

**MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO**

**Cap BRUNO AVELINO DE ARAUJO OLIVEIRA
JAYME BOARIN DE MAGALHÃES ALVIM**

**DESENVOLVIMENTO DE ALGORITMOS DE SEGMENTAÇÃO DE
IMAGENS EM AMBIENTE ANDROID**

**Rio de Janeiro
2017**

INSTITUTO MILITAR DE ENGENHARIA

**Cap BRUNO AVELINO DE ARAUJO OLIVEIRA
JAYME BOARIN DE MAGALHÃES ALVIM**

**DESENVOLVIMENTO DE ALGORITMOS DE
SEGMENTAÇÃO DE IMAGENS EM AMBIENTE ANDROID**

Projeto de Fim de Curso apresentado ao Curso de Graduação em Engenharia de Computação do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Engenheiro de Computação.

Orientadora: Prof^ª. Carla Liberal Pagliari - Ph.D.
Co-Orientador: Prof. Marcelo de Mello Perez - Ph.D.

Rio de Janeiro
2017

INSTITUTO MILITAR DE ENGENHARIA
Praça General Tibúrcio, 80 - Praia Vermelha
Rio de Janeiro - RJ CEP 22290-270

Este exemplar é de propriedade do Instituto Militar de Engenharia, que poderá incluí-lo em base de dados, armazenar em computador, microfilmar ou adotar qualquer forma de arquivamento.

É permitida a menção, reprodução parcial ou integral e a transmissão entre bibliotecas deste trabalho, sem modificação de seu texto, em qualquer meio que esteja ou venha a ser fixado, para pesquisa acadêmica, comentários e citações, desde que sem finalidade comercial e que seja feita a referência bibliográfica completa.

Os conceitos expressos neste trabalho são de responsabilidade do(s) autor(es) e do(s) orientador(es).

Oliveira, Bruno Avelino de Araujo

Desenvolvimento de Algoritmos de Segmentação de Imagens em Ambiente Android / Bruno Avelino de Araujo Oliveira, Jayme Boarin de Magalhães Alvim, orientado por Carla Liberal Pagliari e Marcelo de Mello Perez - Rio de Janeiro: Instituto Militar de Engenharia, 2017.

24p.: il.

Projeto de Fim de Curso (graduação) - Instituto Militar de Engenharia, Rio de Janeiro, 2017.

1. Curso de Graduação em Engenharia de Computação - projeto de fim de curso. 1. Segmentação. 2. Imagem. 3. Visão Computacional. 4. Algoritmos. I. Pagliari, Carla Liberal. II. Perez, Marcelo de Mello. III. Título. IV. Instituto Militar de Engenharia.

INSTITUTO MILITAR DE ENGENHARIA

**Cap BRUNO AVELINO DE ARAUJO OLIVEIRA
JAYME BOARIN DE MAGALHÃES ALVIM**

**DESENVOLVIMENTO DE ALGORITMOS DE
SEGMENTAÇÃO DE IMAGENS EM AMBIENTE ANDROID**

Projeto de Fim de Curso apresentado ao Curso de Graduação em Engenharia de Computação do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Engenheiro de Computação.

Orientadora: Prof^a. Carla Liberal Pagliari - Ph.D.

Co-Orientador: Prof. Marcelo de Mello Perez - Ph.D.

Aprovado em 18 de Maio de 2017 pela seguinte Banca Examinadora:

Prof^a. Carla Liberal Pagliari - Ph.D. do IME - Presidente

Prof. Marcelo de Mello Perez - Ph.D. do IME

Prof. Anderson Fernandes Pereira dos Santos - D.Sc. do IME

Paulo Roberto Rosa Lopes Nunes - Ph.D. do IME

Rio de Janeiro
2017

Dedico aos meus familiares, amigos, minha esposa Mayra e professores do Instituto Militar de Engenharia, pois me deram forças e foram fundamentais para que eu pudesse vencer os obstáculos diários. Dedico aos meus pais e familiares pelo amor incondicional e pelos valores e ensinamentos passados que me fazem crescer e evoluir constantemente.

AGRADECIMENTOS

Agradeço aos que estiveram comigo durante essa árdua jornada de estudos e me auxiliaram nessa etapa de desenvolvimento profissional. Um muito obrigado aos familiares, amigos e mestres.

Um obrigado especial aos Professores Orientadores Ph.D. Carla Liberal Pagliari e Ph.D. Marcelo de Mello Perez, por suas disponibilidades e atenções.

““Computação não se relaciona mais a computadores. Relaciona-se a viver.””

NICHOLAS NEGROPONTE

SUMÁRIO

LISTA DE ILUSTRAÇÕES	7
LISTA DE TABELAS	8
1 INTRODUÇÃO	9
1.1 Motivação	9
1.2 Objetivos	10
1.3 Justificativa	10
2 SEGMENTAÇÃO DE IMAGENS	12
2.1 Conceito	12
2.2 Segmentação para seres humanos e para computadores	12
2.2.1 Thresholding	16
2.2.2 Baseado em Bordas	18
2.2.3 Baseado em Regiões	18
2.2.3.1 Split and Merge	18
2.2.3.2 Region growing (abordagem bottom-up)	19
3 FERRAMENTAS	21
3.1 Android Studio	21
3.2 OpenCV	21
3.3 Android NDK	21
4 CRONOGRAMA	22
4.1 Definição de Etapas	22
4.1.1 Escolha do Tema e Estudo de Viabilidade	22
4.1.2 Revisão Bibliográfica	22
4.1.3 Elaboração da Monografia	22
4.1.4 Estudo e Análise dos Algoritmos de Segmentação de Imagem	22
4.1.5 Implementação	22
4.1.6 Teste	23
4.1.7 Entrega do Relatório Final e Apresentação	23
4.2 Entregáveis	23
5 REFERÊNCIAS BIBLIOGRÁFICAS	24

LISTA DE ILUSTRAÇÕES

FIG.1.1	Tela do Aplicativo Android (imagem original e imagem segmentada).	9
FIG.1.2	Diagrama em Blocos do Projeto.	10
FIG.2.1	Imagem original(ARBELAEZ et al., 2011).	13
FIG.2.2	Imagens segmentadas por seres humanos, gerando 8 e 16 segmentos nas duas primeiras imagens e nas duas últimas, respectivamente.(ARBELAEZ et al., 2011).	13
FIG.2.3	Imagens segmentadas por seres humanos, gerando 22 e 26 segmentos nas duas primeiras imagens e nas duas últimas, respectivamente.(ARBELAEZ et al., 2011).	13
FIG.2.4	Imagem original(DATASET; BENCHMARK, 2017).	14
FIG.2.5	Resultados de 4 algoritmo de detecção de bordas (DATASET; BENCHMARK, 2017).	14
FIG.2.6	Imagem original(STANFORD, 2017).	17
FIG.2.7	Imagem segmentada pelo algoritmo threshold global.	17
FIG.2.8	Imagem segmentada pelo método de segmentação thresholding local, chamado de adaptive gaussian thresholding.	17
FIG.2.9	Uma imagem de um macaco (STANFORD, 2017) à esquerda e à direita segmentada pelo método de detecção de bordas em linguagem de programação python.	18
FIG.2.10	Uma representação visual de uma região (STANFORD, 2017) à esquerda e à direita segmentada pelo algoritmo quadtree em linguagem de programação python.	19
FIG.2.11	Imagem de uma moeda (STANFORD, 2017) à esquerda e à direita segmentada pelo algoritmo watersheed em linguagem de programação python.	20

LISTA DE TABELAS

TAB.4.1	Cronograma das atividades previstas	23
---------	---	----

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

Cada vez mais as máquinas estão sendo utilizadas em atividades de nossa sociedade. Atuando na substituição de profissionais ou no auxílio dos mesmos, elas estão presentes e participando de nosso cotidiano ativamente. A implementação de sistemas de visão computacional tem se tornado, portanto, uma necessidade mais forte à medida que as aplicações que envolvem o tratamento de imagens se desenvolvem e buscam se aproximar da visão e da análise humana. A segmentação é uma importante técnica utilizada nas atividades que envolvem o processamento digital de imagens. Diversas são as aplicações que fazem uso da identificação e análise de uma imagem, necessitando do estudo de regiões específicas das mesmas a fim de alcançar resultados e conclusões de forma eficiente. Por se tratar de um problema sem solução universal e de vasta aplicação, existem inúmeras possibilidades a serem exploradas e muito se tem estudado sobre essa área, com a evolução de novas técnicas e algoritmos que trazem uma nova abordagem ao problema. Esse projeto visa estudar e implementar técnicas de segmentação de imagens no ambiente Android, possibilitando o aprendizado de diferentes métodos da área de processamento de imagens, incluindo a área de visão computacional, bem como de desenvolvimento de aplicativos na plataforma Android. A Figura 1.1 exibe o resultado do aplicativo que este projeto de final de curso está desenvolvendo.

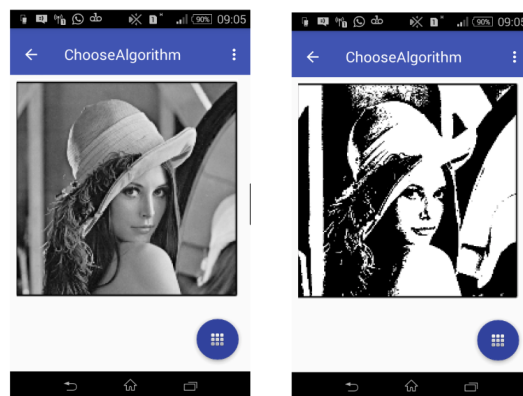


FIG. 1.1: Tela do Aplicativo Android (imagem original e imagem segmentada).

1.2 OBJETIVOS

O objetivo dessa pesquisa é o desenvolvimento de uma aplicação móvel, em plataforma Android, capaz de segmentar imagens por meio de diferentes algoritmos e técnicas de pré-processamento na área de Visão Computacional. A realização de todas as etapas de desenvolvimento até a concepção do produto final, que será disponibilizado para livre utilização, permitirá aos alunos maior conhecimento nesse assunto que é um dos domínios mais promissores da tecnologia e complementará a sua formação como engenheiros de Computação. A Figura 1.2 ilustra os passos do projeto, onde o aplicativo poderá adquirir imagens diretamente da câmera do dispositivo ou da galeria de imagens do dispositivo. O bloco denominado Pré-Processamento ilustra uma possível etapa de pré-processamento das imagens oriundas da câmera ou do banco de imagens, com a finalidade de tratar as imagens para os algoritmos de segmentação. O bloco seguinte, Técnicas de Segmentação implementa um ou mais métodos de segmentação, descritos no capítulo 2. Finalmente, o bloco Exibição vai exibir a imagem segmentada na tela do dispositivo.

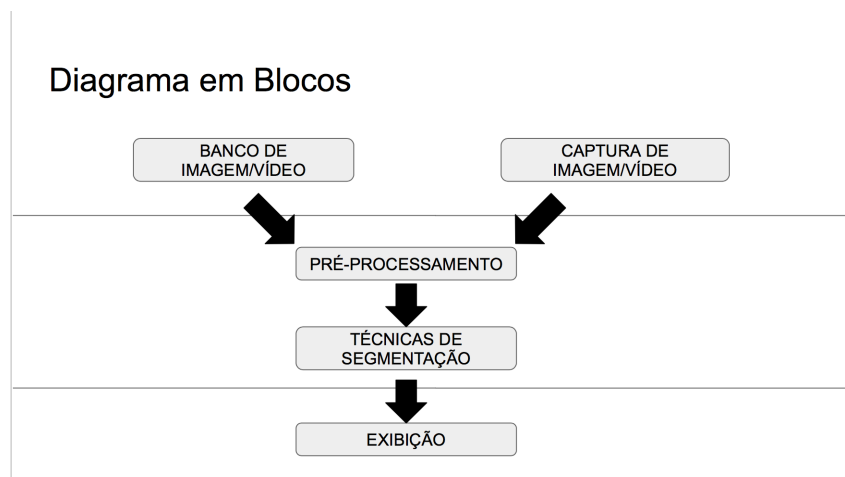


FIG. 1.2: Diagrama em Blocos do Projeto.

1.3 JUSTIFICATIVA

Este trabalho pode ser justificado com base nos seguintes pontos:

- Inexistência de uma solução única para o problema em questão. Trata-se de um problema ainda em aberto com muitas oportunidades a serem exploradas;
- Diversas aplicações necessitam da segmentação de imagens como parte importante

das suas atividades, além de outras que podem ser aprimoradas e desenvolvidas por meio de sua utilização;

- Visão computacional como domínio promissor da tecnologia, com uma atuação cada vez maior no mercado e interação com outros domínios como robótica e inteligência artificial.

2 SEGMENTAÇÃO DE IMAGENS

A segmentação é um problema do tipo "ill-posed" ("mal definido"), uma vez que não existe uma solução única, universal. Por isso, a segmentação é considerada o problema mais difícil de ser resolvido em análise de imagens (POGGIO et al., 1985).

2.1 CONCEITO

A segmentação de imagens tem como uma de suas interpretações como a divisão em regiões ou categorias consideradas "relevantes", que correspondem a objetos ou partes de objetos. Decidir o que é relevante em uma imagem depende do problema a ser resolvido, em que os objetos segmentados devem corresponder às áreas de interesse da aplicação. Dentre essas aplicações, podemos citar os seguintes exemplos:

- Aplicações militares: Reconhecimento de alvos terrestres, aéreos e navais;
- Análise de imagens médicas: Identificação de doenças como tumores;
- Veículos autônomos;
- Robótica.

2.2 SEGMENTAÇÃO PARA SERES HUMANOS E PARA COMPUTADORES

Para seres humanos, a identificação de regiões similares ou objetos diferentes presentes em uma imagem é um processo fácil. Nosso sistema cognitivo auxiliado por nosso sistema visual nos permite reconhecer e segmentar os objetos de forma instantânea sem nos darmos conta desse processo. Além disso, usamos a segmentação por distância como técnica auxiliar, uma vez que nossa visão estereoscópica nos fornece informação de profundidade. No caso de computadores, essa tarefa se torna mais complexa, pois envolve a análise de características de cada pixel ou da distribuição da população de pixels. Para isso, deve-se implementar algoritmos de segmentação, os quais serão explorados com maior profundidade nas seções subsequentes dessa pesquisa. A Figura 2.1 exibe uma imagem que foi manualmente segmentada por 4 seres humanos. É possível notar que cada pessoa não identificou como "relevantes" as mesmas áreas, conforme ilustrado pela Figura 2.2.



FIG. 2.1: Imagem original(ARBELAEZ et al., 2011).

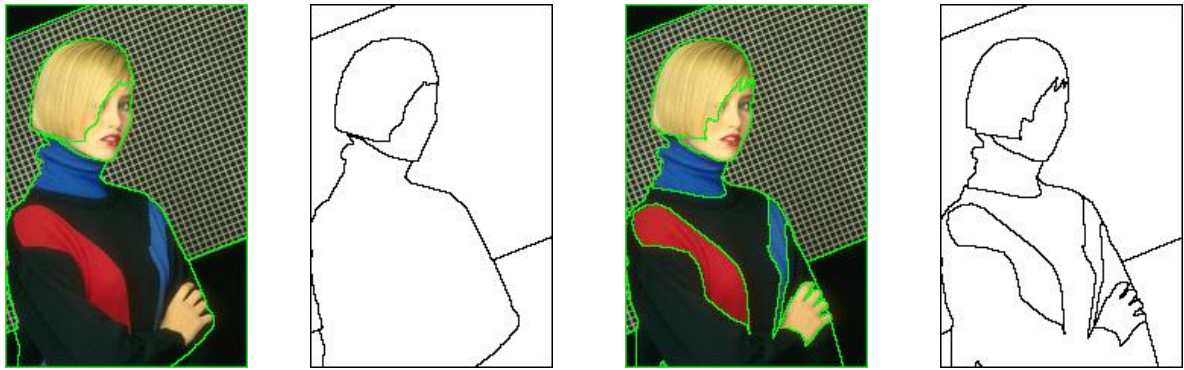


FIG. 2.2: Imagens segmentadas por seres humanos, gerando 8 e 16 segmentos nas duas primeiras imagens e nas duas últimas, respectivamente.(ARBELAEZ et al., 2011).

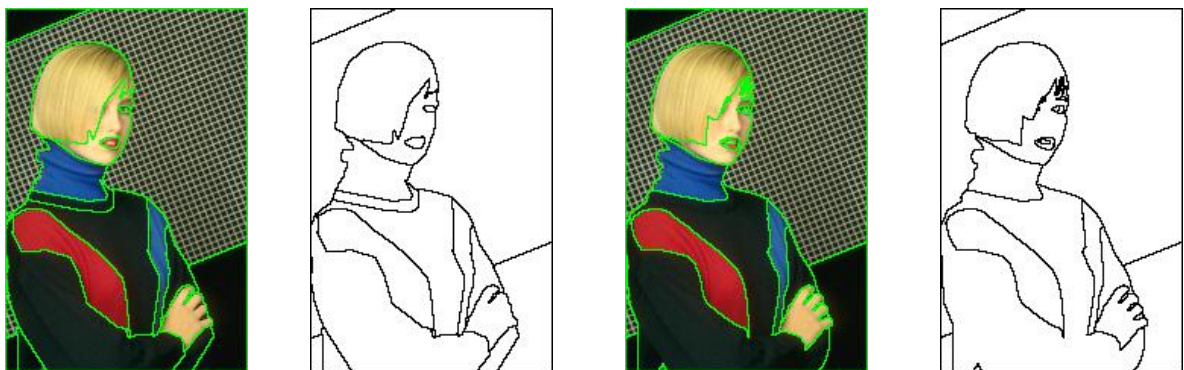


FIG. 2.3: Imagens segmentadas por seres humanos, gerando 22 e 26 segmentos nas duas primeiras imagens e nas duas últimas, respectivamente.(ARBELAEZ et al., 2011).

Ainda que o processo de segmentação para humanos seja fácil e automático, é comum e natural que diferentes pessoas identifiquem objetos ou partes de objetos distintos em uma

dada imagem. Isso se deve à percepção de relevância atribuída a cada região variar com a interpretação pessoal de cada um. O mesmo problema ocorre de forma mais acentuada com relação aos diferentes algoritmos. Cada algoritmo tem a sua própria abordagem para tratar do mesmo problema e, de acordo com sua implementação, leva a diferentes resultados, que podem ser analisados comparativamente. É importante dizer que o mesmo algoritmo pode ser mais ou menos eficiente de acordo com a imagem utilizada como dado de entrada, possibilitando diversos estudos na área como o tema desta pesquisa. A Figura 2.4 ilustra este problema(DATASET; BENCHMARK, 2017) .



FIG. 2.4: Imagem original(DATASET; BENCHMARK, 2017).

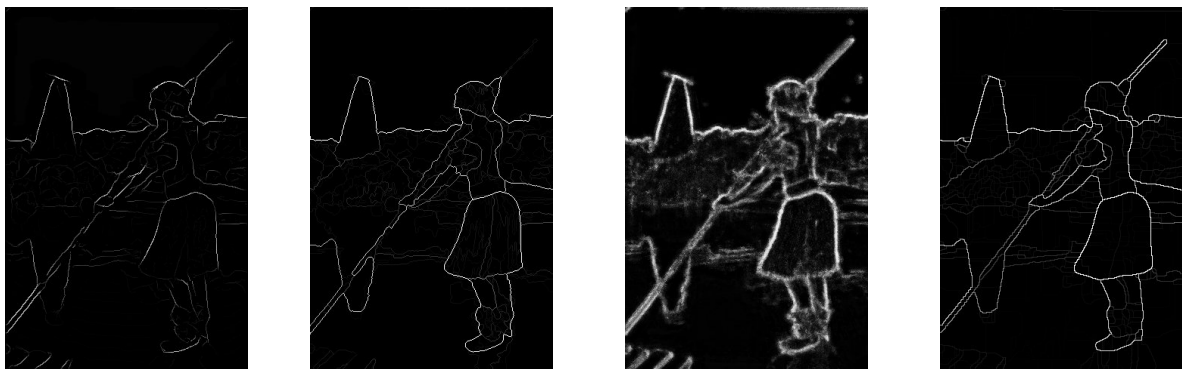


FIG. 2.5: Resultados de 4 algoritmo de detecção de bordas (DATASET; BENCHMARK, 2017).

Devido a grande variedade de algoritmos existentes, nos limitaremos à explicação dos principais e mais utilizados, os quais fornecem um bom entendimento das técnicas utilizadas e servem como base para o desenvolvimento de novas técnicas. O Capítulo 2.2

apresenta as técnicas estudadas.

3 ALGORITMOS DE SEGMENTAÇÃO

Este capítulo descreve 4 técnicas de segmentação de imagens, sendo a por limiar (Thresholding) a mais simples. As técnicas baseadas em bordas e regiões são mais complexas e demandam um ônus computacional mais elevado.

- Thresholding
- Método Baseado em Bordas
- Método Baseado em Regiões
 - Split and Merge
 - Watershed

3.0.1 THRESHOLDING

É o método mais simples de segmentação de imagens. Busca dividir a imagem em duas categorias: objetos (foreground) e plano de fundo (background). Cada pixel é alocado a uma categoria de acordo com seu valor em níveis de cinza.

Dado um limiar T (threshold), o pixel com valor f_{ij} e localizado na posição (i,j) é alocado à:

$$\begin{cases} \text{categoria 1, se: } f_{ij} \leq T. \\ \text{categoria 2, caso contrário.} \end{cases}$$

O limiar T pode ser escolhido manualmente, tentando diferentes valores de T e analisando qual deles é mais eficiente na identificação dos objetos de interesse. O threshold T também pode ser escolhido a partir do Histograma da imagem, e escolhe-se T como o valor entre as duas distribuições de cinza.

A seguir há a figura 2.6, uma imagem original do Einstein, e as duas formas de segmentação de imagens por limiar (thresholding). Enquanto na figura 2.7 há um limiar T fixo, na figura 2.8 ele é variável.

THRESHOLD GLOBAL

O mesmo valor de T é usado para a imagem inteira.

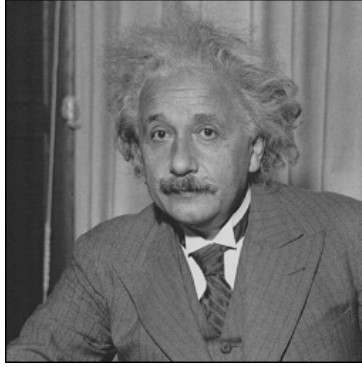


FIG. 3.1: Imagem original(STANFORD, 2017).



FIG. 3.2: Imagem segmentada pelo algoritmo threshold global.

THRESHOLD LOCAL (OU DINÂMICO)

Divide-se a imagem em regiões distintas e adota-se um valor T para cada uma delas, onde esse valor funcionará como threshold local.



FIG. 3.3: Imagem segmentada pelo método de segmentação thresholding local, chamado de adaptive gaussian thresholding.

Pode-se observar pela figura 2.8 que houve a separação em mais regiões nesse método

do que na figura 2.7, já que foi usado um limiar mais apropriado a cada região.

3.0.2 BASEADO EM BORDAS

Primeiramente, classifica-se os pixels como “borda” ou “não-borda”. Depois, divide-se a imagem em regiões, baseado nas bordas detectadas.

As bordas são identificadas por meio das descontinuidades, isto é, variações abruptas nos valores dos pixels.

Como pode-se perceber na figura 2.9, por esse método houve a segmentação da imagem de um macaco em regiões de olhos, nariz e outras partes.

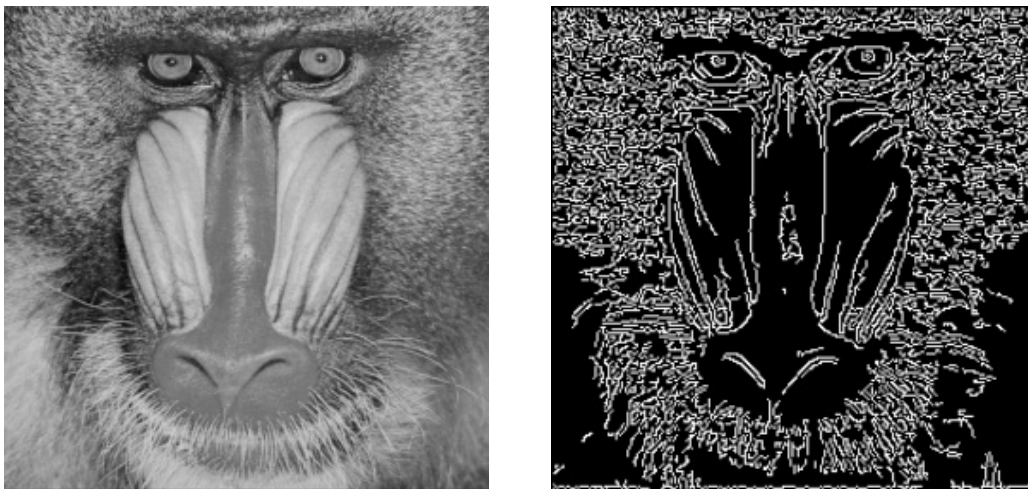


FIG. 3.4: Uma imagem de um macaco (STANFORD, 2017) à esquerda e à direita segmentada pelo método de detecção de bordas em linguagem de programação python.

3.0.3 BASEADO EM REGIÕES

O QUE É UMA REGIÃO?

Uma região pode ser “definida” como um grupo de pixels conectados com propriedades similares.

Porém, é um conceito importante e difícil de definir, já que depende da interpretação do que seria uma região em determinado caso.

Pelas figuras da seção 2.2 do capítulo 2, nota-se as diferentes interpretações do conceito de região pelas diferentes quantidades de regiões notadas por humanos nas Figuras 2.2 e 2.3. Tal fato também acontece com os computadores, como nota-se pelas figuras 2.5.

3.0.3.1 SPLIT AND MERGE

PROCEDIMENTO

As etapas fundamentais deste algoritmo segundo são:

- a) Criar critério para definir o que é uma área homogênea.
- b) Começar com a imagem completa e divide em 4 sub-imagens.
- c) Checar cada sub-imagem e dividi-la novamente em 4 novas sub-imagens caso ela não seja homogênea.
- d) Repetir Passo 3 até que não se consiga mais subdividir.
- e) Comparar sub-imagens com suas regiões vizinhas e agrupá-las se forem homogêneas.
- f) Repetir Passo 5 até que não se consiga mais agrupar.

EXEMPLO - QUADTREE

Um exemplo de segmentação baseada em região Split and Merge é o algoritmo quadtree. A figura 2.10 abaixo ilustra a segmentação de imagem por este algoritmo, e observa-se a segmentação da região que contém água na figura.

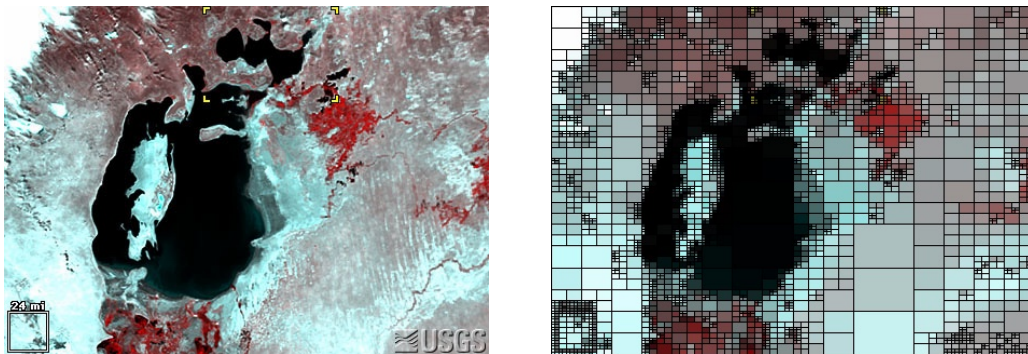


FIG. 3.5: Uma representação visual de uma região (STANFORD, 2017) à esquerda e à direita segmentada pelo algoritmo quadtree em linguagem de programação python.

3.0.3.2 REGION GROWING (ABORDAGEM BOTTOM-UP)

PROCEDIMENTO

As etapas fundamentais deste algoritmo são:

- a) Identificar o ponto de partida.

- b) Incluir pixels vizinhos com características similares (nível de cinza, textura, cor, etc).
- c) Continuar até que todos os pixels estejam associados com um dos pontos de partida.

EXEMPLO - WATERSHEED

Um exemplo de segmentação baseada no método Region growing é o algoritmo watershed. A figura 2.11 abaixo ilustra a segmentação de imagem por este algoritmo, e observa-se regiões distintas correspondentes à cada moeda da figura.

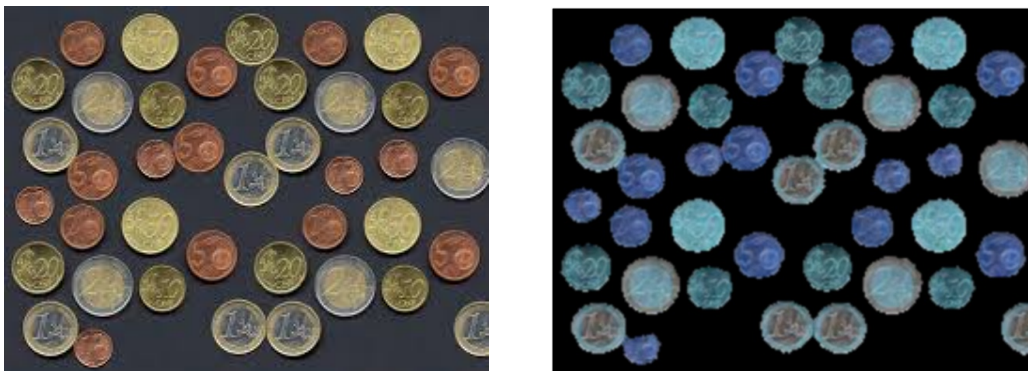


FIG. 3.6: Imagem de uma moeda (STANFORD, 2017) à esquerda e à direita segmentada pelo algoritmo watershed em linguagem de programação python.

4 FERRAMENTAS

4.1 ANDROID STUDIO

Plataforma de desenvolvimento de aplicativos para sistemas Android. Criado especificamente para esse propósito, a IDE oficial do Android oferece diversos recursos que aceleram o desenvolvimento e aumentam a qualidade dos aplicativos criados.

Tais recursos como ferramentas de edição, depuração, testes e geração de perfis de código motivaram a escolha dessa plataforma para o desenvolvimento da aplicação dessa pesquisa. Além da programação na linguagem Java, é permitido o desenvolvimento em linguagem nativa, C/C++, por meio da ferramenta Android NDK, que será utilizada na implementação dos algoritmos de segmentação presentes nessa aplicação(ANDROID, 2017b).

4.2 OPENCV

Biblioteca open source para o desenvolvimento de aplicativos na área de Visão Computacional. Conta com mais de 2500 algoritmos otimizados com abordagens clássicas e no estado da arte nas áreas de visão computacional e aprendizado de máquina, sendo, portanto, uma ferramenta amplamente usada em aplicações que envolvem o processamento de imagens (mais de 14 milhões de downloads). Tem interfaces para linguagens como C++, C, Python e Java e suporta diferentes plataformas como Windows, Linux, Mac OS e Android. A maior performance é obtida com seu uso em C++ pelo fato de ser sua linguagem nativa.(OPENCV, 2017)

4.3 ANDROID NDK

O Android Native Developmentt Kit (NDK) é um conjunto de ferramentas que permite a codificação de arquivos de projeto em linguagens nativas como C/C++. Como dito anteriormente, essa ferramenta auxiliará na implementação dos algoritmos de segmentação de imagens que farão parte de nossa aplicação.(ANDROID, 2017a)

5 CRONOGRAMA

5.1 DEFINIÇÃO DE ETAPAS

5.1.1 ESCOLHA DO TEMA E ESTUDO DE VIABILIDADE

A escolha do tema foi a primeira etapa do projeto. O uso de dispositivos móveis, bem como de suas respectivas câmeras tem sido cada dia mais frequentes. Essas câmeras captam informações do ambiente que os cercam e a segmentação de imagem pode ser usado no processamento de imagens de forma a desenvolver soluções computacionalmente automatizáveis.

Após a escolha do tema, foi feito o estudo de viabilidade, de forma a se verificar a possibilidade da cumprimento do objetivo do tema em tempo aceitável.

5.1.2 REVISÃO BIBLIOGRÁFICA

Rêferências tais como livros, artigos e outras referências foram estudadas durante o período de revisão bibliográfica, de forma a permitir um embasamento bibliográfico do projeto.

5.1.3 ELABORAÇÃO DA MONOGRAFIA

Esta fase do projeto se estende até o termino do projeto de fim de curso. Nesta última, confecciona-se um relatório utilizando-se os conhecimentos adquiridos desde o início do projeto.

5.1.4 ESTUDO E ANÁLISE DOS ALGORITMOS DE SEGMENTAÇÃO DE IMAGEM

Na aplicação proposta, alguns algoritmos de segmentação de imagem serão implementados no dispositivo móvel, de modo a estudá-los e verificar o mais apropriado à aplicação.

5.1.5 IMPLEMENTAÇÃO

A implementação ocorrerá após o estudo dos algoritmos e da definição de quais deles são mais indicados à proposta. Em uma primeira fase os algoritmos serão implementados em outros ambientes de desenvolvimento e em uma segunda etapa será realizada a portabilidade para o ambiente Android.

5.1.6 TESTE

Os testes com as imagens em diferentes algoritmos serão realizados e, com a implementação pronta, pode-se observar e comparar os resultados dos algoritmos.

5.1.7 ENTREGA DO RELATÓRIO FINAL E APRESENTAÇÃO

A partir das análises e de todo conteúdo, implementação, análises e testes, será feito o relatório e a apresentação à banca.

5.2 ENTREGÁVEIS

Os entregáveis serão:

- Projeto do Aplicativo
- Escolha dos Algoritmos
- Aplicativo em Ambiente Android de Segmentação de Imagens implementado
- Testes e análises comparativas

Etapa	Meses								
	FEV	MAR	ABR	MAI	JUN	JUL	AGO	SET	OUT
Escolha do Tema e Estudo de Viabilidade	X	X	X						
Revisão Bibliográfica	X	X	X	X					
Elaboração da Monografia	X	X	X	X	X	X	X	X	X
Estudo e Análise dos Algoritmos de Segmentação de Imagem						X	X	X	
Implementação						X	X	X	
Teste								X	
Entrega do Relatório Final e Apresentação									X

TAB. 5.1: Cronograma das atividades previstas

6 REFERÊNCIAS BIBLIOGRÁFICAS

- DEVELOPERS ANDROID. Primeiros passos com o NDK. Disponível em: <https://developer.android.com/ndk/guides/index.html?hl=pt-br>. Acesso em: 10 mai. de 2017.
- DEVELOPERS ANDROID. Tudo de que você precisa para criar aplicativos no Android. Disponível em: <https://developer.android.com/studio/features.html>. Acesso em: 10 mai. de 2017.
- ARBELAEZ, P.; MAIRE, M.; FOWLKES, C. ; MALIK, J. Contour detection and hierarchical image segmentation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 33, n. 5, p. 898–916, 2011.
- THE BERKELEY SEGMENTATION DATASET AND BENCHMARK. Berkeley Segmentation Dataset. Disponível em: <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300>. Acesso em: 10 mai. de 2017.
- OPENCV. OpenCV About. Disponível em: <http://opencv.org/about.html>. Acesso em: 10 mai. de 2017.
- POGGIO, T.; TORRE, V. ; KOCH, C. Computational vision and regularization theory. **Nature**, v. 317, n. 26, p. 314–319, 1985.
- STANFORD. The Stanford Center for Image Systems Engineering. Disponível em: <https://scien.stanford.edu/index.php/test-images-and-videos/>. Acesso em: 10 mai. de 2017.