

# Optical Character Recognition system development on android platform to aid edit text

D. Alencar Lima, G. B. Holanda, J. W. M. de Souza, P. P. Rebouças Filho

**Abstract**— Texts contained in digital images are usually limited in terms of interaction and editing by users that need to extract this content. Thus, this article proposes an Optical Character Recognition system capable to convert a text contained in the image to an editable text file. For segmentation step, the Connected Contours method were more efficient, with accuracy greater than 99% and processing time less than one second for all possibilities tested. To pattern recognition step, the extractor Central Moments with Multi-Layer Perceptron network were the best combination for the system developed with 99.86% accuracy, 99.93% specificity and testing time in the order of nanosecond.

**Index Terms**— Optical Character Recognition, Digital Image Processing, Computer Vision.

## I. INTRODUÇÃO

GERALMENTE, quando um arquivo é digitalizado ou fotografado através de um dispositivo acoplado a um sistema de captura ótica, o mesmo é armazenado em formato de imagem. Com isto, o usuário objetiva posteriormente fazer anotações, cópias de trechos ou efetuar modificações no corpo do texto.

Entretanto, há uma determinada complexidade para realização dessas simples atividades devido ao fato de que o texto presente no documento em formato de imagem não é editável. O mesmo ocorre com alguns arquivos no formato PDF (Portable Document Format), que não permitem a cópia de texto.

Para auxiliar a resolução destes problemas, várias ferramentas que utilizam tecnologias inovadoras foram desenvolvidas. Usualmente, estas ferramentas utilizam sistemas de Reconhecimento Ótico de Caracteres (OCR, do inglês Optical Character Recognition).

Em [1], Zhu apresenta uma abordagem em que a detecção de texto é realizada usando cluster adaptativo. Neste trabalho, a melhor precisão obtida foi de 83,98% usando IC-DAR 2013.

Diferentes alfabetos são utilizados como base para o desenvolvimento de sistemas OCR. Nos trabalhos de Al-Marakeby [2] e Mousa [3], foram desenvolvidos sistemas de OCR baseados no alfabeto árabe. O sistema proposto por Al-Marakeby [2] apresentou diferentes algoritmos para cada

aplicação, em que o usuário escolhe a técnica aplicada. Mousa [3] propôs um novo algoritmo para o reconhecimento óptico de caracteres árabes usando detectores invariantes á escala, descritores baseados em gradientes e clusterização k-means. Enquanto no trabalho de Pengcheng [4] os caracteres da caligrafia chinesa foram usados como base do OCR proposto. Para isso, foi desenvolvido um novo descritor, denominado GIST-SC, baseado em formas.

No contexto de aplicações móveis é interessante mencionar Ferreira Neto [5], que desenvolveu um sistema que permite, ao usuário com um dispositivo móvel, tirar uma foto e ouvir o texto reconhecido na imagem capturada.

A revisão da literatura demonstra que, grandes partes das pesquisas focam no desenvolvimento de um sistema que trabalha apenas com uma forma de captura da imagem, a qual, na maioria dos casos, é utilizada uma câmera ou escâner. Poucos estudos descrevem sistemas embarcados em dispositivos móveis e pouquíssimas pesquisas realizam testes comparativos entre diferentes técnicas de cada etapa do OCR.

Considerando o que foi apresentado, este trabalho propõe um novo sistema OCR baseado no alfabeto latino e embarcado em plataforma Android, capaz de reconhecer e extrair textos em imagens e converter este conteúdo para um documento editável. Caso o usuário deseje, o texto também é apresentado na forma de áudio. Neste trabalho, foram utilizados dois modos de captura de imagens: um scanner e a câmera de um dispositivo móvel.

Com o intuito de otimizar o sistema desenvolvido, diferentes métodos de segmentação e reconhecimento de padrões foram comparados a fim de aumentar a acurácia e velocidade de cada etapa.

## II. RECONHECIMENTO ÓTICO DE CARACTERES

Sistemas de Reconhecimento Ótico de Caracteres são capazes de realizar a identificação de símbolos gráficos que compõem um alfabeto de qualquer língua, simulando a capacidade humana de ler [6,7]. Estes sistemas iniciam com a aquisição de uma imagem através de equipamentos ópticos acoplados a um dispositivo de captura, geralmente uma câmera ou escâner, gerando uma imagem digital. Técnicas de Processamento Digital de Imagens (PDI) e Reconhecimento

D. de Alencar Lima, Instituto Federal de Ciência e Tecnologia do Ceará (IFCE), Fortaleza, Ceará, Brasil, danielalencar@lapisco.ifce.edu.br

G. B. Holanda, Instituto Federal de Ciência e Tecnologia do Ceará (IFCE), Fortaleza, Ceará, Brasil, gabrielbandeira@lapisco.ifce.edu.br

J. W. Mendes de Souza, Instituto Federal de Ciência e Tecnologia do Ceará (IFCE), Fortaleza, Ceará, Brasil, wellmendes@lapisco.ifce.edu.br

P. P. Rebouças Filho, Instituto Federal de Ciência e Tecnologia do Ceará (IFCE), Maracanaú, Ceará, Brasil, pedrosarf@ifce.edu.br

de Padrões (RP) são aplicadas à imagem em ordem para obter um código ASCII, por exemplo, cada um representando um caractere [8].

Nestes sistemas, a etapa de pré-processamento consiste em retirar falhas e eliminar regiões indesejáveis da mesma, melhorando a qualidade da imagem digital. A segmentação é a etapa responsável por separar da imagem as regiões de interesse [8]. Logo em seguida, a região segmentada é utilizada para extrair atributos que, por sua vez, são utilizados para identificar padrões previamente estabelecidos.

O sistema OCR proposto neste trabalho pode ser dividido em duas etapas: primeiramente, a etapa de PDI, na qual são utilizadas técnicas para filtrar ruídos na imagem, assim como, algoritmos para localização, segmentação e ordenação dos caracteres e símbolos gráficos presentes; e posteriormente, a etapa RP que é responsável pelo reconhecimento de cada caractere e símbolo gráfico, através da utilização de códigos para extrair atributos e algoritmos para classificar padrões.

#### A. Pré-Processamento

Geralmente, em um sistema OCR, a etapa de pré-processamento busca melhorar a qualidade da disposição dos dados a serem segmentados em uma etapa posterior. Portanto, algoritmos são aplicados para eliminar ou reduzir regiões indesejáveis em uma imagem, oriundas da etapa de aquisição. Para isto, normalmente são utilizados os filtros passa-baixa. Este tipo de filtro suaviza a imagem e reduz o número de níveis de cinza e conseqüentemente minimiza o ruído. As altas frequências, que correspondem a transições rápidas, são atenuadas. Para este fim, a técnica utilizada neste trabalho foi o filtro da mediana.

O filtro da mediana substitui o nível de cinza de cada pixel pela mediana dos níveis de cinza dos vizinhos. Este método é particularmente eficaz para atenuar os ruídos, perdendo o mínimo de informação das bordas.

Outro processo muito comum na etapa de pré-processamento é a limiarização. Esta técnica tem como princípio básico determinar um valor como o limiar, geralmente em tom de cinza, e partir deste valor separar a imagem em conjuntos de pixels [8]. As técnicas de limiarização são limitadas pela dependência do valor deste limiar. Vários estudos têm sugerido técnicas para encontrar valores de limiar ótimo [9,10], mas apenas alguns conseguiram resultados satisfatórios em aplicações específicas. Neste artigo foi utilizada a limiarização proposta por Otsu [11]. Este método usa o histograma de aproximação em uma imagem de duas funções gaussianas e seleciona o limiar para mitigar a variância intra-classe.

Para padronizar as informações contidas nas imagens, ângulos irregulares são comumente corrigidos. Para isto, foi utilizada a Transformada de Hough [12]. A Transformada de Hough é uma técnica usada para isolar características de uma forma específica em uma imagem. Neste método os caracteres devem ser representados na forma paramétrica. A Transformada de Hough clássica é comumente usada para a detecção de curvas regulares, como linhas, círculos e elipses.

#### B. Segmentação

De acordo com Gonzales e Woods [8], a segmentação consiste na atribuição de um rótulo a cada pixel da imagem, identificando assim, a qual região este pixel pertence. Para isto, as técnicas de segmentação baseiam-se em duas propriedades básicas: similaridade e descontinuidade. A primeira categoria baseia-se na divisão da imagem em regiões não similares, onde o critério de similaridade é pré-determinado. A segunda separa as regiões da imagem com base em mudanças abruptas nas características dos pixels vizinhos.

Para realização de análises e comparações, foram escolhidos dois métodos de similaridade, Crescimento de Região (CR) e Watershed, e um método baseado na descontinuidade, Contornos Conectados.

O Crescimento de Região é um método tradicional e amplamente relatado na literatura [8]. Esta técnica realiza um agrupamento de subconjuntos ou grupos de pixels na região desejada. Isto é possível através da expansão de uma região que é inicializada por um pixel, chamado semente [13, 14]. Esta expansão ocorre pela agregação dos pixels na região por interações sucessivas na vizinhança. A análise desta técnica é válida, pois é possível estabelecer uma configuração específica para cada aplicação, já que este método depende da regra de associação e do critério de inicialização da semente.

O Watershed é baseado na visualização da imagem 3D, em que as coordenadas da imagem (x,y) e a intensidade do nível de cinza são interpretadas como uma imagem topográfica. Três pontos são considerados: os pontos (pixels) que pertencem a um mínimo regional; os pontos em que a gota de água irá cair em um mínimo único, se colocada em qualquer local destes pontos, chamado barragem; e os pontos de divisão em que é igualmente provável que a água caia, mas não se pode determinar o mínimo [8]. A técnica de Watershed por inundação fornece contornos fechados, o que é bastante útil para a segmentação em imagens digitais. Além disso, os limites regionais sempre correspondem aos contornos como linhas de objetos desobstruídas [15].

O método Contornos Conectados utiliza os contornos dos objetos, previamente identificados por um gradiente detector de bordas. Esses contornos são alocados em um vetor bidimensional que transporta as posições de pixels representando as arestas [16]. A seguir, é realizado um processo de pesquisa, em sentido horário, pelo pixel adjacente conectado até que não haja pixels vizinhos ou o último pixel analisado seja ligado ao pixel inicial [8]. Este método é utilizado para obter mais informações sobre as diferentes regiões em uma imagem, com aplicações específicas para análise e reconhecimento de texturas.

#### C. Extração de Atributos

A etapa de Extração de Atributos busca descrever numericamente as características da imagem. Neste artigo, alguns métodos comumente encontrados na literatura foram utilizados. A base teórica de cada método é mostrada a seguir.

Os momentos estatísticos são extraídos a partir da distribuição dos níveis de cinza, que são comumente

calculados pelo histograma. Estas características proporcionam uma descrição estatística da relação entre os níveis de cinza [15,17].

Os momentos centrais são derivados da redução dos momentos espaciais com o centro de gravidade do objeto. Assim, todos os momentos centrais referem-se ao centro de gravidade do objeto [18,19]. A utilização deste método é comum, pois o mesmo apresenta invariância à translação de objetos da imagem.

Os momentos de Hu, também conhecidos como Momentos Invariantes, são baseados no reconhecimento dos objetos, independentemente da posição, tamanho e orientação dos mesmos na imagem. Essa teoria, proposta por Hu [17], apresenta uma descrição alternativa de figuras geométricas planas a partir dos momentos invariantes bidimensionais [20-22]. Por estes motivos, o método de Hu é um descritor adequado para uso em sistemas de OCR.

Haralick, em [23], propôs a GLCM que vem do inglês Gray-Level Co-occurrence Matrix. Neste trabalho é descrita uma metodologia para descrever texturas baseadas em estatística de segunda ordem, onde características derivadas do cálculo de matrizes de co-ocorrência consistindo na contagem de quantas combinações diferentes de níveis de cinza são colocadas em determinada direção da imagem [23]. As matrizes de co-ocorrência formam a base para a preparação de várias medidas estatísticas conhecidas como descritores de Haralick [24]. A análise de textura tem sido amplamente utilizada em diferentes campos, tais como, reconhecimento de caracteres [25], análise de imagem médica [26], e pesquisa de imagem [27], entre outros.

O operador conhecido do inglês como Local Binary Patterns (LBP) foi originalmente projetado para a descrição da textura. O LBP atribui um rótulo a cada pixel de uma imagem ao limiar a vizinhança  $3 \times 3$  de cada pixel com o valor de pixel central e considerando o resultado como um número binário. Em seguida, o histograma dos rótulos pode ser usado como um descritor de textura. Para ser capaz de lidar com texturas em escalas diferentes, o operador LBP foi posteriormente estendido para usar vizinhanças de diferentes tamanhos [28]. Definir a vizinhança local como um conjunto de pontos de amostragem espaçados em um círculo permite qualquer raio e qualquer número de pontos de amostragem. A interpolação bilinear é usada quando um ponto de amostragem não cai no centro de um pixel. Com isto, temos que o LBP é um poderoso descritor de textura que é invariante à escala de cinza e à rotação.

#### D. Reconhecimento de Padrões

Na etapa de Reconhecimento de Padrões várias abordagens podem ser utilizadas. Assim como na segmentação, foram escolhidos métodos de diferentes abordagens com o intuito de realizar uma avaliação e comparação entre métodos. Com isto, foi escolhido um método baseado na regra do vizinho mais próximo, o KNN, um método que utiliza abordagem estatística, o Bayes, duas redes neurais, MLP e SVM e um classificador de abordagem fundamentada em conceitos de teoria dos grafos, o OPF.

O k-vizinho mais próximo (KNN) é um algoritmo de aprendizagem supervisionada, introduzido por [29]. A lógica básica deste algoritmo é encontrar k amostras mais próximas etiquetadas de amostras não classificadas e usar esses rótulos para classificar a amostra desconhecida. Este método requer pouco esforço computacional na fase de treinamento. Em contraste, o custo de computação para rotular uma amostra desconhecida é relativamente alto. O número de vizinhos é o parâmetro deste método.

A teoria de decisão bayesiana é uma abordagem estatística fundamental para o problema da classificação de padrões. Esta abordagem baseia-se na quantificação do trade-off entre várias decisões de classificação utilizando a probabilidade e os custos que acompanham essas decisões. Faz-se a suposição de que o problema de decisão é colocado em termos probabilísticos e que todos os valores de probabilidade relevantes são conhecidos [30, 31].

A rede neural MLP conhecida no inglês como Multilayer Perceptron, consiste em múltiplas camadas com nós ou neurônios que interagem usando conexões ponderadas. Cada camada está totalmente conectada a próxima. Entre as camadas de entrada e saída, pode haver uma ou mais camadas ocultas [32,33]. Os pesos medem o grau de correlação entre os níveis de atividade dos neurônios que eles conectam. Além disso, um algoritmo de treinamento precisa ser usado para ajustar os pesos [34]. O mais popular, o Backpropagation, é composto de duas fases [33].

A técnica Máquina de Vetor de Suporte (SVM, do inglês Support Vector Machine) é um modelo de aprendizagem supervisionado utilizado para a classificação e análise de regressão. Este método é apoiado pela teoria da aprendizagem estatística e foi desenvolvido por [35]. Esta teoria estabelece uma série de princípios que devem ser seguidos para obter classificadores com boa capacidade de generalização. O nome, Máquina de Vetores de Suporte, enfatiza a importância dos vetores mais próximos para separar a margem na complexidade da SVM. Embora, SVM foi originalmente formulado para resolver problemas binários, existem algumas formulações SVM para problemas multi-classe [36]. No entanto, devido à sua complexidade, outras abordagens, como um versus um, um versus todos, DAGSVM e código de correção de erros são os mais amplamente utilizados.

O algoritmo Floresta de Caminhos Ótimos (OPF, do inglês Optimum Path Forest) com um gráfico completo foi apresentado pela primeira vez por Papa [37, 38], e tem sido amplamente utilizado em várias aplicações [39-41]. Protótipos participam de um processo de competição para outras amostras, oferecendo-lhes caminhos de menor custo em seus respectivos rótulos. No final deste processo, um conjunto de treinamento particionado em árvores de caminho ótimo é obtido, e a união do mesmo se refere a uma Floresta de Caminhos Ótimos.

#### E. Medidas de avaliação do desempenho

Para calcular as métricas de desempenho, é necessário primeiro construir a matriz de confusão e extrair as seguintes informações:

- Verdadeiro Positivo (VP): Detecta a condição quando a condição está presente.

- Verdadeiro Negativo (VN): Não detecta a condição quando a condição está ausente.
- Falso Positivo (FP): Detecta a condição quando a condição está ausente.
- Falso Negativo (FN): Não detecta a condição quando a condição está presente.

As métricas utilizadas neste trabalho foram: Acurácia (Acu), Especificidade (Esp), Sensibilidade (Sen), Predição do Valor Positivo (PVP) e a média harmônica entre Sensibilidade e PVP chamada F-Score representadas pelas equações: 1, 2,3,4, e 5, respectivamente.

$$Acu = \frac{(VP + VN)}{(VP + VN + FP + FN)} \quad (1)$$

$$Esp = \frac{VN}{(VN + FP)} \quad (2)$$

$$Sen = \frac{VP}{(VP + FN)} \quad (3)$$

$$PVP = \frac{VP}{(VP + FP)} \quad (4)$$

$$F - score = 2 * \frac{(Sen * PVP)}{(Sen + PVP)} \quad (5)$$

### III. METODOLOGIA

Nesta seção são descritas todas as etapas do sistema móvel desenvolvido. A Fig. 1 apresenta um resumo do todas as etapas deste sistema.

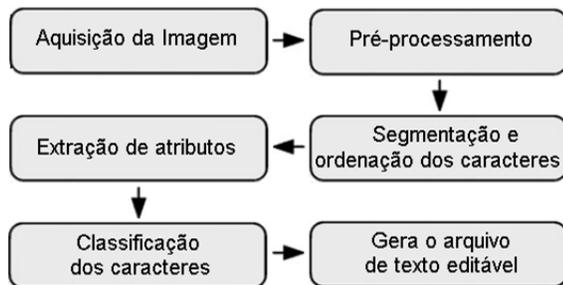


Figura 1. Diagrama de blocos do sistema desenvolvido.

O Sistema desenvolvido possui três etapas, o pré-processamento, segmentação dos caracteres e a etapa de extração e RP, que utiliza os caracteres segmentados. Estas três etapas ocorrem em ciclos contínuos sucessivamente. Na Fig. 2 é apresentado um exemplo de imagem obtida nas etapas de aquisição, pré-processamento e segmentação.

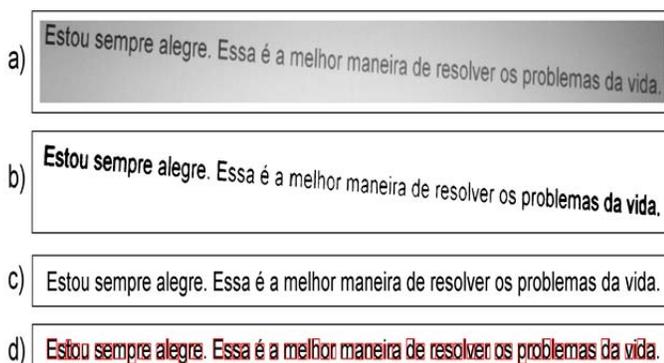


Figura 2. Exemplo de resultado obtido nas etapas de, a) aquisição da imagem em tons de cinza, b) e c) pré-processamento e d) segmentação.

#### A. Pré-Processamento

A etapa de pré-processamento está representada da Fig. 2(a) até 2(c). O primeiro passo do sistema desenvolvido é a captura da imagem, ainda em formato RGB. Em seguida, esta imagem é convertida para tons de cinza, o que aumenta a velocidade de processamento, tornando a ferramenta viável para a aplicação proposta, porque todas as operações são realizadas em somente um canal. Este processo é representado na Fig. 2(a). Posteriormente, foi aplicado o filtro passa-baixa da Mediana na imagem para remover ruídos. Para obter a imagem binária foi utilizada a limiarização de Otsu apresentada na Fig. 2(b).

Na imagem binária foi aplicada a transformada de Hough com o intuito de: i) localizar a folha e o texto. Nesta etapa, o texto foi dividido em linhas, ordenadas com relação ao eixo y da imagem; ii) corrigir o ângulo dos caracteres. A inclinação média de todas as linhas encontradas para girar a imagem foi calculada, corrigindo assim o ângulo de texto. Essa etapa é necessária para facilitar o processo de ordenação dos caracteres, assim como, para minimizar os erros de reconhecimento que podem ocorrer. O resultado da aplicação da transformada de Hough é apresentado na Fig. 2(c).

#### B. Segmentação e ordenação dos caracteres

Na Fig. 2(d) é apresentada a etapa de segmentação. Nesta etapa, para o método Contornos Conectados, inicialmente foi aplicado o gradiente detector de bordas Canny [42]. Entretanto, para os métodos baseados na similaridade, Crescimento de Região e Watershed, nenhum operador gradiente foi aplicado.

Para o método Crescimento de Região a implementação da inicialização das sementes consistiu em varrer a imagem buscando um ponto pertencente à região de interesse. Caso esse ponto seja encontrado, a região inicia o crescimento até que o critério de parada seja satisfeito. Após segmentar o caractere encontrado, o algoritmo continua varrendo a imagem buscando outro ponto que pertença à região de interesse, mas que ainda não foi assimilado a algum caractere.

Para o método Watershed, inicialmente foram definidas as marcas da imagem. Assim, foram identificadas as linhas de watershed pela abordagem de inundação.

No método Contornos Conectados foi utilizada uma matriz de busca de ordem 3. Em sequência, foi aplicada uma aproximação poligonal para melhor definir os contornos segmentados.

Ao concluir a etapa de segmentação, todos os caracteres são ordenados utilizando o algoritmo quicksort. Vale ressaltar que como o texto já está previamente ordenado com relação ao eixo y (ordenação das linhas feita pela transformada de Hough), o algoritmo quicksort necessitou apenas ordenar os caracteres com relação ao eixo x (ordenação dos caracteres em cada linha).

#### C. Extração de Atributos e Reconhecimento de Padrões

Após todos os caracteres serem devidamente segmentados e ordenados, foram aplicados os métodos de extração de atributos. Na Tabela I são apresentados todos os métodos

utilizados nesta etapa, assim como, o número de atributos gerados por cada método.

Após a etapa de extração de atributos foram aplicados os métodos de reconhecimento de padrões. Nesta etapa, foram utilizados cinco métodos de classificação supervisionada: KNN, Bayes, MLP, SVM e OPF.

TABELA I. NÚMERO DE ATRIBUTOS GERADOS POR CADA MÉTODO.

Métodos de Extração de Atributos	Nº de Atributos
LBP	48
GLCM	14
Momentos de Hu	7
Momentos Centrais	7
Momentos Estatísticos	10

Com o intuito de corrigir qualquer erro no reconhecimento dos caracteres, assim como, adicionar os acentos que não foram segmentados, a ferramenta desenvolvida utiliza um dicionário do sistema Android.

#### IV. RESULTADOS

Nesta seção, são apresentados os resultados obtidos nas etapas de segmentação e RP. Para melhor compreensão, cada etapa é avaliada separadamente analisando a precisão e o tempo de processamento de cada técnica. Para esta análise foi utilizado um banco composto por 100 imagens capturadas pela câmera de um smartphone com resolução de 8 MP e 100 imagens com resolução de 600 DPI capturadas por uma scanner HP PRO 8600. Os testes foram realizados em um computador com processador Intel Core i5 de 2,4 GHz, 8 GB de RAM e sistema operacional Mac X El Captain 10.11.2.

Foram utilizados 10 textos com fonte Arial 12. Cada texto possuía em média 674 caracteres. Com o intuito de obter resultados mais confiáveis, foram capturadas 10 imagens de cada texto variando a angulação e a distância entre o texto e o dispositivo de captura, assim como a luminosidade do ambiente.

Para a etapa de extração de atributos, aproximadamente 100.000 caracteres em fonte Arial foram utilizados. Na Tabela II são apresentados todos os caracteres adotados para o sistema proposto.

TABELA II. CARACTERES ADOTADOS.

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z	a	B	C	d	e	f	g
h	i	j	k	l	M	N	o	p	q	r
s	t	u	v	w	Y	z	0	1	2	3
4	5	6	7	8	9	/	!	?	<	>

##### A. Resultados da etapa de segmentação

A eficácia dos métodos de segmentação foi analisada usando as métricas acurácia e f-score. Os resultados obtidos pelos métodos Contornos Conectados, Crescimento de Região e Watershed são apresentados nas Fig. 3 e 4 para imagens fotografadas e nas Fig. 5 e 6 para imagens escaneadas.

Pela análise das Fig. 3, 4, 5 e 6 verifica-se que o método Contornos Conectados apresentou melhor desempenho para segmentar os caracteres neste OCR em ambas as formas de

aquisição de imagem. O método Watershed teve resultados próximos ao melhor método. Já o Crescimento de Região demonstrou ser menos eficiente na segmentação dos caracteres, embora apresente resultados satisfatórios sempre superior a 90%. A inferioridade deste método se deve principalmente a alta sensibilidade de agregação para variações nos valores dos pixels.

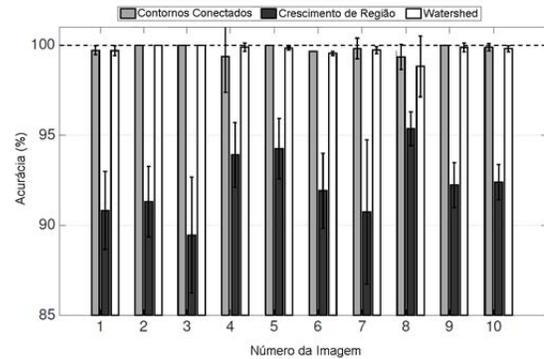


Figura 3. Acurácia obtida usando os métodos de segmentação Contornos Conectados, Watershed e Crescimento de Região para as imagens fotografadas.

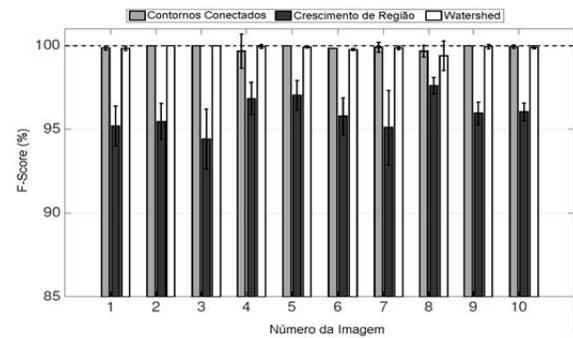


Figura 4. F-score obtido usando os métodos de segmentação Contornos Conectados, Watershed e Crescimento de Região para as imagens fotografadas.

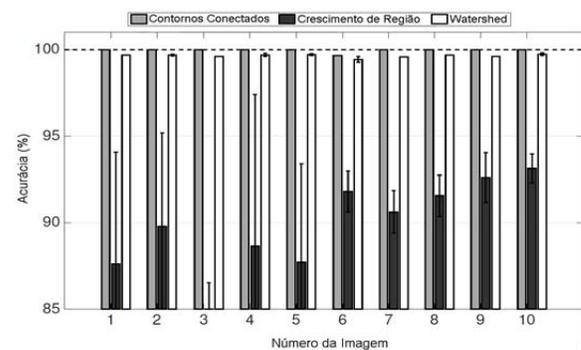


Figura 5. Acurácia obtida usando os métodos de segmentação Contornos Conectados, Watershed e Crescimento de Região para as imagens escaneadas.

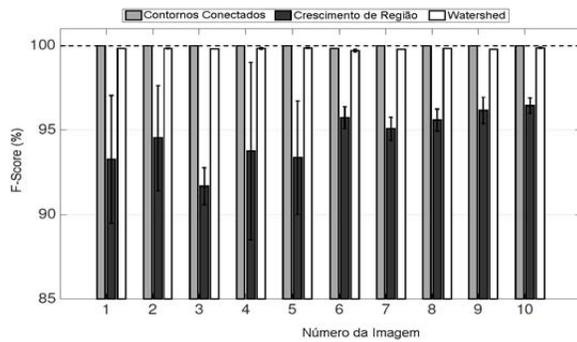


Figura 6. F-score obtido usando os métodos de segmentação Contornos Conectados, Watershed e Crescimento de Região para as imagens fotografadas.

Com relação à forma de captura, observa-se que as imagens escaneadas tiveram melhores resultados do que as fotografadas. Esta superioridade ocorre, pois as imagens escaneadas possuem maior resolução e também não sofrem influência pelo ambiente e/ou quaisquer erro de manuseio.

Em aplicações embarcadas, o tempo de processamento é um elemento bastante crítico. Com isto, foram analisados os tempos de segmentação para cada método. Na Tabela III é apresentada a média, assim como, o desvio padrão dos tempos de processamento de cada método de segmentação para as imagens fotografadas e escaneadas.

TABELA III. TEMPO DE SEGMENTAÇÃO UTILIZANDO OS MÉTODOS CONTORNOS CONECTADOS (CC), WATERSHED (WA) E CRESCIMENTO DE REGIÃO (CR).

Imagens Fotografadas			
Texto	CC (s)	CR (s)	Wa (s)
1	0.527±0.030	10.200±0.875	0.781±0.042
2	0.491±0.031	10.850±1.121	0.771±0.013
3	0.437±0.013	7.954±0.646	0.680±0.019
4	0.481±0.038	7.471±0.867	0.678±0.028
5	0.518±0.018	9.843±1.282	0.746±0.040
6	0.477±0.018	8.574±0.942	0.727±0.032
7	0.409±0.012	5.817±1.143	0.617±0.044
8	0.488±0.055	7.568±0.785	0.685±0.040
9	0.439±0.023	7.397±1.100	0.682±0.037
10	0.508±0.012	9.629±0.584	0.767±0.019
<b>Média</b>	<b>0,477±0,025</b>	<b>8,529±0,935</b>	<b>0,713±0,031</b>
Imagens Escaneadas			
Texto	CC (s)	CR (s)	Wa (s)
1	0.440±0.016	13.042±1.212	0.905±0.168
2	0.419±0.013	12.579±1.235	0.849±0.121
3	0.355±0.010	9.019±0.303	0.777±0.064
4	0.372±0.011	9.487±1.010	0.699±0.028
5	0.422±0.013	13.291±1.262	0.809±0.072
6	0.404±0.010	10.704±0.468	0.765±0.043
7	0.358±0.008	7.070±0.257	0.652±0.019
8	0.403±0.021	9.179±2.167	0.850±0.147
9	0.350±0.016	8.005±1.743	0.678±0.045
10	0.430±0.005	10.496±0.339	0.787±0.050
<b>Média</b>	<b>0,395±0,012</b>	<b>10,2870±0,999</b>	<b>0,777±0,076</b>

O método Contornos Conectados apresentou a maior velocidade de segmentação, seguido pelo Watershed, conforme mostrado na Tabela III.

A análise da etapa de segmentação mostrou que o método Contornos Conectados foi mais eficiente, pois apresentou maior taxa de acurácia e f-score, assim como, menor tempo exigido para segmentar todos os caracteres.

### B. Resultados da etapa de Reconhecimento de Padrões

O processo de validação cruzada envolve particionar uma amostra do banco de dados gerado pela extração de atributo sem subconjuntos complementares, realizar uma análise em um subconjunto (denominado conjunto de treinamento) e validar a análise aos outros subconjuntos (chamados grupos de validação ou conjuntos de teste). Para reduzir a variabilidade, muitas iterações de validação cruzada são executadas usando diferentes partições.

No que diz respeito à definição da amostra de treino e teste, foi utilizado um processo de validação cruzada, Hold Out, com 50% para o treinamento e 50% para os conjuntos de teste, sendo o mesmo executado 10 vezes.

Para a classificação com o método KNN foi considerado os três vizinhos mais próximos. O SVM foi ajustado para o kernel RBF. A distância euclidiana foi utilizada para o método OPF. O classificador Bayes foi validado usando a distribuição de probabilidade Gaussiana Normal. Cada uma destas técnicas foi testada dez vezes.

Diferentes topologias foram utilizadas para cada extrator na configuração da rede MLP. Estas topologias foram avaliadas com uma configuração utilizando de 5 a 300 neurônios na camada oculta. Com isto, temos que foram utilizados 80, 35, 25, 250 e 90 neurônios na camada oculta para os extratores GLCM, LBP, Momentos de Hu, Momentos Estatísticos e Momentos Centrais, respectivamente.

Na Tabela IV é apresentado o resumo dos resultados dos tempos de treinamento e teste, a especificidade e acurácia obtidas por cada método considerando as configurações utilizadas.

TABELA IV. ACURÁCIA (ACU), ESPECIFICIDADE (ESP), TEMPO DE TREINAMENTO (TTR) E TEMPO DE TESTE (TTS) DA ETAPA DE RECONHECIMENTO DE PADRÕES.

GLCM			
Método TTr	TTr(μs)	Esp(%)	Acu(%)
Bayes	35.73±2.320(ms)	181.47±13.975	99.16±0.02
KNN	0.13±0.071(ms)	221.74±8.472	99.32±0.00
MLP	0.10±0.001(ks)	0.01±0.001	98.99±0.01
OPF	0.60±0.002(s)	171.02±4.190	99.24±0.00
SVM	82.95±2.410(s)	218.25±38.713	99.38±0.00
<b>Método TTr(sec)</b>	<b>TTr(μs)</b>	<b>Esp(%)</b>	<b>Acu(%)</b>
Bayes	170.83±14.942(ms)	383.30±17.430	99.08±0.04
KNN	0.24±0.021(ms)	400.13±18.460	99.81±0.00
MLP	0.13±0.040(ks)	0.010±0.001	99.37±0.17
OPF	1.95±0.080(s)	441.38±13.350	99.80±0.00
SVM	169±1.571(s)	320.06±26.380	99.88±0.00
Momentos de Hu			
Método TTr	TTr(μs)	Esp(%)	Acu(%)
Bayes	1.38±0.001(ms)	128.89±20.490	99.78±0.02
KNN	0.784±0.002(ms)	215.69±20.310	99.67±0.00
MLP	0.10±0.029(ks)	0.01±0.001	99.36±0.37
OPF	0.53±0.085(s)	0.20±23.560	99.69±0.00
SVM	76.53±1.925(s)	237.52±10.790	99.57±0.01
Momentos Centrais			

Método TTr	TTs( $\mu$ s)	Esp(%)	Acu(%)
Bayes	1.40 $\pm$ 0.183(ms)	149.91 $\pm$ 10.340	99.92 $\pm$ 0.00
KNN	0.17 $\pm$ 0.058(ms)	250.71 $\pm$ 24.390	99.88 $\pm$ 0.00
MLP	0.26 $\pm$ 0.147(ks)	0.01 $\pm$ 0.001	99.93 $\pm$ 0.03
OPF	0.45 $\pm$ 0.008(s)	90.66 $\pm$ 3.720	99.88 $\pm$ 0.00
SVM	79.71 $\pm$ 6.293(s)	251.50 $\pm$ 89.960	99.92 $\pm$ 0.00
Momentos Estatísticos			
Método TTr	TTs( $\mu$ s)	Esp(%)	Acu(%)
Bayes	2.64 $\pm$ 0.001(ms)	129.83 $\pm$ 12.260	99.95 $\pm$ 0.00
KNN	0.11 $\pm$ 0.001(ms)	252.47 $\pm$ 29.030	99.78 $\pm$ 0.00
MLP	0.72 $\pm$ 0.219(ks)	0.01 $\pm$ 0.001	99.52 $\pm$ 0.06
OPF	0.55 $\pm$ 0.006(s)	126.34 $\pm$ 2.220	99.77 $\pm$ 0.00
SVM	79.45 $\pm$ 2.204(s)	204.36 $\pm$ 27.940	99.88 $\pm$ 0.00

A Tabela IV mostra que alguns métodos destacam-se pelo baixo tempo de treinamento, tais como o KNN e o Bayes. No entanto, o tempo de teste é mais relevante, uma vez que este é um sistema embarcado. Neste caso, o método MLP foi o mais rápido e teve o tempo de teste na ordem de  $\eta$ s.

Com relação a análise da forma de extração, pela Tabela IV observa-se que a classificação utilizando os dados extraídos pelos métodos baseados na forma do objeto, os momentos, obtiveram melhores resultados se comparados aos mesmos classificadores utilizando os dados previamente extraídos pelos métodos baseados na textura, GLCM e LBP.

No entanto, Todos os classificadores apresentaram taxas de acurácia e especificidade superiores a 98%, e também permaneceram estáveis com baixos valores de desvios-padrão.

TABELA V. CARACTERES COM PIORES TAXAS DE RECONHECIMENTO PARA OS EXTRATORES GLCM, LBP E MOMENTOS DE HU, CENTRAIS E ESTATÍSTICOS.

GLCM		
Classe	Acurácia	Especificidade
i	97,52 $\pm$ 0.12	97,61 $\pm$ 0.13
!	97,60 $\pm$ 0.07	97,70 $\pm$ 0.02
<	97,86 $\pm$ 0.05	97,89 $\pm$ 0.02
t	97,99 $\pm$ 0.00	97,96 $\pm$ 0.01
j	98,03 $\pm$ 0.02	98,05 $\pm$ 0.01
k	98,10 $\pm$ 0.02	98,15 $\pm$ 0.00
LBP		
Classe	Acurácia	Especificidade
t	97,61 $\pm$ 0.12	97,64 $\pm$ 0.10
l	97,68 $\pm$ 0.07	97,65 $\pm$ 0.05
i	97,79 $\pm$ 0.05	97,80 $\pm$ 0.06
l	97,95 $\pm$ 0.05	97,98 $\pm$ 0.01
l	98,06 $\pm$ 0.02	98,10 $\pm$ 0.01
7	98,13 $\pm$ 0.01	98,16 $\pm$ 0.00
Momentos de Hu		
Classe	Acurácia	Especificidade
!	98,48 $\pm$ 0.09	98,55 $\pm$ 0.02
l	98,53 $\pm$ 0.04	98,54 $\pm$ 0.01
>	98,62 $\pm$ 0.03	98,66 $\pm$ 0.00
t	98,69 $\pm$ 0.00	98,65 $\pm$ 0.01
i	98,91 $\pm$ 0.01	99,01 $\pm$ 0.00
/	99,00 $\pm$ 0.00	99,11 $\pm$ 0.00
Momentos Centrais		
Classe	Acurácia	Especificidade
i	98,55 $\pm$ 0.04	98,61 $\pm$ 0.03
t	98,69 $\pm$ 0.04	98,70 $\pm$ 0.04
f	98,85 $\pm$ 0.02	98,89 $\pm$ 0.02
!	98,96 $\pm$ 0.01	99,96 $\pm$ 0.00

g	99,03 $\pm$ 0.00	99,05 $\pm$ 0.01
?	99,14 $\pm$ 0.00	99,15 $\pm$ 0.00
Momentos Estatísticos		
Classe	Acurácia	Especificidade
l	98,52 $\pm$ 0.05	98,57 $\pm$ 0.03
l	98,60 $\pm$ 0.01	98,66 $\pm$ 0.02
l	98,86 $\pm$ 0.02	98,88 $\pm$ 0.03
/	98,99 $\pm$ 0.00	98,94 $\pm$ 0.00
!	99,02 $\pm$ 0.00	99,05 $\pm$ 0.00
t	99,11 $\pm$ 0.01	99,12 $\pm$ 0.00

Grande parte dos erros obtidos na etapa de reconhecimento de padrões se deve ao fato do extrator não gerar uma descrição suficientemente diferente entre caracteres semelhantes, de forma que o classificador não seja capaz de reconhecer aquele caractere corretamente. Assim, foi realizada uma análise por classe. Para isto, foi calculada a acurácia e especificidade média obtida pelos classificadores e assim determinar a eficiência do reconhecimento a partir das bases de dados geradas pelos cinco métodos de extração usados neste artigo. A Tabela V apresenta os valores dessas métricas relativos aos seis caracteres com piores taxas de reconhecimento para os extratores GLCM, LBP e Momentos de Hu, Centrais e Estatísticos, respectivamente.

Analisando a Tabela V observa-se que os caracteres i, j, l, ! e t, assim como era esperado, geralmente apresentam as menores taxas de acurácia e especificidade, uma vez que estes caracteres são bastante semelhantes tanto em forma como em textura. Embora exista essa dificuldade, vale ressaltar que as taxas de reconhecimento são superiores a 97,5% para todos os caracteres adotados neste OCR.

No geral, a combinação do extrator Momentos Centrais com a rede MLP e Momentos Estatísticos com o classificador bayesiano mostraram o melhor resultado para as métricas calculadas, pois possuem alta acurácia e especificidade com baixa flutuação estatística.

Uma análise completa da etapa de reconhecimento de padrões mostra que extrator Momentos Centrais com a rede MLP apresentou os melhores resultados, pois tem o menor tempo de teste e atingiu taxas de acurácia e especificidade acima de 99%. Assim, demonstrando ser a melhor combinação para o sistema proposto, pois demonstra eficiência de acordo com a análise dos tempos em conjunto com as métricas utilizadas.

## V. CONCLUSÃO

A principal contribuição deste trabalho é o desenvolvimento de um eficiente sistema OCR embarcado em plataforma Android para auxiliar as pessoas em atividades diárias que necessitem editar textos presentes em imagens. Neste sistema, o usuário escolhe o método de captura de imagem e a forma de reprodução do texto extraído (áudio ou arquivo de texto editável). Finalmente, há a vantagem do baixo tempo de processamento deste sistema, juntamente com as altas taxas de acurácia e especificidade.

A taxa de acerto na etapa de reconhecimento é diretamente proporcional à qualidade dos caracteres de entrada. Quando os caracteres segmentados obedecem ao padrão de treinamento, o reconhecimento se aproxima de 100%. Assim, o treinamento

foi feito com textos obtidos em diferentes circunstâncias para aumentar as taxas de sucesso em condições adversas.

O método de segmentação Contornos Conectados mostrou os melhores resultados neste OCR, sempre com acurácia superior a 99% e tempo de processamento abaixo de 1s, ao contrário do método Crescimento de Região que teve muitas dificuldades para segmentar os caracteres, apresentando acurácia média de 92% e tempo de processamento na ordem de 10s. Esta dificuldade gerada pelo método de inicialização da semente resultou em agregações entre caracteres ou fragmentação de um único caractere em mais de uma região, afetando assim significativamente a qualidade da segmentação. Na etapa de reconhecimento de padrões, a rede MLP em conjunto com o extrator Momentos Centrais revelou-se muito eficaz, uma vez que foram obtidas altas taxas de acurácia e especificidade (acima de 99%) e tempo de teste na ordem de ns.

As principais dificuldades estão relacionadas com a qualidade da imagem de entrada. Pois, imagens com má qualidade aumentam os erros no estágio de segmentação, ocasionando a ocorrência de caracteres aglutinados ou fragmentados. No entanto, esses erros são corrigidos com uma etapa de pós-processamento, quando os caracteres aglutinados são separados e os caracteres fragmentados são corrigidos.

Após todos os caracteres serem segmentados e reconhecidos, é utilizado um dicionário do sistema Android capaz de corrigir todos os caracteres que foram erroneamente classificados, assim como, adicionar os caracteres que não foram adotados neste sistema, como por exemplo, os acentos.

O software final desenvolvido utilizou a melhor técnica de cada etapa. Para a segmentação, foi utilizado o método Contornos Conectados; Para a etapa de extração de atributos, o extrator Momentos Centrais; E para o reconhecimento de padrões, a rede MLP.

Com o objetivo de aumentar a flexibilidade deste sistema estuda-se em trabalhos futuros importar esta aplicação para outros sistemas operacionais, como iOS e Windows Phone. Pretende-se também expandir este trabalho para fontes com outros tipos de letras e diferentes tipos de textos impressos, tais como jornais e revistas.

#### REFERÊNCIAS

- [1] [1] Y. Zhu, C. Yao and X. Bai, "Scene text detection and recognition: re- cent advances and future trends", *Frontiers of Computer Science*, vol. 10, pp. 19-36, 2015.
- [2] [2] A. Al-Marakeby, F. Kimura, M. Zaki and A. Rashid, "Design of an embedded Arabic optical character recognition", *Journal of Signal Processing Systems*, vol.70, pp.249-258, 2012.
- [3] [3] M. A. A. Mousa, M. S. Sayed, M. I. Abdalla, "An efficient algorithm for arabic optical font recognition using scale-invariant detector", *International Journal on Document Analysis and Recognition*, vol. 18, pp. 263-270, 2015.
- [4] [4] G. Pengcheng, W. Jiangqin, L. Yuan, X. Yang and M. Tianjiao, "Fast chinese calligraphic character recognition with large-scale data", *Multimedia Tools and Applications*, vol. 74, pp. 7221-7238, 2014.
- [5] [5] R. Ferreira Neto and N. Fonseca, *Camera reading for blind people*, *Procedia Technology*, vol. 16, pp. 1200-1209, 2014.
- [6] [6] E. Cavalcanti Neto, E. S. Rebouças, J. L. de Moraes, S. L. Gomes and P. P. Rebouças Filho, *Development control parking access using techniques digital image processing and applied computational intelligence*, *IEEE Latin America Transactions*, vol. 13, pp. 272-276, 2015.
- [7] [7] E. Cavalcanti Neto, S. L. Gomes, P. P. Rebouças Filho and V. H. C. Albuquerque, *Brazilian vehicle identification using a new embedded plate recognition system*, *Measurement*, vol. 70, pp. 36-46, 2015.
- [8] [8] R. C. Gonzalez and R. E. Woods, *Digital image processing*, 3rd ed. Pearson Prentice Hall, 2008.
- [9] [9] J. Bernsen, *Dynamic thresholding of gray-level images*, *International Conference of Pattern Recognition*, pp. 1251-1255, 1986.
- [10] [10] G. Johannsen, J. Bille, *A threshold selection method using information measures*, *International Conference of Pattern Recognition*, vol. 82, pp. 140-143, 1982.
- [11] [11] N. Otsu, *A threshold selection method from gray-level histograms*, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, pp. 62-66, 1979.
- [12] [12] P. V. C. Hough, *Method and means for recognizing complex patterns*, *US Patent 3,069,654*, 1962.
- [13] [13] F. D. L. Moreira, M. N. Kleinberg, H. F. Arruda, F. N. C. Freitas, M. M. V. Parente, V. H. C. de Albuquerque and P. P. Rebouças Filho, "A novel vickers hardness measurement technique based on adaptive balloon active contour method", *Expert Systems with Applications*, vol. 45, pp. 294-306, 2016.
- [14] [14] E. Souza Rebouças, R. Moura Sarmiento and P. Rebouças Filho, "3D Adaptive balloon active contour: method of segmentation of structures in three dimensions", *IEEE Latin America Transactions*, vol. 13, pp. 195-203, 2015.
- [15] [15] W. K. Pratt, *Digital Image Processing*, 4th ed. John Wiley & Sons, Inc., 2007.
- [16] [16] S. L. Gomes, P. P. Rebouças Filho, et al, "Computer vision system to aid drivers of vehicles through vertical signaling traffic", *International Journal of Computer Applications*, vol. 115, pp. 1-7, 2015.
- [17] [17] M. K. Hu, "Visual pattern recognition by moment invariants", *IEEE Transactions on Information Theory*, vol. 8, pp. 179-187, 1962.
- [18] [18] A. Paul, N. Bhattacharya and A. S. Chowdhury, "Digit recognition from pressure sensor data using euler number and central moments", *2012 International Conference on Communications, Devices and Intelligent Systems*, pp. 93-96, 2012.
- [19] [19] T. Czachorski, S. Kozielski and U. Stanczyk, *Man-Machine Interactions 2, Advances in Intelligent and Soft Computing*, Springer Berlin Heidelberg, 2011.
- [20] [20] J. Flusser, T. Suk, J. Boldys and B. Zitova, "Projection operators and moment invariants to image blurring", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, pp. 786-802, 2015.
- [21] [21] G. Licciardi, A. Villa, M. Dalla Mura, L. Bruzzone, J. Chanussot and J. Benediktsson, *Retrieval of the height of buildings from worldview-2 multi-angular imagery using attribute filters and geometric invariant moments*, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, pp. 71-79, 2012.
- [22] [22] J. Yang, N. Xiong, A. Vasilakos, Z. Fang, D. Park, X. Xu, S. Yoon, S. Xie and Y. Yang, "A fingerprint recognition scheme based on assembling invariant moments for cloud computing communications", *IEEE Systems*, vol. 5, pp. 574-583, 2011.
- [23] [23] R. Haralick, "Statistical and structural approaches to texture", *Proceedings of IEEE*, vol. 67, pp. 786-804, 1979.
- [24] [24] R. M. Haralick, K. Shanmugam and I. Dinstein, "Textural features for image classification", *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-3, pp. 610-621, 1973.
- [25] [25] M. Naeimzaghiani, S. N. H. S. Abdullah, B. Bataineh and F. PirahanSiah, "Character recognition based on global feature extraction", *2011 International Conference on Electrical Engineering and Informatics*, pp. 1-4, 2011.
- [26] [26] S. Beura, B. Majhi and R. Dash, "Mammogram classification using two dimensional discrete wavelet transform and gray-level co-occurrence matrix for detection of breast cancer", *Neurocomputing*, vol. 154, pp. 1-14, 2015.
- [27] [27] S. Chu, L. Hong, C. Liu and J. Chen, "A new regional shape index for classification of high resolution remote sensing images", *2014 3rd International Workshop on Earth Observation and Remote Sensing Applications*, pp. 156-160, 2014.

- [28] [28]T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, pp. 971-987, 2002.
- [29] [29] D. W. Aha, D. Kibler and M. K. Albert, "Instance-based learning algorithms", Machine Learning, vol. 6, pp. 37-66, 1991.
- [30] [30]R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification, 2nd ed. Wiley-Interscience, 2000.
- [31] [31]S. Theodoridis and K. Koutroumbas, Pattern Recognition, 4th ed. Academic Press, 2008.
- [32] [32]D. W. Ruck, S. K. Rogers, M. Kabrisky, M. E. Oxley and B. W. Suter, "The multilayer perceptron as an approximation to a bayes optimal discriminant function", IEEE Transactions on Neural Network, vol. 1, pp. 296-298, 1990.
- [33] [33]S. O. Haykin, Neural Networks and Learning Machines. Pearson Prentice Hall, 2008.
- [34] [34] C. Cortes, V. Vapnik, "Support-vector networks", Machine Learning, vol. 20, pp. 273-297, 1995.
- [35] [35] V. N. Vapnik, Statistical Learning Theory. John Wiley & Sons, Inc., 1998.
- [36] [36] K. Crammer, Y. Singer, N. Cristianini, J. Shawe-taylor and B. Williamson, "On the algorithmic implementation of multiclass kernel-based vector machines", Journal of Machine Learning Research, vol. 2, 2001.
- [37] [37] J. P. Papa, A. X. Falcão and C. T. N. Suzuki, "Supervised pattern classification based on optimum-path forest", International Journal of Imaging Systems and Technology, vol. 19, pp. 120-131, 2009.
- [38] [38] J. P. Papa, A. X. Falcão, V. H. C. Albuquerque and J. M. R. S. Tavares, "Efficient supervised optimum-path forest classification for large datasets", Pattern Recognition, vol. 45, pp. 512-520, 2012.
- [39] [39] T. M. Nunes, A. L. V. Coelho, C. A. M. Lima, J. P. Papa and V. H. C. de Albuquerque, "EEG signal classification for epilepsy diagnosis via optimum path forest – a systematic assessment, Neurocomputing, vol. 136, pp. 103-123, 2014.
- [40] [40] A.S. Iwashita, J.P. Papa, A.N. Souza, A. X. Falcão, R. A. Lotufo, V. M. Oliveira, V. H. C. de Albuquerque and J. M. R. S. Tavares, "A path- and label-cost propagation approach to speedup the training of the optimum-path forest classifier", Pattern Recognition Letters, vol. 40, pp. 121-127, 2014.
- [41] [41] V.H.C. Albuquerque, C. V. Barbosa, C. C. Silva, E. P. Moura, P. P. Rebouças Filho, J. P. Papa and J. M. R. S. Tavares, "Ultrasonic sensor signals and optimum path forest classifier for the microstructural characterization of thermally aged Inconel 625 alloy", Sensors, vol. 15, pp. 12474-12497, 2015.
- [42] [42] J. Canny, "A computational approach to edge detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, pp. 679-698, 1986.

interest are biomedical signal processing and pattern recognition.



**Pedro Pedrosa Rebouças Filho**, received the PhD degree in Teleinformatics Engineering from Universidade Federal do Ceará, Brazil, in 2013, and is a professor at Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Brazil. His current research interest is applications in Computational Vision. For more information: <http://professorpedrosa.com>



**Daniel de Alencar Lima**, Mechanical Engineering student at the Instituto Federal de Ciência, Educação e Tecnologia do Ceará, Maracanaú, Ceara, Brazil. His current research's interest are applications in Computational Vision and Pattern Recognition.



**Gabriel Bandeira Holanda**, Mechanical Engineering student at the Instituto Federal de Ciência, Educação e Tecnologia do Ceará, Maracanaú, Ceara, Brazil. His current research's interest are applications of mobile devices.



**João Wellington Mendes Souza**, Computer Engineering student at the Instituto Federal de Ciência, Educação e Tecnologia do Ceará, Fortaleza, Ceara, Brazil. His current research