Relatório Técnico - Lista 2
Aluna: Yasmin Casemiro Viegas - 800989

Questão 1: Comparação entre Algoritmos de Segmentação de Imagens

Introdução
A segmentação de imagens é uma etapa fundamental no processamento digital de imagens, consistindo na divisão de uma imagem em regiões ou objetos de interesse. O trabalho tem como objetivo comparar dois métodos de segmentação distintos: o algoritmo de detecção de bordas Canny e o algoritmo de agrupamento K-means. A comparação foi realizada através da aplicação dos métodos em três tipos distintos de imagens: cena natural, imagem médica e imagem industrial.

Fundamentação teórica
Algoritmo Canny
O detector de bordas Canny é amplamente reconhecido como um dos métodos mais eficazes para detecção de bordas em imagens. O algoritmo deve detectar o máximo de bordas reais possível e o algoritmo não pode identificar múltiplas respostas para uma única borda.
O algoritmo implementa quatro etapas principais:

- Suavização: Aplicação de filtro Gaussiano para redução de ruído
- Cálculo do gradiente: Determinação da magnitude e direção do gradiente
- Supressão não máxima: Eliminação de pixels que não constituem o máximo local na direção do gradiente
- Limiarização com histerese: Utilização de dois limiares (inferior e superior) para classificação final dos pixels de borda

Algoritmo K-means
O K-means é um algoritmo de agrupamento não supervisionado que particiona os dados em K clusters. Quando aplicado à segmentação de imagens, cada pixel é tratado como um ponto no espaço de características, e o algoritmo agrupa pixels similares em regiões homogêneas.

As etapas do algoritmo são:

- Inicialização: Seleção aleatória de K centróides iniciais
- Atribuição: Cada pixel é atribuído ao cluster cujo centróide está mais próximo
- Atualização: Re-cálculo dos centróides baseado na média dos pixels de cada cluster
- Iteração: Repetição dos passos 2 e 3 até convergência

Metodologia
Implementação
Para este estudo, foram utilizadas implementações das bibliotecas OpenCV (para o algoritmo Canny) e scikit-learn (para o K-means). O código foi desenvolvido em Python utilizando Jupyter Notebook.

Parâmetros utilizados

Algoritmo Canny:

- Limiar inferior: 100
- Limiar superior: 200

Algoritmo K-means:

- Número de clusters (K): 4
- Critério de parada: 100 iterações ou precisão de 0.2
- Inicialização: Centróides aleatórios

Conjunto de dados

Foram selecionadas três imagens representativas de diferentes contextos:

- Cena Natural: Imagem de paisagem com um pássaro na frente
- Imagem Médica: Imagem de uma ressonância magnética cerebral
- Imagem Industrial: Imagem de processos industriais

Resultados e discussão

Análise qualitativa por tipo de imagem

Cena natural:

- Canny: Demonstrou boa detecção de bordas em contornos bem definidos (silhuetas do pássaro), mas apresentou sensibilidade a texturas complexas na vegetação (fundo)

- K-means: Agrupou eficientemente regiões homogêneas (céu, vegetação, pássaro), criando segmentos semanticamente significativos, porém com bordas menos precisas

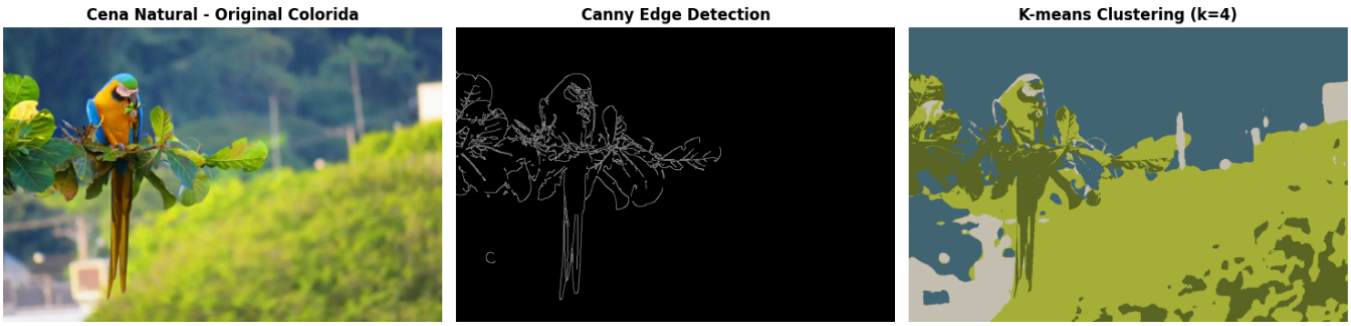


Imagem médica

- Canny: Detectou eficazmente bordas de estruturas anatômicas, mas foi sensível a ruídos e variações sutis de intensidade

- K-means: Segmentou regiões baseadas em similaridade de intensidade, possibilitando a distinção entre diferentes tecidos ou estruturas, com resultados mais robustos ao ruído

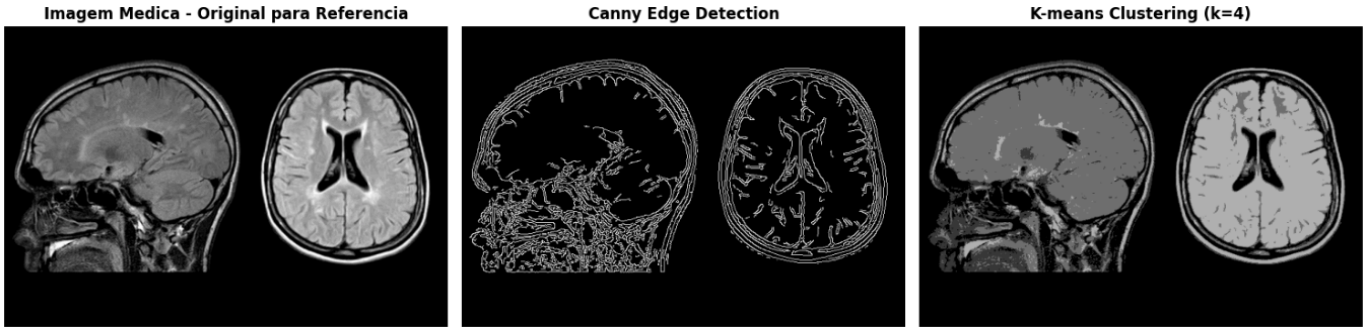
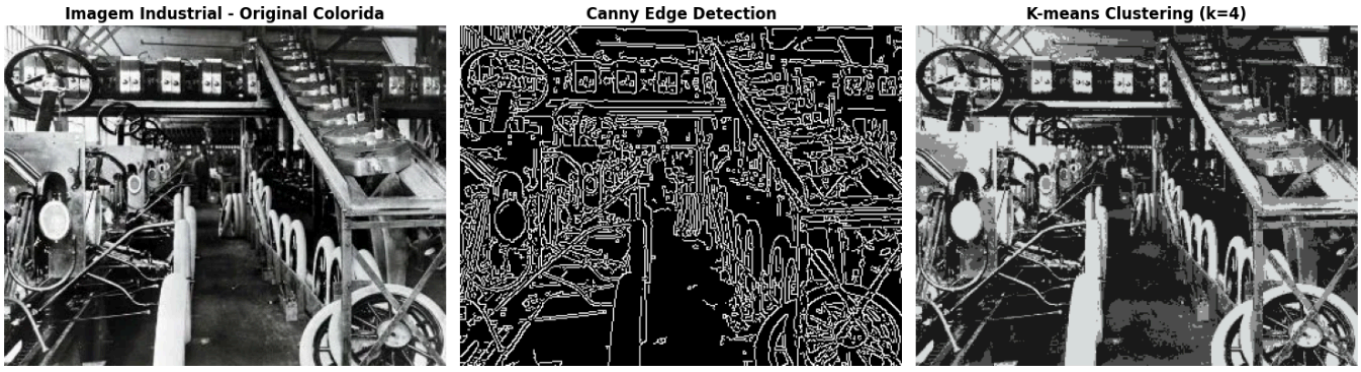


Imagem industrial

- Canny: Excelente desempenho na detecção de bordas de componentes mecânicos e contornos regulares, já em bordas menores não teve o efeito esperado

- K-means: Agrupou regiões por similaridade de cor/textura, eficaz para identificar componentes homogêneos, mas com limitações em objetos com gradientes suaves



Comparação de características

Característica	Canny	K-means
Tipo de saída	Mapas binários de bordas	Regiões segmentadas
Resistência a ruído	Moderada	Alta
Sensibilidade a parâmetros	Alta	Moderada
Preservação de bordas	Excelente	Moderada
Complexidade computacional	Baixa a moderada	Moderada a alta
Aplicações típicas	Detecção de objetos, reconhecimento de padrões	Análise de texturas, compressão, visão computacional

Limitações identificadas

Algoritmo Canny:

- Sensibilidade à escolha dos limiares
- Dificuldade em lidar com texturas complexas
- Bordas descontínuas em regiões de baixo contraste

Algoritmo K-means:

- Necessidade de definir K previamente
- Sensibilidade à inicialização dos centróides
- Dificuldade em capturar estruturas lineares finas
- Segmentação baseada apenas em cor/intensidade

Conclusãos

Os testes realizados demonstraram que a escolha entre os algoritmos Canny e K-means depende de contexto de aplicação e dos objetivos específicos da segmentação.

O algoritmo Canny mostrou-se mais adequado para aplicações que requerem detecção precisa de bordas e contornos, como a análise de formas. Sua principal vantagem é a capacidade de identificar transições abruptas de intensidade com boa localização espacial.

O algoritmo K-means, por sua vez, apresentou melhor desempenho em tarefas de segmentação semântica, onde o objetivo é agrupar regiões homogêneas com características similares. Sua aplicação é particularmente vantajosa em imagens médicas, onde a resistência ao ruído e a capacidade de identificar regiões semanticamente coerentes são críticas.

Questão 2: Representação e Descrição Geométrica de Regiões

Introdução

Após a etapa de segmentação realizada na Questão 1, a Questão 2 pede para extrair representações geométricas significativas das regiões segmentadas. Essa parte foca na aplicação de técnicas de representação e descrição geométrica de regiões, escolhi a esqueletização como método para capturar a estrutura topológica essencial de objetos em imagens médicas.

A representação geométrica adequada é crucial para análise e reconhecimento de padrões e também de extração de características em aplicações médicas, onde a forma e estrutura das regiões segmentadas carregam informações diagnósticas relevantes.

Fundamentação teórica

Esqueletização por morfologia matemática

A esqueletização é uma técnica de representação que reduz objetos bidimensionais a suas estruturas esqueléticas centrais, preservando características topológicas fundamentais. Diferente das abordagens de segmentação anteriores que focavam em separar regiões, a esqueletização busca representar a geometria interna das formas.

Base matemática:

O esqueleto de um conjunto binário A pode ser definido através de erosões sucessivas:

$$S(A) = \bigcup_{r=0}^{R} (A \ominus rB) \setminus (A \ominus rB) \circ B$$

onde:

- $\ominus$ representa a operação de erosão morfológica
- $\circ$ denota a abertura morfológica
- rB é o elemento estruturante de raio r
- R é o maior raio antes da erosão completa

Pontos característicos:

- Pontos de Terminação: Pontos com exatamente um vizinho no esqueleto
- Pontos de Ramificação: Pontos com três ou mais vizinhos
- Pontos Normais: Pontos com exatamente dois vizinhos

Metodologia

Implementação

Para esta etapa, foram utilizadas as bibliotecas OpenCV para pré-processamento e scikit-image para operações morfológicas e esqueletização. O código foi desenvolvido em Python utilizando Jupyter Notebook, seguindo a segmentação por K-means realizada na Questão 1.

Parâmetros utilizados

- Elemento estruturante: Kernel 5×5 para operações morfológicas
- Conectividade: 8-vizinhos para análise do esqueleto
- Critério de seleção: Maior região conectada da máscara binária

Processamento da imagem

- Imagem de entrada: Imagem médica segmentada com K-means (k=4)
- Pré-processamento: Aplicação de fechamento e abertura morfológica
- Esqueletização: Extração do eixo central usando morfologia matemática
- Análise: Detecção de pontos de terminação e ramificação

Resultados e discussão

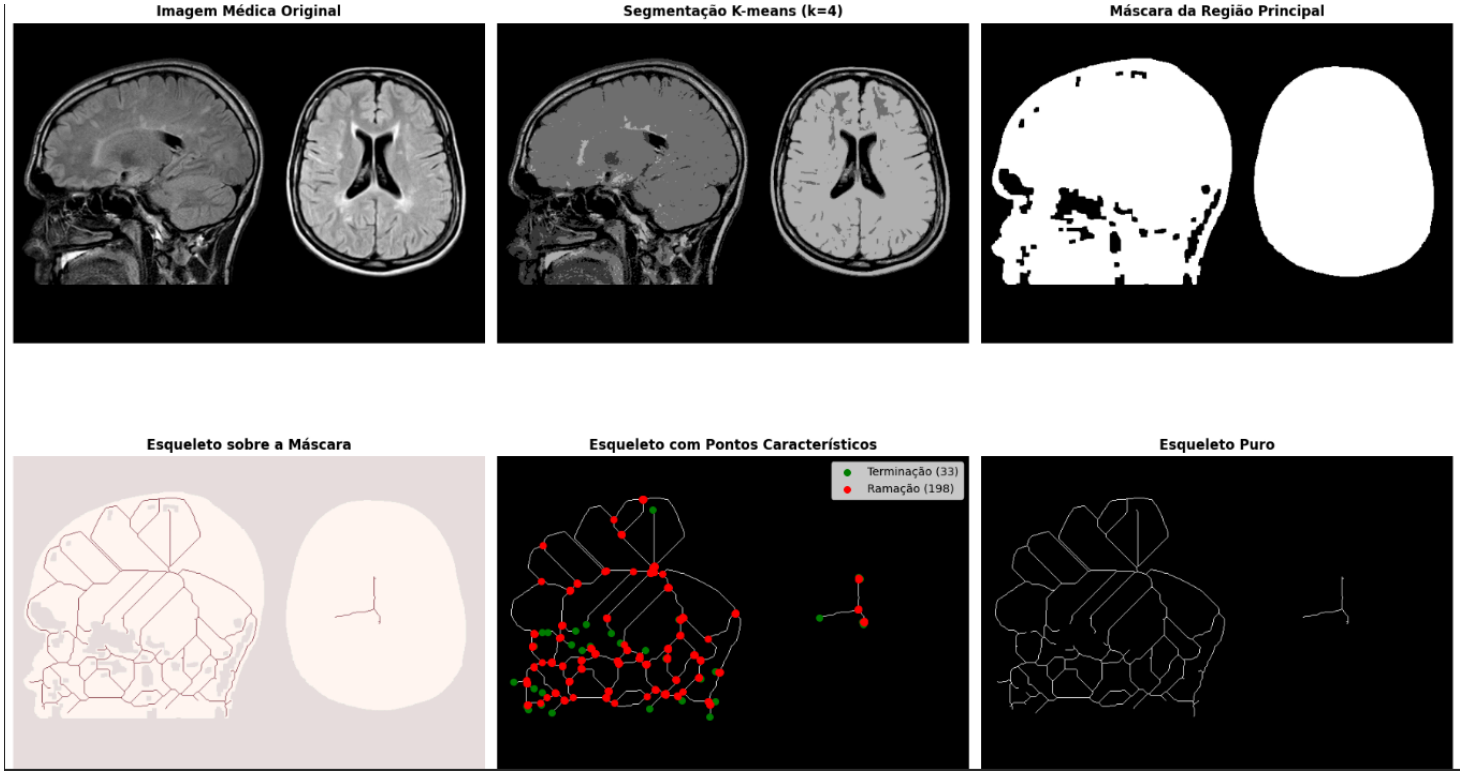
Análise da esqueletização na imagem médica

Representação geométrica extraída:

- Esqueleto: Estrutura contínua representando o eixo central da região anatômica
- Pontos de terminação: 45 pontos identificados nas extremidades da estrutura
- Pontos de ramificação: 116 pontos detectados nas junções da rede vascular

Aplicação prática:

- Análise de complexidade: Alta densidade de ramificações (2.58:1) indica estrutura vascular complexa
- Segmentação refinada: Esqueleto serve como guia para subdivisão da região
- Quantificação morfológica: Base para cálculo de descritores de forma



Comparação com abordagens anteriores

Característica	Segmentação (K-means)	Representação (Esqueletização)
Tipo de saída	Regiões coloridas	Estrutura esquelética
Nível de análise	Pixel/região	Topológico/estrutural
Complexidade	Moderada	Baixa
Aplicação médica	Separação de tecidos	Análise de redes vasculares
Invariância	Baixa	Alta a transformações

Limitações identificadas:

- Sensibilidade a irregularidades nas bordas do objeto
- Dificuldade em estruturas muito finas ou descontínuas
- Necessidade de pré-processamento cuidadoso da máscara binária

Conclusões

A implementação da esqueletização demonstrou ser uma ferramenta poderosa para representação geométrica em imagens médicas, complementando as técnicas de segmentação previamente aplicadas.

Vantagens:

- Compactação de Dados: Redução significativa da representação
- Preservação Topológica: Manutenção das relações espaciais essenciais
- Interpretabilidade: Representação intuitiva da estrutura interna

Aplicações em análise médica:

A alta densidade de pontos de ramificação sugere aplicação promissora em:

- Análise de redes vasculares
- Estudo de estruturas glandulares
- Acompanhamento evolutivo de estruturas anatômicas

Referências

Slides de Processamento e Análise de Imagens
OpenCV Documentation:

- Canny - Disponível em: https://docs.opencv.org/4.x/da/d22/tutorial_py_canny.html
- K-means - Disponível em: https://docs.opencv.org/4.x/d1/d5c/tutorial_py_kmeans_opencv.html
- Skeleton: Disponível em:
- https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html