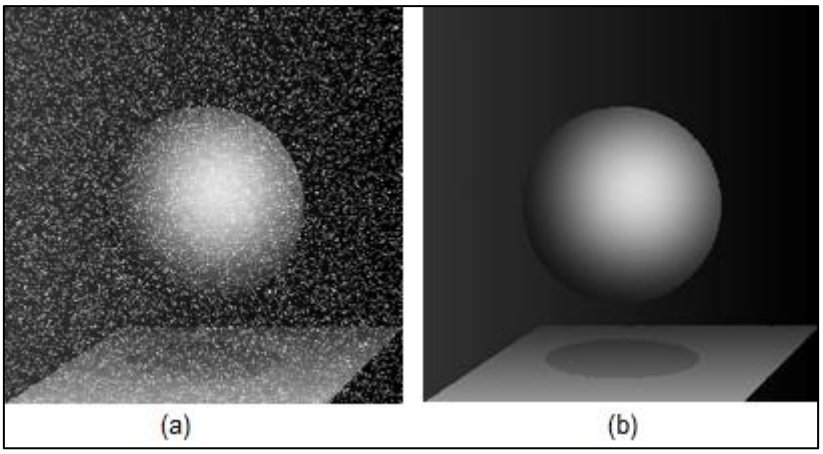


Figura 2: (a) imagem original com ruídos e (b) o resultado da aplicação da Mediana.

2.2.1.3. Filtro de Moda

O filtro da moda de ordem n produz como valor do pixel de saída a moda dos valores dos pixels da imagem de entrada em uma vizinhança de (i, j) contendo n pixels (QUEIROZ; GOMES, 2001). Este filtro é bastante utilizado para limpar *pixels* isolados em certas classes da imagem. A Figura 3 apresenta a aplicação do filtro de moda em uma imagem com ruídos.

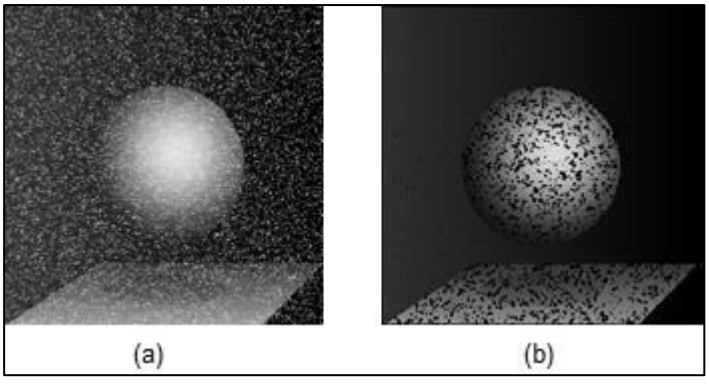


Figura 3: (a) imagem original com ruídos e (b) imagem após aplicação do filtro de moda.

2.3. Binarização

A binarização ou limiarização da imagem, consiste na alteração das cores de uma imagem, de sua cor real para preto e branco. É definido um limiar para que valores acima do limiar alterem o RGB do pixel para 255, onde R = 255, G = 255 e B = 255, têm-se a cor branca, e valores abaixo do limiar têm os valores do RGB alterados a 0, entende-se que o oposto de branco é preto, ou seja ausência de cor (IBGE, 2000). A Figura 4, ilustra a conversão da imagem original para tons de cinza, para posterior limiarização da imagem.

Figura 4: (a) imagem original, (b) imagem em tons de cinza e (c) imagem binária.

O método de Otsu é baseado na análise de discriminante, onde o limiar L dos níveis de cinza é atribuído automaticamente ao código, assim sendo adequado ao processo de binarização (OTSU, 1979).

O método de Johannsen, se baseia na entropia da imagem para realizar o calculo da binarização, primeiro é calculada a entropia para preto *Sb(t)* e para branco *Sw(t)*, depois a imagem é divida em duas partes, reduzindo a correlação entre os *NCs* (NIBLACK, 1986 *apud* SILVA, 2009).

O método de multi-binarização, como o método de OTSU, busca um limiar *L* para repartir a imagem em duas classes. O limiar é escolhido com base no histograma, ele fica localizado em meio a dois picos no histograma, onde cada pico corresponde a uma classe. Este método é o mais indicado para documentos, já que

reconhecimento de caracteres necessitam de imagens binárias (BRITTO JUNIOR et al., 2001).

2.4. Segmentação

Segundo Britto Junior et al. (2001), *"[...]O objetivo da segmentação é obter, a partir de uma imagem, um conjunto de "primitivas" ou "segmentos significativos" que contém a informação semântica relativa à imagem.[...]"*. O problema da segmentação de uma imagem é a incapacidade de saber a quantidade e o tipo de estruturas contidas na imagem, elas podem ser reconhecidas de acordo com sua geometria, topologia, forma, cor, etc., escolhendo as que melhor satisfazem.

Facon (2005), diz que a segmentação por região se da pela ligação de um conjunto de pontos, onde um ponto pode-se chegar a outro da mesma região através de um caminho contido na região, ou seja, essa região apresenta homogeneidade quanto ao NC.

A segmentação por textura, é um pouco mais complexa, mas se houver desenvolvimento de algoritmos próprios para texturas a extração de informações é mais proveitosa para resolver tarefas de segmentação e classificação (FACON, 2005).

A partir de Gonzalez & Woods (2000), nota-se uma certa semelhança com a segmentação por região, porém, há grande divergência, já que a busca pela ligação dos pontos em uma determinada área da imagem, ocorre afim de encontrar uma fronteira, ou borda. Esse processo pode ocorrer local, ou seja, em uma região específica da imagem, através da análise dos *pixels* de uma pequena vizinhança ou global utilizando o calculo da transformada de Hough.

2.5. Morfologia Matemática

Na biologia, a morfologia se refere a estrutura de plantas e animais, enquanto a morfologia matemática, foca seu estudo nos componentes existentes na imagem. A ideia básica da morfologia matemática é a extração de informação em relação à geometria e à topologia de uma imagem (Facon, 2005).

A teoria de conjuntos é utilizada com linguagem da morfologia matemática, assim tornado a morfologia uma abordagem eficiente, para diversos problemas em processamento de imagens Gonzalez & Woods (2000). Na morfologia matemática, os conjuntos são representados pelas formas de objetos contidos na imagem. Em se tratando de imagens binárias esses conjuntos são parte de um espaço bidirecional de números inteiros Z^2, e quando imagens representadas em níveis de cinza, os conjuntos são componentes do espaço Z^3.

2.5.1.Morfologia Matemática Binária

A morfologia matemática binária como o nome já diz, se aplica imagens que possuem apenas *pixels* brancos e pretos. Portanto entende-se que o conjunto que contém todos os *pixels* pretos da imagem, caracteriza a imagem por si só, já que os demais *pixels*, por definição serão brancos. A seguir serão descritos os operadores morfológicos binários.

- **Dilatação:** Segundo Sgarbi (2013) os resultados obtidos ao se aplicar a dilatação na imagem é diminuir e preencher cavidades e aumentar os objetos contidos na imagem, podendo também interligá-los ou não.

- **Erosão:** Já na erosão binária Sgarbi (2013) define que, ao aplicar a erosão o resultado esperado é a diminuição dos objetos existentes na imagem, ou desconectá-los e aumentar e abrir cavidades.

- **Abertura:** Na abertura binária aplica-se primeiro a erosão e posteriormente o resultado da erosão é ditado, a fim de eliminar ruídos (Sgarbi, 2013).

- **Fechamento:** De acordo com Sgarbi (2013), no fechamento binário ocorre é erodido o resultado da dilatação, do fechamento resulta, o fechamento de cavidades, com nenhuma alteração no tamanho dos blocos.

2.6. Rotulação de Componentes Conexos

A conectividade dos *pixels* é essencial quando se trabalha com o processamento de uma imagem, é utilizando o conceito de vizinhança de um *pixel* que são encontradas bordas de objetos na imagem, possibilita manipulação de elementos existentes na imagem (Santiago, 2009).

A técnica de rotulação de componentes conexos baseia-se em rotular os *pixels* de um elemento qualquer da imagem binária com um valor X, de caráter numérico Sgarbi (2013).

Gonzalez & Woods (2000) propõe um algoritmo de rotulação da seguinte forma, Para cada pixel *P* pertencente à figura, tomamos um modelo que representa toda a sua vizinhança (vizinhança-4, -8 ou -m). A cada iteração, procuramos então a vizinhança dos pixels vizinhos a P, e fazemos isso sucessivamente até que todos os pixels da componente conexa tenham sido explorados.

2.7. Técnica de *Bounding Box*

Para Sgarbi (2013) a ideia desta técnica é que para cada bloco, para cada elemento rotulado dentro da imagem, é criado um retângulo de acordo com as coordenadas dos *pixels* da rotulação.

Haralick e Phillips (2014), em seu estudo sobre reconhecimento de estruturas de documentos, definem a *bounding box* como o menor retângulo que compreende um bloco de texto.

2.8. OCR

Soluções OCR, foram criadas com o intuito de reconhecer os caracteres alfanuméricos de uma imagem, e posteriormente transformá-los em um texto editável. A maior parte destas aplicações têm seu funcionamento baseado em apenas dois processos: entrada da imagem e a escolha do idioma (se disponível para escolha), o restante ficam por conta da aplicação (TECMUNDO, 2011).

2.8.1.Tesseract

O Tesseract-OCR é um motor para reconhecimento óptico de caracteres idealizado por Thomas (1987). Este programa foi desenvolvido utilizando a linguagem C++ e teve início em 1984 nos laboratórios da Hewlett-Packard (HP) (THOMAS).

A HP investiu no desenvolvimento do Tesseract-OCR até o ano de 1994, ano em que o mesmo foi relegado para projetos de pesquisa na Universidade de Nevada, Estados Unidos. Esta decisão foi tomada visando concentrar os esforços e investimentos na linha de produtos de escritório da HP, já que o Tesseract-OCR até o presente momento não representava um produto para a empresa. (HOLAHAN, 2006).

Segundo os testes do *Information Science Research Institute* (ISRI), o *Tesseract*-OCR estava entre os três melhores OCR's do mercado, apresentando excelentes resultados contra outros motores de OCR comerciais da época (RICE, JENKINS, NARTKER, 1995).

O seu desenvolvimento foi retomado em meados de 2005, quando a HP tornou o *Tesseract*-OCR um software *open source*, disponibilizando o seu código-fonte para a comunidade científica através do Google Code2 (HOLAHAN, 2006).

No próximo capítulo será explicado passo a passo o método proposto para o desenvolvimento desta pesquisa, explicando cada etapa de seu desenvolvimento.

3. MÉTODO PROPOSTO

O presente trabalho se enquadra em uma pesquisa sobre técnicas de processamento de imagens a fim de encontrar a melhor maneira de realizar o reconhecimento automático de *layout* em documentos. A Figura 5 apresenta as etapas implementadas para o processo de reconhecimento automático de layout e indexação.

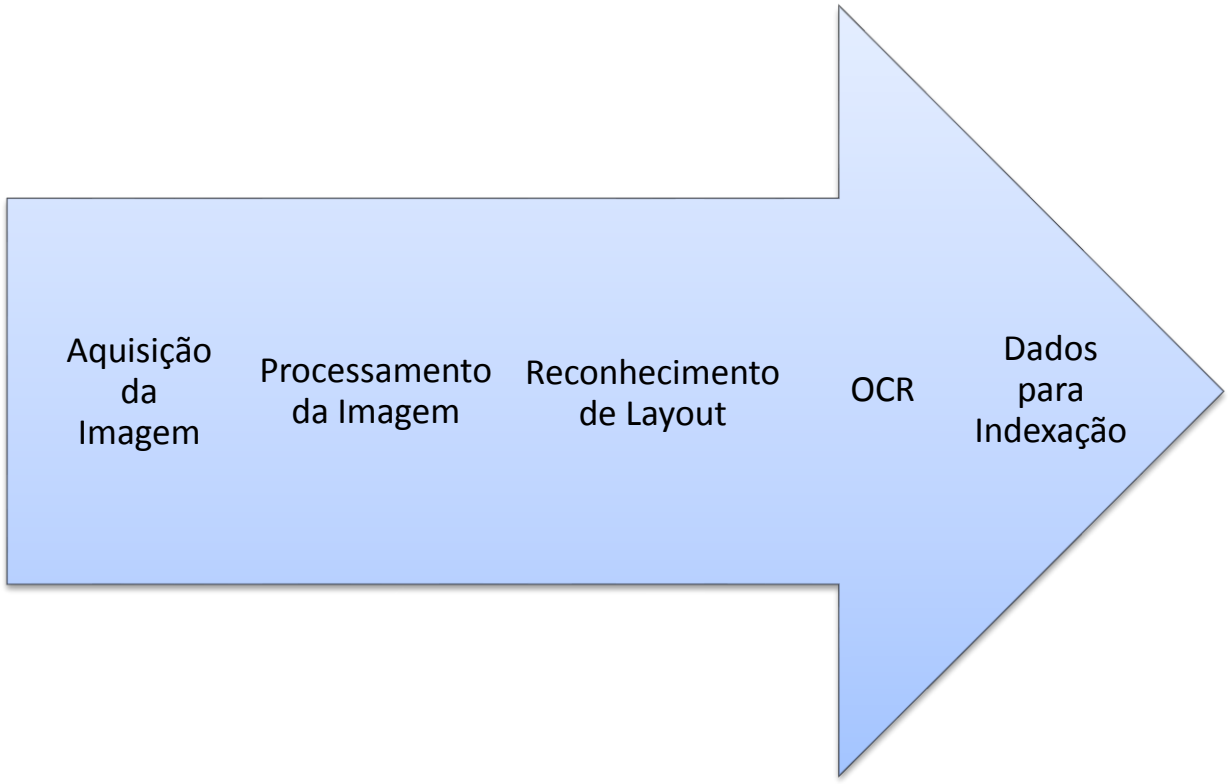
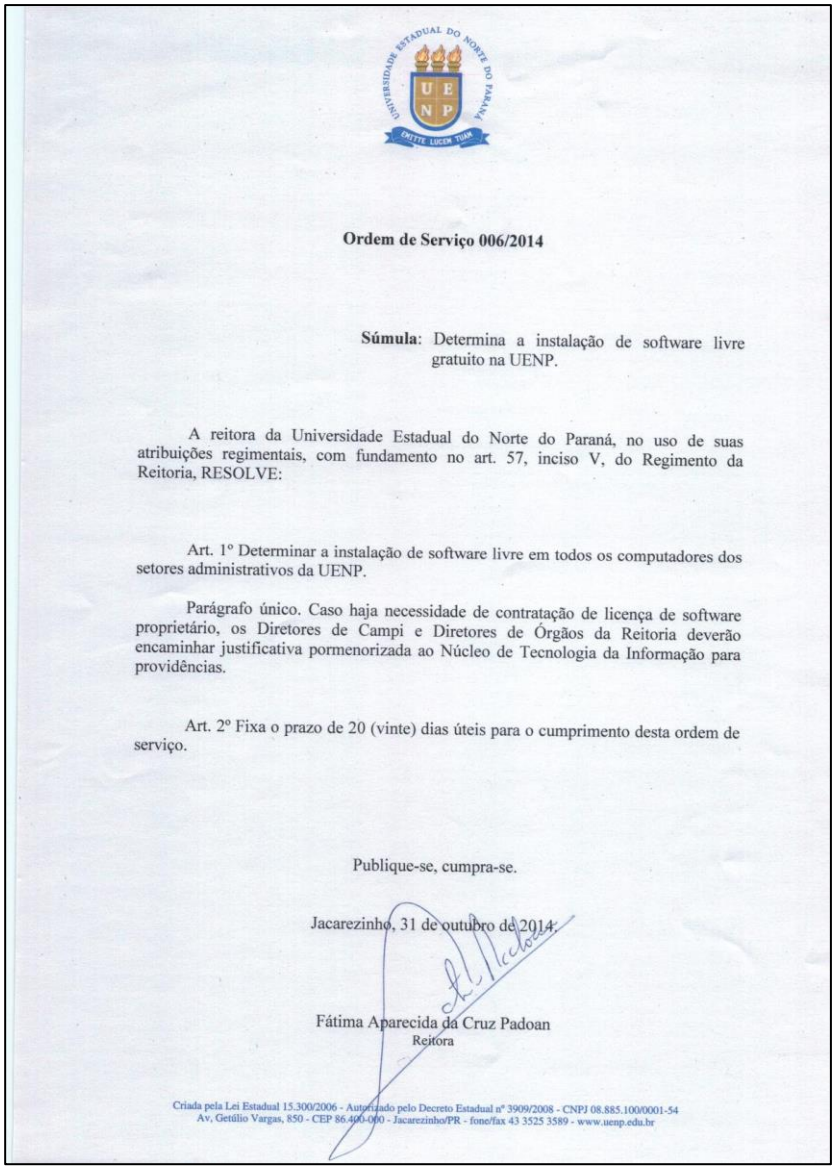


Figura 5: Etapas de todo o processo para o reconhecimento de layout e posterior indexação.

De acordo com a Figura 5 é preciso seguir 05 etapas para o reconhecimento automático de *layouts* que serão descritos a seguir.

3.1. Aquisição da Imagem

A digitalização do documento foi realizada através de um *scanner* comum, com a resolução de 150 *dpi*. O processo de digitalização é rápido, e muito simples. A Figura 6 mostra o documento digitalizado.



Ordem de Serviço 006/2014

Súmula: Determina a instalação de software livre gratuito na UENP.

A reitora da Universidade Estadual do Norte do Paraná, no uso de suas atribuições regimentais, com fundamento no art. 57, inciso V, do Regimento da Reitoria, RESOLVE:

Art. 1º Determinar a instalação de software livre em todos os computadores dos setores administrativos da UENP.

Parágrafo único. Caso haja necessidade de contratação de licença de software proprietário, os Diretores de Campi e Diretores de Órgãos da Reitoria deverão encaminhar justificativa pormenorizada ao Núcleo de Tecnologia da Informação para providências.

Art. 2º Fixa o prazo de 20 (vinte) dias úteis para o cumprimento desta ordem de serviço.

Publique-se, cumpra-se.

Jacarezinho, 31 de outubro de 2014.

Fátima Aparecida da Cruz Padoan
Reitora

Criada pela Lei Estadual 15.300/2006 - Autorizado pelo Decreto Estadual nº 3909/2008 - CNPJ 08.885.100/0001-54
Av. Getúlio Vargas, 850 - CEP 86.400-000 - Jacarezinho/PR - fone/fax 43 3525 3589 - www.uenp.edu.br

Figura 6: Documento Digitalizado.

Após a aquisição da imagem, a mesma deve passar por um processamento, que é composto por vários filtros, com o intuito de remover ruídos advindos da aquisição ou até mesmo da própria imagem original.

3.2. Processamento da Imagem

Após obtida a imagem em cores, primeiramente ela é convertida para escala de cinza, para posteriormente ser convertida para o formato binário através de um filtro automático de limiarização. A Figura 7 ilustra os resultados da aplicação dos filtros de escala de cinza e limiarização.

Figura 7: (a) imagem sem tratamento, (b) imagem em tons de cinzas e (c) imagem após a binarização. (Fonte: Autor)

Para que ocorra a remoção dos ruídos, foi utilizado o filtro não linear de Mediana. A Figura 8, mostra o resultado da aplicação do filtro de mediana.

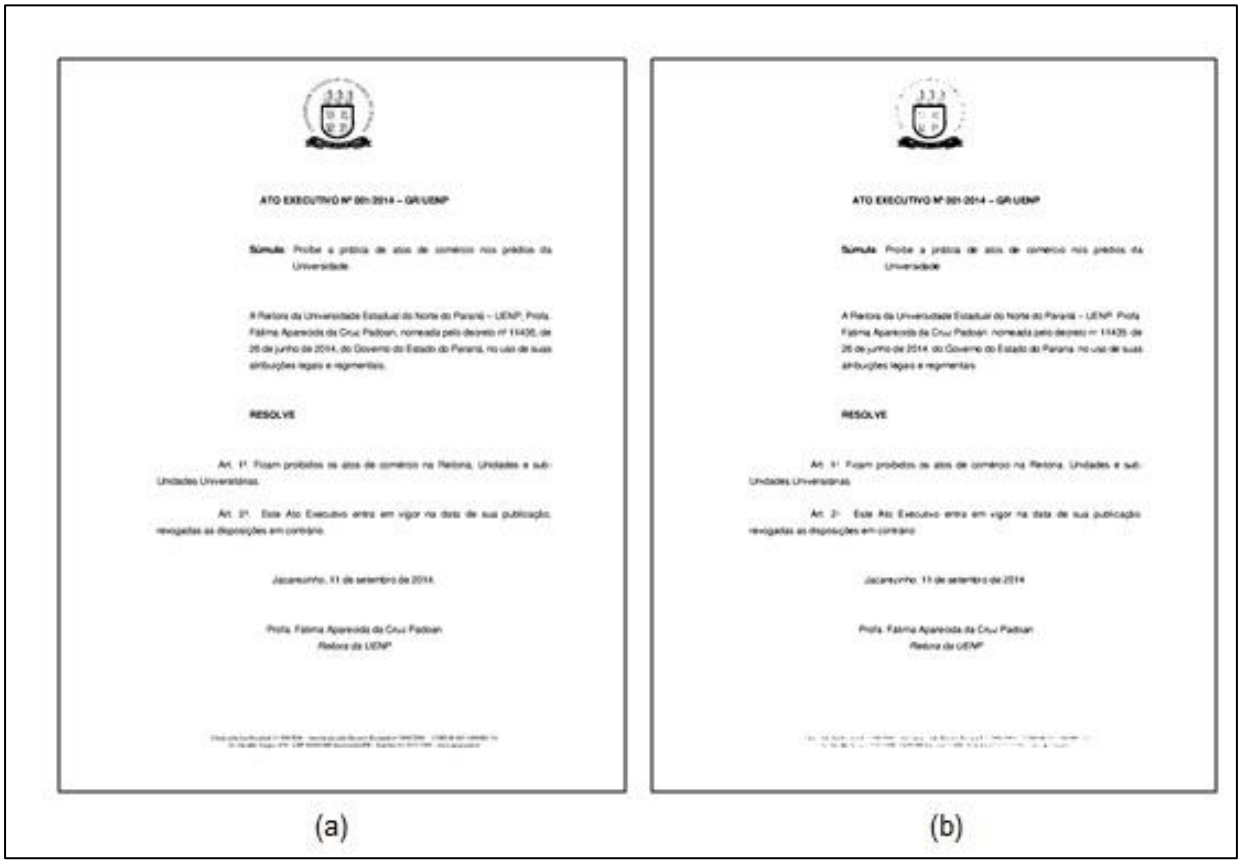


Figura 8: (a) imagem binária e (b) imagem com filtro de mediana.

Após aplicado o filtro de mediana, e livre de ruídos a imagem está pronta para aplicação das técnicas para reconhecimento automático de *layout*. Agora um último filtro é aplicado para a formação dos blocos, aplica-se o operador morfológico binário de erosão, a fim de juntar as linhas mais próximas do documento para que seja possível a formação de blocos. A Figura 9 ilustra a utilização dessa técnica.

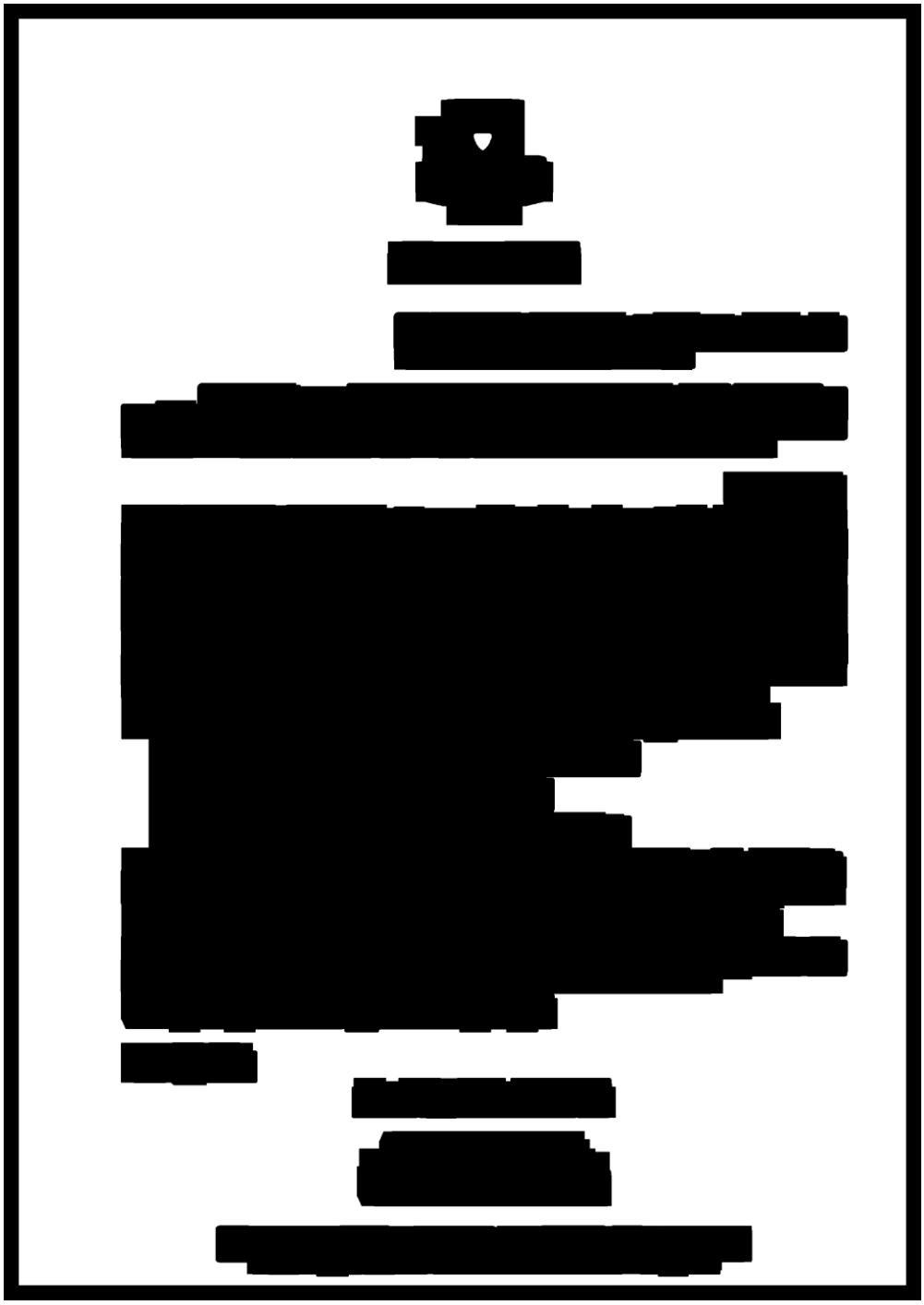


Figura 9: Documento após aplicação de 40 iterações de erosão.

Na seção a seguir será explicado passo a passo, o funcionamento de cada técnica de processamento de imagem, para o reconhecimento automático de *layout.*

3.3. RECONHECIMENTO DE LAYOUT

Neste trabalho o reconhecimento de *layout* do documento se dá, por um corte XY em toda a imagem. Primeiro é delimitado blocos por toda a imagem, e posteriormente é definido qual a relevância para sua indexação. Depois de definidos os blocos, foi aplicado o corte XY para que a OCR seja aplicada separadamente em cada bloco. A seguir será explanado cada passo.

3.3.1.Rotulação de Componentes Conexos

Para a rotulação de componentes conexos, foi utilizada a vizinhança-4. Onde cada bloco preto da Figura 9 é apresentado como um componente conexo. A Figura 10 ilustra cada bloco gerado pintado de uma cor diferente.

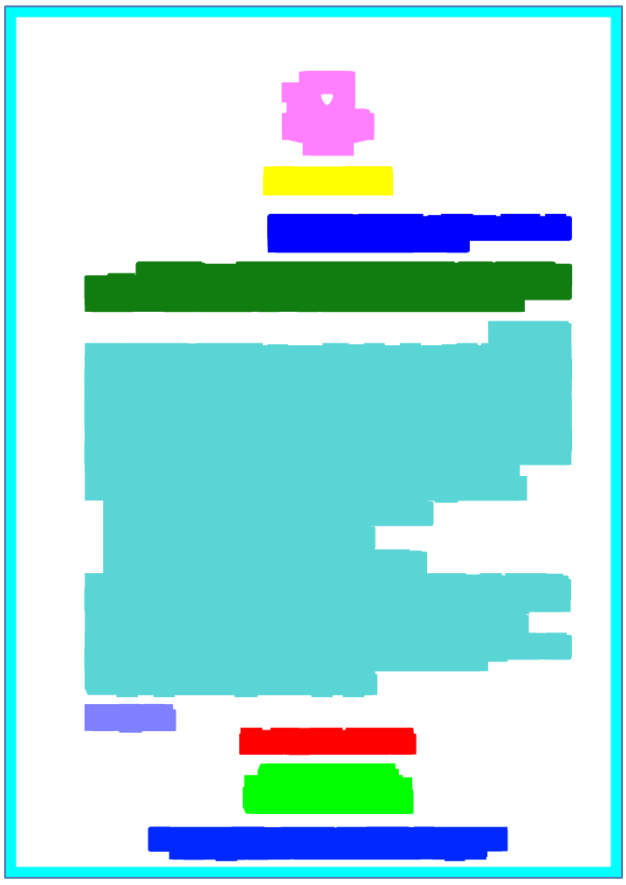


Figura 10: Imagem rotulada com cores.

Embora a Figura 10, ilustre os blocos em cores, eles são rotulados por números. Após o processo de rotulação, a técnica de *bounding box* é aplicada, como veremos na próxima seção.

3.3.2.Técnica de Bounding Box

Após os blocos devidamente rotulados, e definidos os blocos relevantes para indexação, é extraída a menor e maior coordenada X e Y de cada bloco. Após isso são feitos recortes nos blocos, para depois aplicar a OCR. A Figura 11, mostra os retângulos em volta dos blocos significantes, demarcando a imagem com os caracteres.

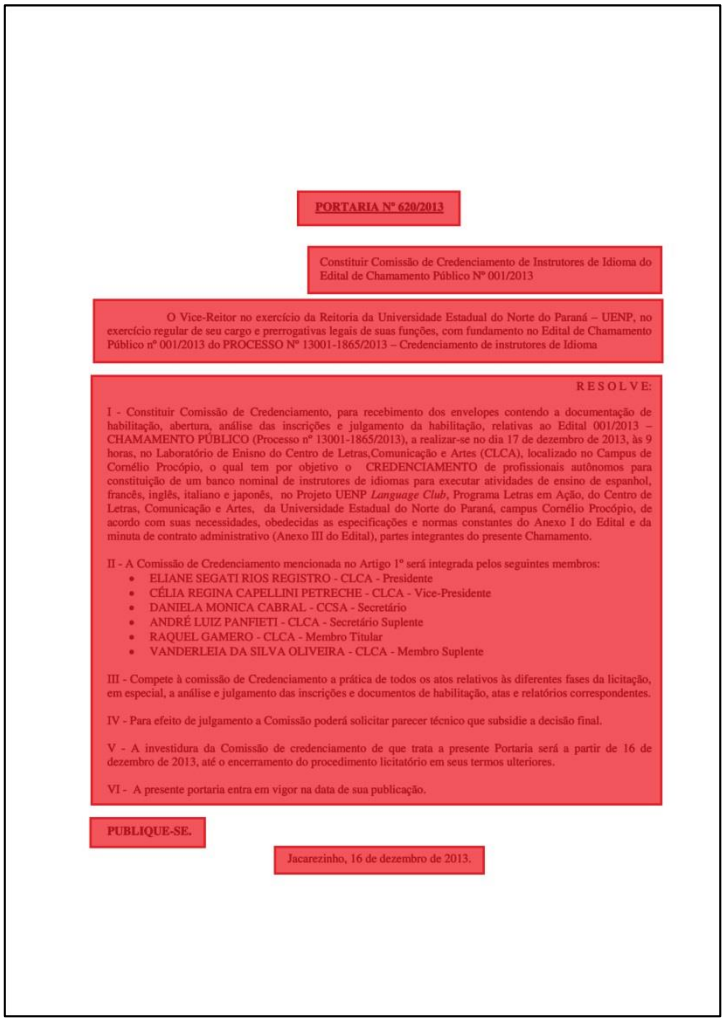


Figura 11: Blocos relevantes demarcados pela técnica de bounding box.

3.4. OCR

Após todo processamento da imagem, e a delimitação dos blocos de informações significativos para indexação, ocorre a aplicação da OCR em cada um dos blocos. A Figura 6, apresenta a aplicação da OCR no boco título de uma portaria.

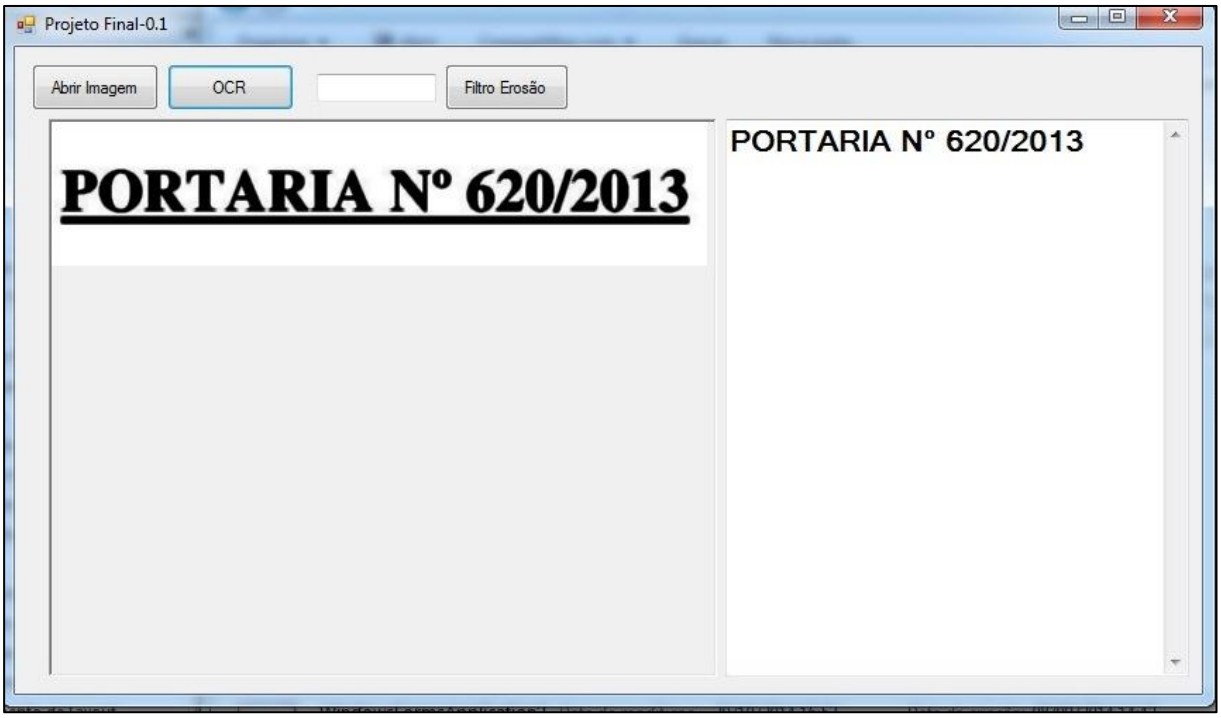


Figura 12: Aplicação da OCR em um dos blocos do documento (Fonte: Autor).

Após a aplicação da OCR, não há garantias que todos os caracteres extraídos da imagem serão reconhecidos.

3.5. Dados Para Indexação

Ao fim de todo este processo, são gerados documentos .txt, que poderão ser utilizados no processo de indexação de imagens. No próximo capítulo serão demonstrados todos os testes realizados para chegar a conclusão deste trabalho.

4. RESULTADOS EXPERIMENTAIS

Neste capítulo, serão expostos os testes realizados e resultados alcançados pelo método proposto. A organização do capítulo está da seguinte forma: na seção 4.1 encontra-se a base de imagens em que foram realizados os testes e, na seção 4.2 o resultado dos experimentos.

4.1. Base de Imagens

Foram testadas 30 imagens de cada tipo de documento, formando uma base com o total de 90 imagens, cada imagem possui dimensão de 1240x1753 *pixels*, no formato JPEG, e resolução de 150 dpi. A base foi obtida através do site da UENP (http://uenp.edu.br/index.php/documentos/cat_view/64-publicacoes-do-gabinete-do-reitor), todas as imagens foram impressas e digitalizadas novamente. A figura 13 ilustra base de imagens.

Figura 13: Base de imagens, sendo (a) Ordem de Serviço 006/2014, (b) Ato Executivo 001/2014 e (c) Portaria 430/2014.

Após um estudo sobre os documentos, foram constatadas as similaridades entre eles, e a partir disso pode se concluir qual parte dos documentos, possuem comum relevância. Na próxima seção serão mostrados os testes realizados para a remoção dos ruídos encontrados nas imagens.

4.2. Eliminação de Ruídos

Os filtros de binarização eliminam grande parte dos ruídos. A Figura 14, mostra os testes entre os algoritmos de OTSU, Johannsen, e Limiarização por Threshold T = 128.

Figura 14: Teste entre algoritmos de binarização, (a) imagem em escala de cinza, (b) imagem binarizada por Johannsen, (c) imagem binarizada por Threshold T = 128, (d) imagem binarizada por OTSU.

Depois de aplicado o filtro de binarização, algumas imagens ainda possuíam pequenos ruídos. Para a remoção dos ruídos foi utilizado fechamento por morfologia matemática. O operador foi testado com três iterações, *i*=1, *i=2, i=3*. A Figura 15 apresenta os resultados do fechamento com as iterações citadas acima.

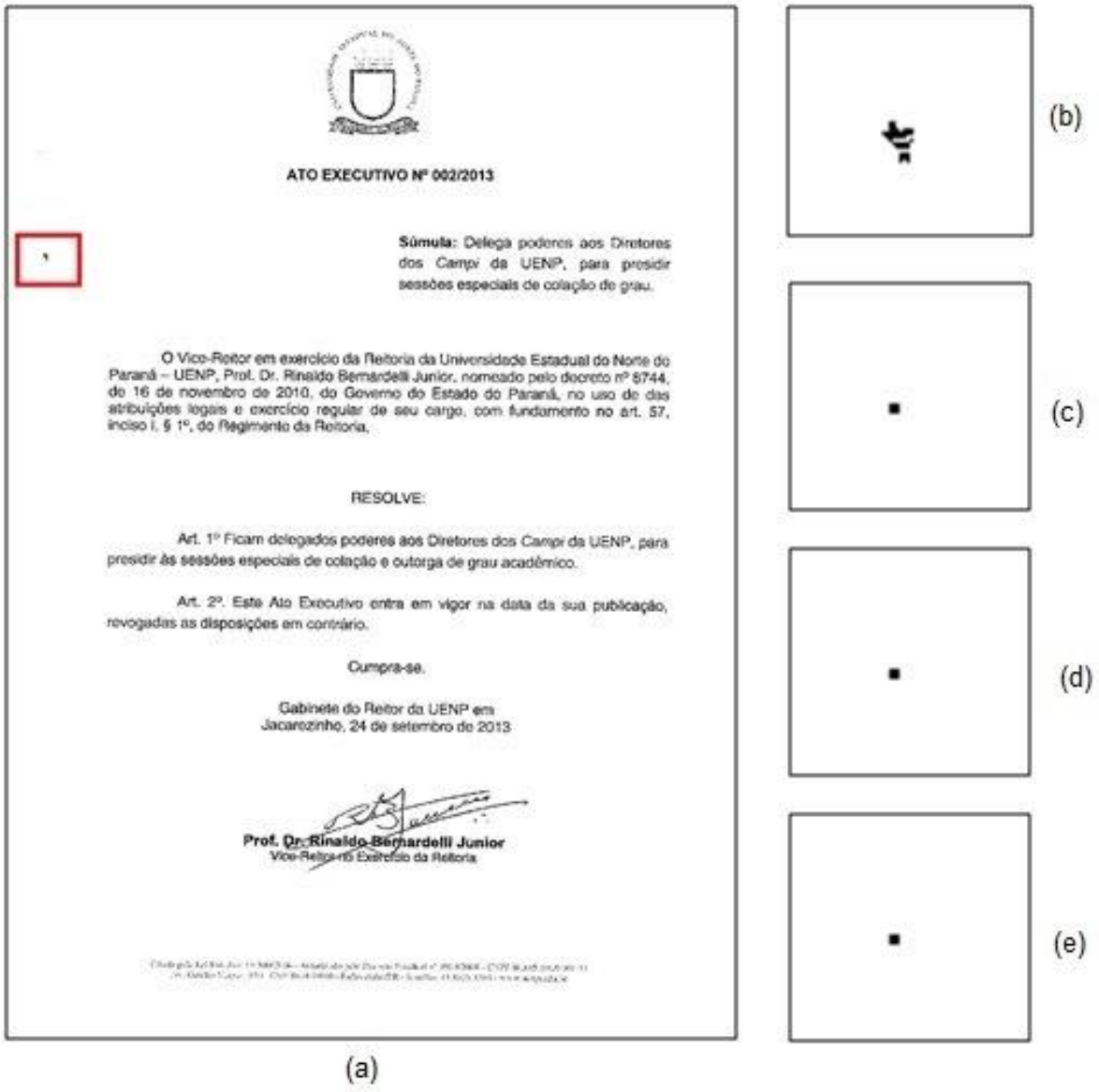


Figura 15: Teste de abertura com algumas iterações, (a) imagem original binária, (b) ruído original, (c) fechamento com *i*=1, (d) fechamento com i=2, (e) fechamento com i=3.

A partir dos testes realizados na Figura 15, foi observado que após a iteração *i*=1, não houve mudança significativa na imagem. O que indica que o filtro de fechamento não remove todo o ruído sozinho.

Também foram realizados testes com o filtro de mediana, que é indicado para remoção de pequenos ruídos na imagem. A Figura 16, ilustra a aplicação do filtro com o mesmo número de iterações, utilizados na Figura 15.

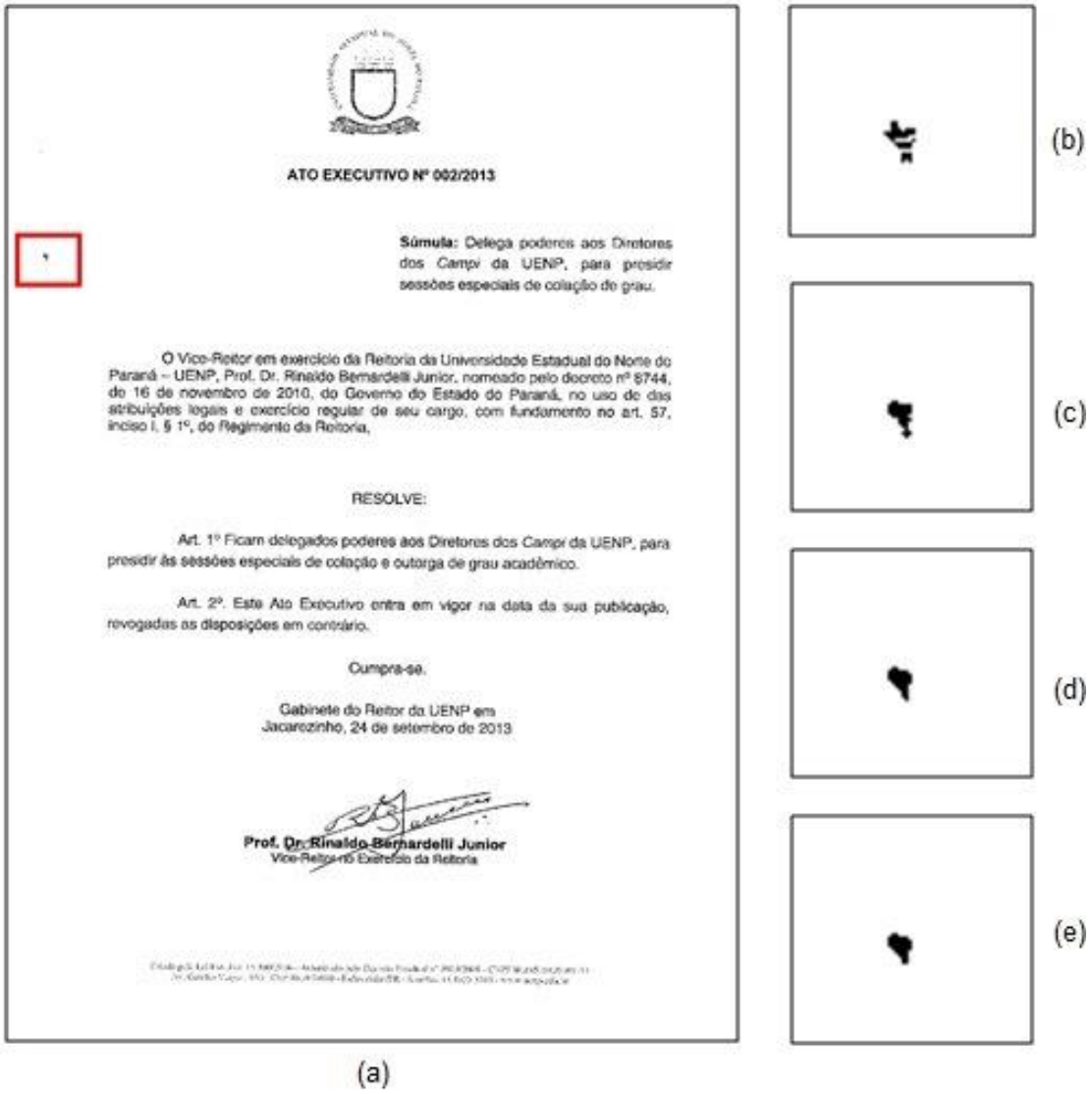


Figura 16: Teste do filtro de mediana com algumas iterações, (a) imagem original binária, (b) ruído original, (c) mediana com i=1, (d) mediana com i=2, (e) mediana com i=3.

Analisando os testes, pode-se perceber que os dois filtros não eliminam o ruído sozinho. O fechamento reduziu consideravelmente o tamanho do ruído em apenas uma iteração, enquanto a mediana o diminuiu aos poucos. A Figura 17 apresenta o resultado da aplicação da mediana com três iterações, *i*=1, *i*=2 e *i*=3, na imagem resultante do fechamento com apenas uma.

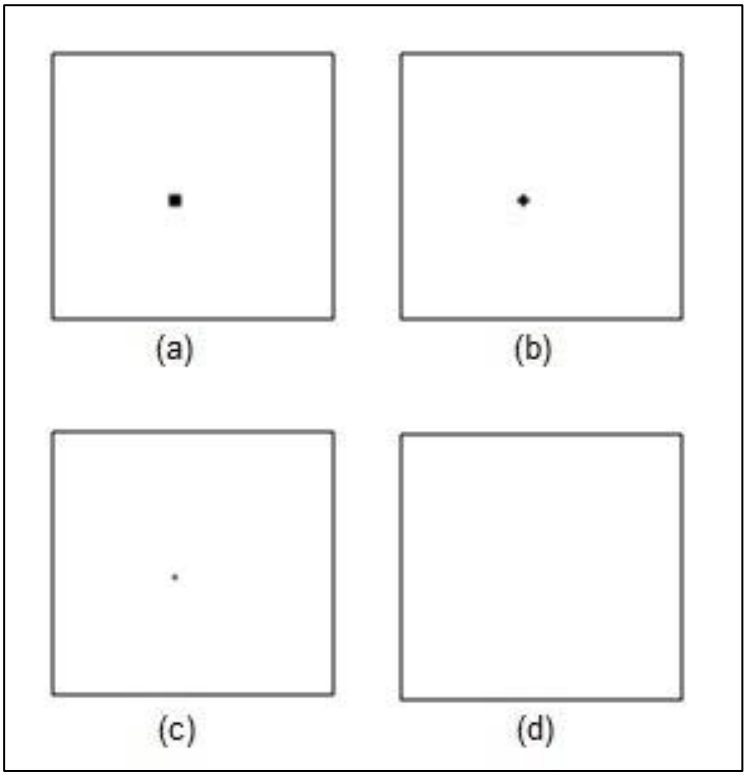


Figura 17; (a) imagem com ruídos, após fechamento com i=1, (b) mediana com i=1, (c) mediana com i=2, (d) mediana com i=3.

De acordo com a Figura 17, a eliminação de ruídos, pode ser alcançada apenas com o fechamento morfológico e com três iterações do filtro de mediana. Para garantir a remoção dos ruídos neste trabalho foi aplicada mais duas iterações do filtro de mediana, porém não foi encontrada nenhuma imagem na base a qual fosse necessária este número de iterações. A próxima seção abordará os testes realizados para a montagem dos blocos.

4.3. *Bounding Box*

Para que a técnica de *bounding box*, seja efetivamente realizada, é necessário que as linhas mais próximas se juntem, assim formando blocos, pelo documento. Para que isso seja possível, a erosão por morfologia matemática foi utilizada. A Figura 18 ilustra os testes realizados para se chegar a um resultado satisfatório, os testes foram realizados com as seguintes iterações, *i*=1*, i*=10*, i*= 20, *i*=30, *i*=40, *i*=60, *i*=70 e *i*=80.

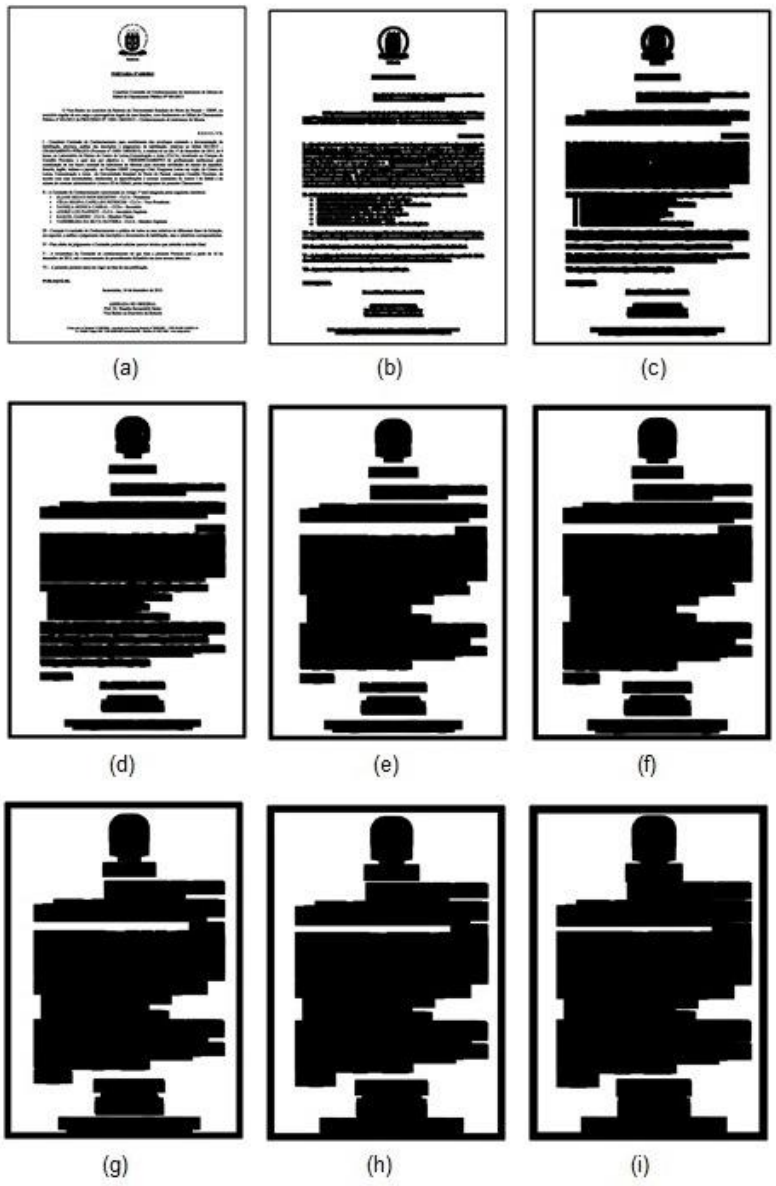


Figura 18: Imagens com diferentes iterações de erosão, (a) i=1, (b) i=10, (c) i=20, (d) i=30, (e) i=40, (f) i=50, (g) i=60, (h) i=70, (i) i=80.

Na Figura 18, pode-se observar que a imagem (e) com 40 iterações foi a que mais se enquadrou nos requisitos para esta pesquisa, ela formou os blocos corretamente, e ainda manteve uma distancia razoável entre eles. A Figura 19 apresenta os blocos rotulados por números.

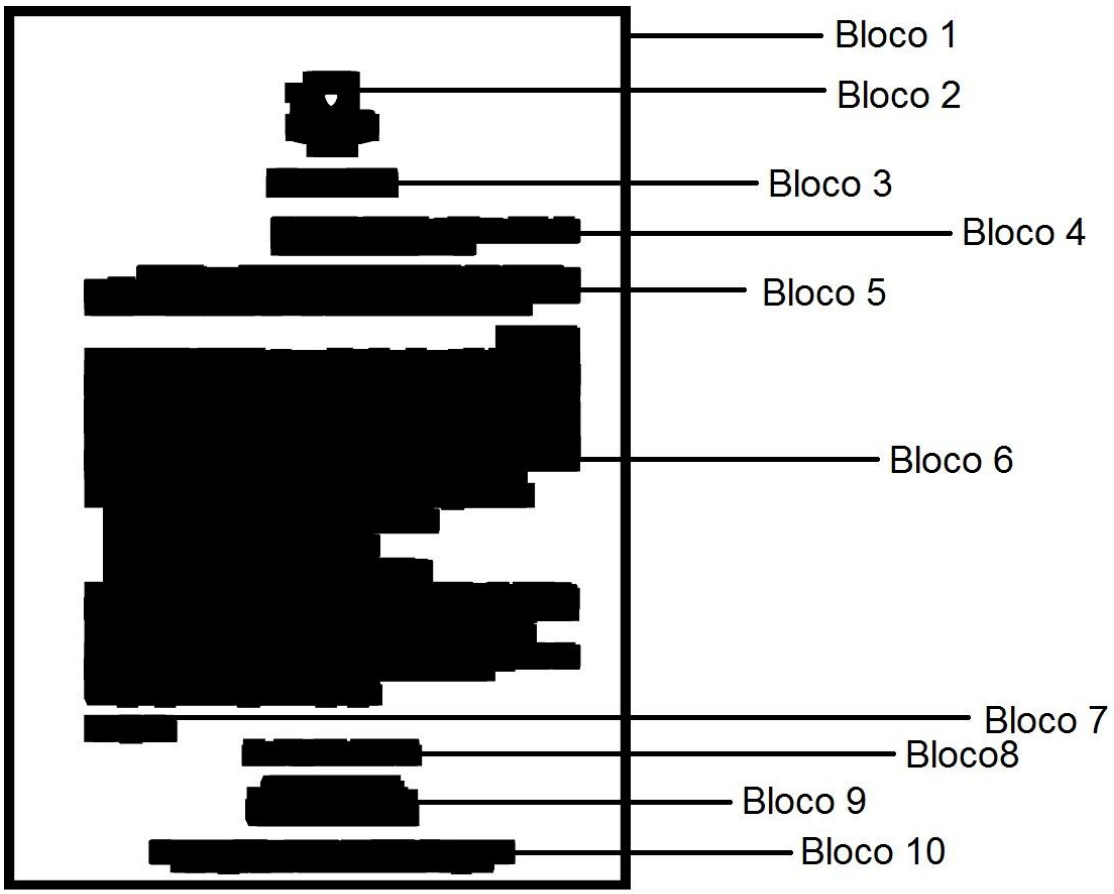


Figura 19: Blocos de imagens rotulados por números.

A Figura 19, mostra os blocos de imagens rotulados por números, desses blocos sabemos que, o bloco 1 é a borda criada após a erosão, o bloco 2 é o timbre da instituição, o bloco 3 é o título do documento, os blocos 4 e 5 fazem parte da súmula, os blocos 6 e 7 são o corpo do texto, ou seja a informação mais relevante do documento, o bloco 8 contém a data de expedição do documento, o bloco 9 a assinatura do responsável pelo documento, e o bloco 10 o rodapé do documento.

Após a análise dos documentos estudados neste trabalho, observou-se que para todos os documentos estudados os blocos serão os mesmos. Os blocos relevantes são os blocos do 3 ao 8, o restante são descartáveis. A Figura 20 mostra os blocos de imagens relevantes para indexação separados, e recortados após todo o processamento.

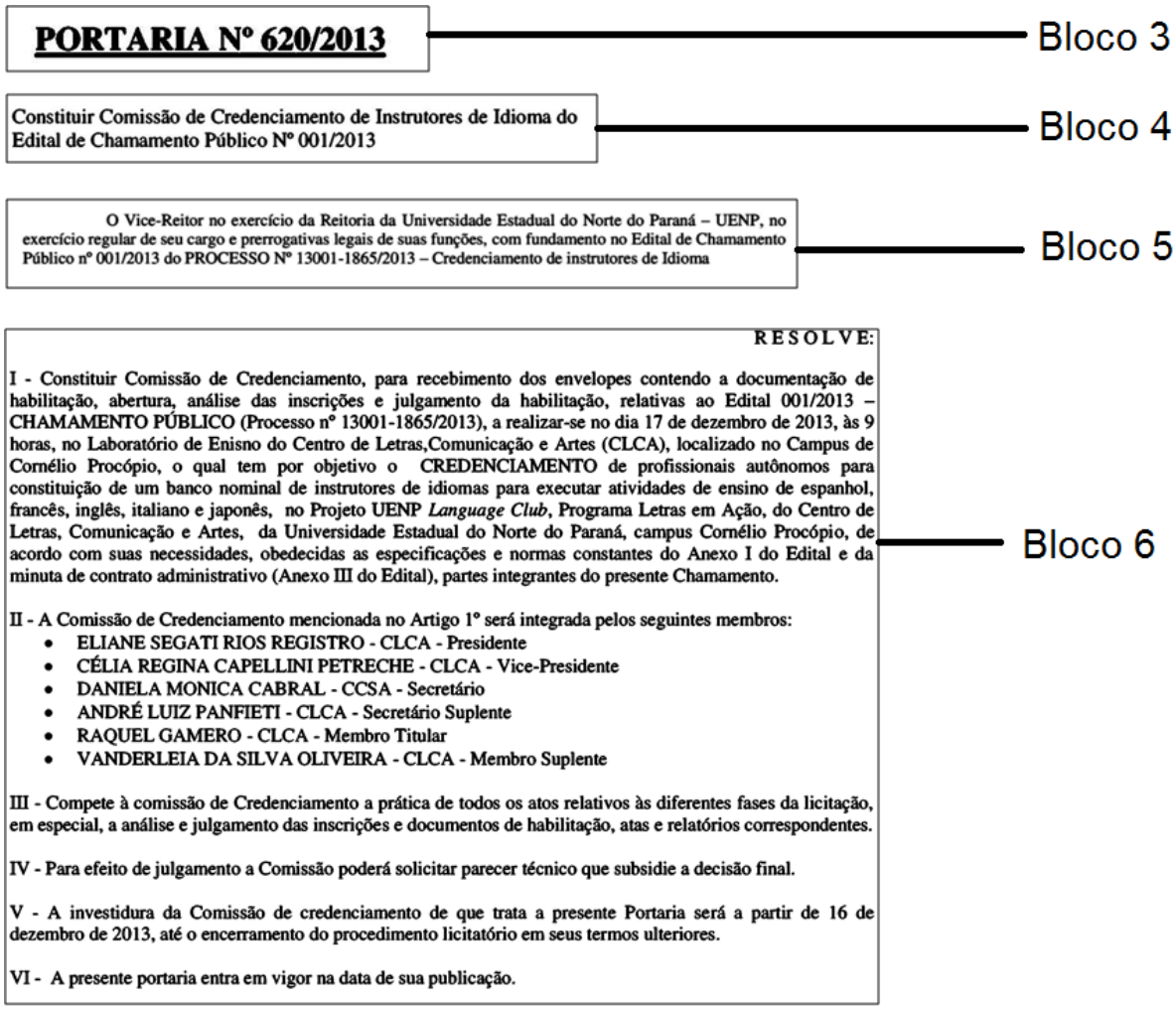


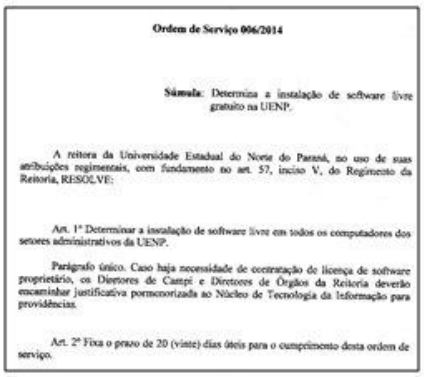
Figura 20: Imagens dos blocos, após a técnica de *bounding box*.

Após separados os blocos de informações da imagem, a mesma esta pronta para a aplicação da OCR. Na próxima seção, será apresentado os resultados aplicados na imagem com a OCR.

4.4. Teste OCR

Para concluir o presente trabalho, a OCR foi aplicada em cada bloco recortado da imagem, para que fosse realizada a extração da informação da imagem em caráter editável.

Foram realizados os testes entre versões da OCR Tesseract 2 e 3. A Figura 21 apresenta um teste realizado, comparando as duas versões.



Ordem de Serviço 006/2014

Súmula: Determina a instalação de software livre gratuito na UENP.

A reitora da Universidade Estadual do Norte do Paraná, no uso de suas atribuições regimentais, com fundamento no art. 57, inciso V, do Regimento da Reitoria, RESOLVE:

Art. 1º Determinar a instalação de software livre em todos os computadores dos setores administrativos da UENP.

Parágrafo único. Caso haja necessidade de contratação de licença de software proprietário, os Diretores de Campi e Diretores de Órgãos da Reitoria deverão encaminhar justificativa pormenorizada ao Núcleo de Tecnologia da Informação para providências.

Art. 2º Fixa o prazo de 20 (vinte) dias úteis para o cumprimento desta ordem de serviço.

(a)

Ordem de Serviço 006/2014

Súmula: Detemtína a instalação de software livre gratuito na UENP.

A reitora da Universidade Estadual do Norte do Paraná, no uso de suas atribuições regimentais, com fundamento no art. 57, inciso V, do Regimento da Reitoria,

RESOLVE: Art. l° Determinar a instalação de soŕware livre em todos os computadores dos setores administrativos da UENP.

Parágrafo único. Caso haja necessidade de contratação de licença de software proprietário, os Diretores de Campi e Diretores de Orgãos da Reitoria deverão encaminhar justificativa pormenorizada ao Núcleo de Tecnologia da Informação para providências.

Art. 2° Fixa 0 prazo de 20 (vinte) dias úteis para 0 cumprimento desta ordem de serviço.

(b)

Ordem de Serviço 006/2014

Súmula: Determina a instalação de software livre gratuito na UENP.

A reitora da Universidade Estadual do Norte do Paraná, no uso de suas atribuições regimentais, com fundamento no art. 57, inciso V, do Regimento da Reitoria, RESOLVE:

Art. 1° Determinar a instalação de soŕware livre em todos os computadores dos

setores administrativos da UENP.

Parágrafo único. Caso haja necessidade de contratação de licença de software proprietário, os Diretores de Campi e Diretores de Órgãos da Reitoria deverão encaminhar justificativa pormenorizada ao Núcleo de Tecnologia da Informação para providências.

Art. 2° Fixa o prazo de 20 (vinte) dias úteis para o cumprimento desta ordem de serviço.

(c)

Figura 21: Parte do texto da imagem original processada, (a) imagem original, (b) texto extraído pela OCR Tesseract 2, (c) texto extraído pela OCR Tesseract 3.

Na Figura acima a imagem (a), é o recorte do documento original, a imagem (b) resulta da aplicação da versão 2 da OCR Tesseract na imagem (a), e a imagem (c) é resultante da aplicação da OCR Tesseract 3 na imagem (a). A versão 3 se mostrou mais eficaz, comparada à versão 2. Após esta etapa são gerados documentos .txt a partir da extração de informação realizada pela OCR.

4.5. Considerações Finais dos Resultados Experimentais

Os experimentos foram realizados a partir de uma base contendo 90 documentos de um órgão público, sendo eles separados em três tipos: Portarias, Atos Executivos e Ordens de Serviços. Foram estudados cada documento, e observou-se a similaridade entre eles, para a definição dos blocos de informações relevantes que eles possuem em comum. Feito isso, foi utilizada a morfologia matemática binária para que as linhas se juntassem formando blocos por toda a imagem.

Após a criação dos blocos, os mesmos foram recortados e aplicada a OCR a cada bloco individualmente, assim foram extraídas as informações textuais sobre o documento, gerando assim um arquivo .txt em forma de texto editável.

A OCR não garantiu o reconhecimento total de todos os caracteres, a ultima linha da Figura 21 da imagem (b), apresenta um problema em relação ao reconhecimento de caracteres da OCR. Onde o caractere (o) é reconhecido como (0), enquanto na imagem (c) o mesmo caractere é reconhecido como (o), o qual é realmente na imagem original.

Os resultados obtidos após todo esse processo, foram satisfatórios, não houve perda de informação ao se fazer o recorte nos blocos, as perdas em relação ao reconhecimento de caracteres foram mínimas, não alterando o texto de maneira a inviabilizar a indexação.

5. CONCLUSÃO

A técnica de reconhecimento automático de *layout* de documentos, estudada e implementada neste trabalho, obteve a extração de todas as informações textuais dos documentos analisados, demonstrando uma solução interessante para ser utilizada em gerenciamentos automáticos de documentos eletrônicos com indexação.

A Morfologia Matemática binária em conjunto com outros filtros para o melhoramento da imagem, foi necessária para a implementação, demonstrando ser uma ótima opção para a técnica de reconhecimento de documentos.

Os testes realizados entre as versões da OCR Tesseract 2, e Tesseract 3, apresentaram uma pequena divergência entre eles, onde a primeira reconheceu alguns caracteres de forma incorreta, enquanto a segunda versão obteve melhores resultados quanto ao reconhecimento.

Para trabalhos futuros, pode ser aplicada a metodologia proposta neste trabalho em documentos antigos, com o auxílio da morfologia matemática em cores. Este trabalho também pode ser adaptado, ao reconhecimento automático de manuscritos, com a ajuda da ICR (*Intelligent Character Recognition*).

REFERÊNCIAS

ALVES, G. M. Método fundamentado em processamento digital de imagens para contagem automática de unidades formadoras de colônias, São Carlos: UFSCar, 2006.

BRITTO JUNIOR, Alceu de Souza et al. Técnicas em Processamento e Análise de Documentos Manuscritos Alceu. RITA, Curitiba, v. 8 , n. 2, p.47-68, out. 2001. Disponível em: <http://www.etsmtl.ca/ETS/media/ImagesETS/Labo/LIVIA/Publications/2001/BrittoRITA.pdf>.Acesso em: 18 nov. 2013.

FACON, Jacques. Processamento e Análise de Imagens. Pontifícia Universidade Católica do Paraná, Curso de Mestrado em Informática Aplicada. Agosto, 2005.Curitiba-PR

FACON, Jacques. Técnicas de Processamento Digital de Imagens Aplicadas à Área da Saúde. ERI 2006 - XIII Escola Regional de Informática da SBC – Paraná.

GOMES, Otávio da Fonseca Martins. Processamento e Análise de Imagens Aplicados à Caracterização Automática de Materiais. 2001. 141 f. Dissertação (Mestrado) - Curso de Ciência de Materiais e Metalurgia, Puc - Rj, Rio de Janeiro - Rj,

GONZALEZ, R. C.; WOODS, R. E. Digital Image Processing. Third Edition, New Jersey: Pearson Education, 2008.

HARALICK, Jaekyu Ha & Robert M.; PHILLIPS, Ihsin T.. Recursive X-Y Cut using Bounding Boxes of Connected Components. Disponível em: <http://www.haralick.org/conferences/71280952.pdf>. Acesso em: 16 out. 2014.

HOLAHAN, C. Google Seeks Help with Recognition. Setembro (2006) Disponível em:<http://www.businessweek.com/technology/content/sep2006/tc20060907_732714.htm?chan=top+news_top+news+index_technology> .

IBGE. Introdução ao processamento digital de imagens. Rio de Janeiro, Rj: IBGE, 2000.

KUBIÇA, Stefano. Metodologia Para Melhoramento De Conteúdos Impressos De Imagens De Documentos Complexos. 2004. 119 f. Dissertação (Mestrado) - Curso de Informática Aplicada, Puc-pr, Curitiba, Pr, 2004. Disponível em: <Metodologia Para Melhoramento De Conteúdos Impressos De Imagens De Documentos Complexos>. Acesso em: 15 nov. 2013.

MARQUES FILHO, Ogê; VIEIRA NETO, Hugo. Processamento Digital de Imagens, Rio de Janeiro: Brasport, 1999.

OTSU, Nobuyuki . "A threshold selection method from gray-level histograms". IEEE Trans. Sys., Man., Cyber. 9 (1): 62–66, 1979.

QUEIROZ, José Eustáquio Rangel de; GOMES, Herman Martins. Introdução ao Processamento Digital de Imagens. UFRGS, Porto Alegre - Rs, 2001.

RICE, S. V.; Jenkins, F. R.; Nartker, T. A. The Fourth Annual Test of OCR Accuracy.Las Vegas: 1995.

SANTIAGO, Diego João Costa. Otimização E Eficiência De Algoritmos De Rotulação De Componentes Conexos Em Imagens Binárias. 2009. 36 f. TCC (Graduação) - Curso de Ciência da Computação, Ufpe, Recife.

SANTOS, Tiago Souza dos. SegmentaçãoFuzzy de Texturas e Vídeos. 2012. 66 f. Dissertação (Mestrado) - Curso de Sistema e Computação, Universidade Federal do Rio Grande do Norte, Natal, Rn, 2012. Disponível em: <http://bdtd.bczm.ufrn.br/tde_arquivos/14/TDE-2013-04-15T152914Z-4999/Publico/TiagoSS_DISSERT.pdf>. Acesso em: 18 nov. 2013.

SGARBI, Ederson Marcos. SEGMENTACÃO DO CONTEUDO E ESTIMATIVA DO FUNDO POR MORFOLOGIA MATEMÁTICA EM COR DA PRIMEIRA BIBLIA DE GUTENBERG. 2014. 240 f. Tese (Doutorado) - Curso de Informática, PUC-Pr, Curitiba-Pr, 2013.

SILVA, Maíra Saboia da. BINARIZAÇÃO DE IMAGENS DE CHEQUES. 2009. 58 f. TCC (Graduação) - Curso de Engenharia da Computação, Universidade de Pernambuco, Recife, Pe, 2009. Disponível em: <http://tcc.ecomp.poli.br/20091/Tcc_MairaSaboia_BinarizacaoImagensCheque_2009.1.pdf>. Acesso em: 20 nov. 2013.

SOUZA, Taciana; CORREIA, Suzete. Anais. In: CONGRESSO DE PESQUISA E INOVAÇÃO DA REDE NORTE NORDESTE DE EDUCAÇÃO TECNOLÓGICA, 2., 2007, João Pessoa, Pb. ESTUDO DE TÉCNICAS DE REALCE DE IMAGENS DIGITAIS E SUAS APLICAÇÕES. João Pessoa, Pb: Connepi, 2007. p. 1 - 4 Disponível em: <http://www.redenet.edu.br/publicacoes/arquivos/20080127_131848_INFO-022.pdf>. Acesso em: 22 nov. 2013.

THOMAS, Rémi. Tessnet2 a .NET 2.0 Open Source OCR assembly using Tesseract engine. Disponível em: <http://www.pixel-technology.com/freeware/tessnet2/>. Acesso em: 22 nov. 2013.

VON WANGENHEIM, Aldo; SANTOS, Cleiton Almeida dos. Morfologia Matemática. Disponível em: <http://www.inf.ufsc.br/~visao/morfologia.pdf>. Acesso em: 15 out. 2014.