

V Semana Estadual de Tecnologia da Informação e Comunicação - TIC

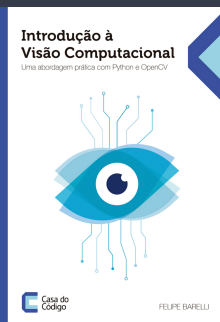
Lançamento do Livro Introdução à Visão Computacional

# Visão Computacional

As aplicações e os impactos em nosso cotidiano

Palestrante

Felipe da Costa Barelli



# Quem é esse rapaz aí?

- ❑ **Bacharel em Ciência da Computação pela Universidade Vila Velha**
- ❑ **Especialista em Engenharia Elétrica pelo Instituto Federal do Espírito Santo**
  - ❑ Sistema de controle de nível com realimentação por imagem.
  - ❑ Sistema de monitoramento do uso de Equipamentos de Proteção Individual.
- ❑ **Mestrando em Informática pela Universidade Federal do Espírito Santo**
  - ❑ Linha de pesquisa em Inteligência Computacional.
- ❑ **Programador de Sistemas na Sollo Brasil**
  - ❑ Desenvolvimento de chatbots.
  - ❑ Desenvolvimento de sistemas inteligentes para monitoria de qualidade.

# Quem são vocês?

- ❑ **Estudantes ou profissionais da área de desenvolvimento de software?**
- ❑ **Profissionais de outras áreas?**
- ❑ **Alguém já trabalhou no desenvolvimento de um sistema de Visão Computacional?**



# O que posso esperar desta palestra?

- ❑ Entender o que é um sistema de Visão Computacional.
- ❑ Conversar sobre as aplicações desses sistemas em nosso cotidiano.
- ❑ Aprender técnicas e conhecer ferramentas para desenvolvê-los.

# Qual o motivo para ação?

- ❑ **Conhecer melhor a tecnologia que está substituindo homens por máquinas.**
- ❑ **Para quem deseja seguir carreira em uma empresa.**
  - ❑ Ser capaz de propor soluções para automatizar processos manuais.
  - ❑ Desenvolver ferramentas de baixo custo comparadas as disponíveis no mercado.
- ❑ **Para quem deseja empreender ou iniciar uma startup de tecnologia.**
  - ❑ Conhecer as tecnologias que viabilizam o desenvolvimento desses sistemas.
  - ❑ Desenvolver soluções para automatizar processos realizados por humanos.
- ❑ **Para quem deseja atuar na área de pesquisa.**
  - ❑ Conhecer e estudar os algoritmos utilizados nesses sistemas.
  - ❑ Propor novos algoritmos ou otimizações dos algoritmos existentes.

# Expectativa x Realidade



# Beleza! Mas o que é Visão Computacional?

- ❑ **Visão Computacional é a ciência que estuda e desenvolve tecnologias que permitem máquinas enxergar e extrair características do meio, através de imagens capturadas por câmeras de vídeo, sensores, scanners e outros dispositivos.**

**Computer Vision**

**Dana H. Ballard, Christopher M. Brown**

**Prentice Hall; First edition (May 1, 1982)**

# Aplicações da Visão Computacional

- ❑ **Detecção, classificação e contagem de objetos.**
- ❑ **Rastreamento de objetos em movimento.**
  - ❑ Estimar a velocidade e a posição de um alvo em movimento.
- ❑ **Reconstrução de cena.**
  - ❑ Desenvolver um modelo tridimensional da cena a partir de duas ou mais imagens.
- ❑ **Restauração de imagens.**
  - ❑ Remoção de ruídos e reconstrução de regiões com falhas.



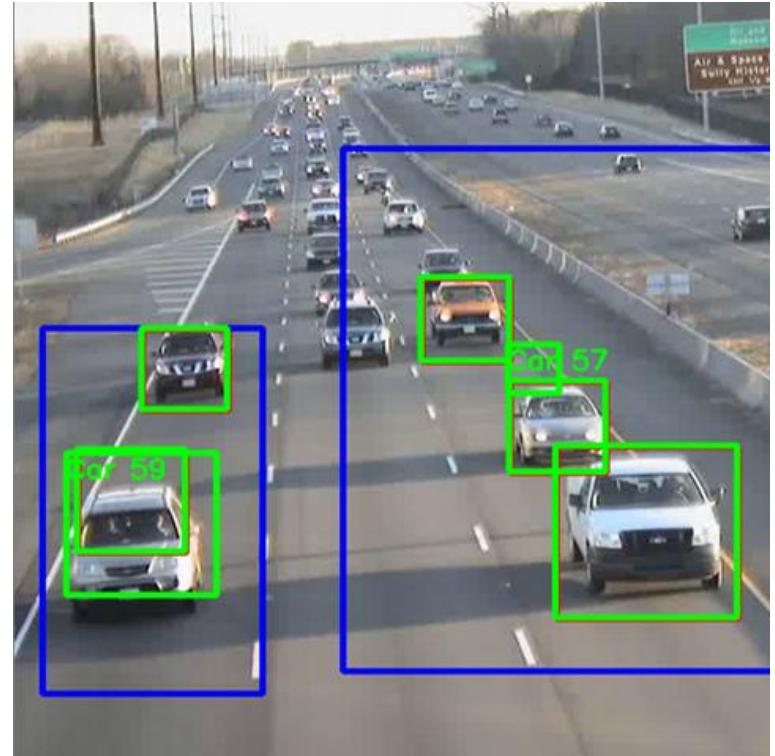
# Supermercados inteligentes

- ❑ Loja sem caixas de atendimento e sem filas. O cliente entra, escolhe o que quer, põe tudo em uma sacola e vai embora. Simples assim.



# Análise de tráfego

- ❑ Sistemas capazes de analisar o fluxo de veículos e a velocidade de cada um deles em uma rodovia.

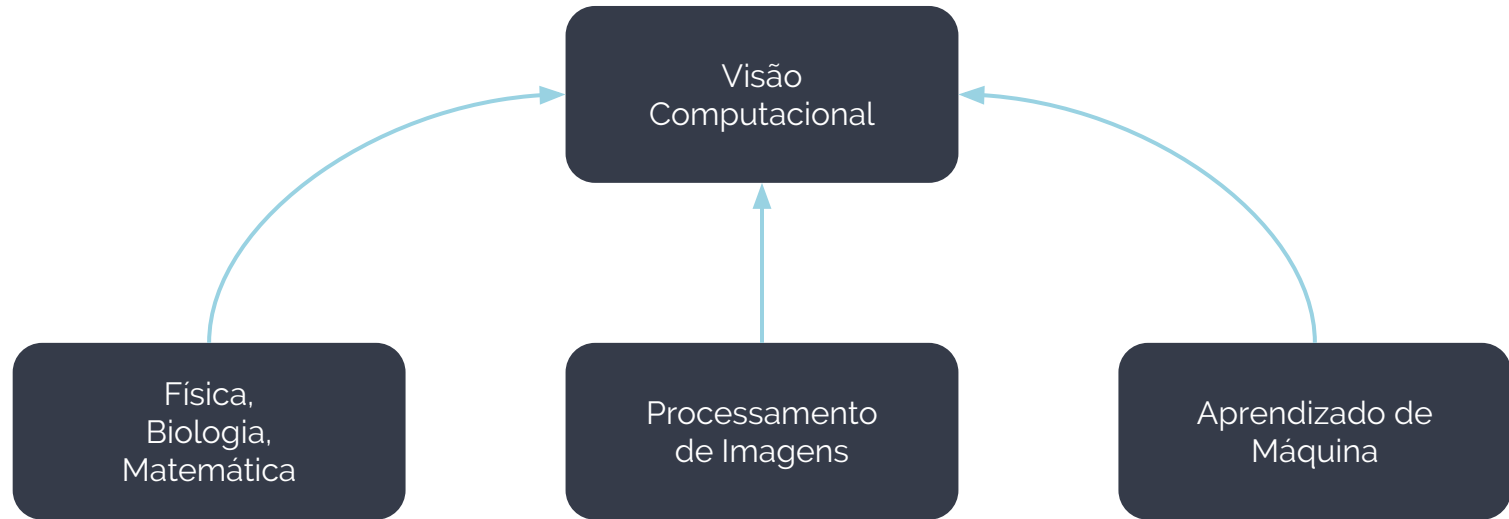


# Sistema de inspeção visual

- ❑ Indústria utiliza sistemas de Visão Computacional para inspecionar se peças metálicas foram cortadas corretamente e não possuem falhas.



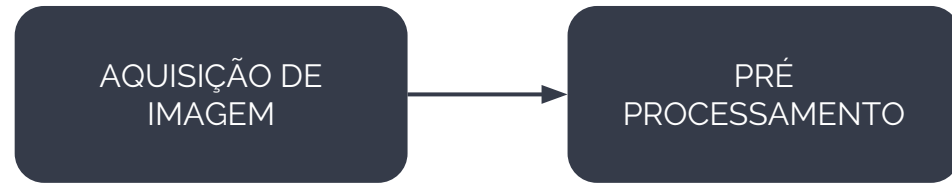
# Campos de estudo relacionados



# Fluxo comum desses sistemas

AQUISIÇÃO DE  
IMAGEM

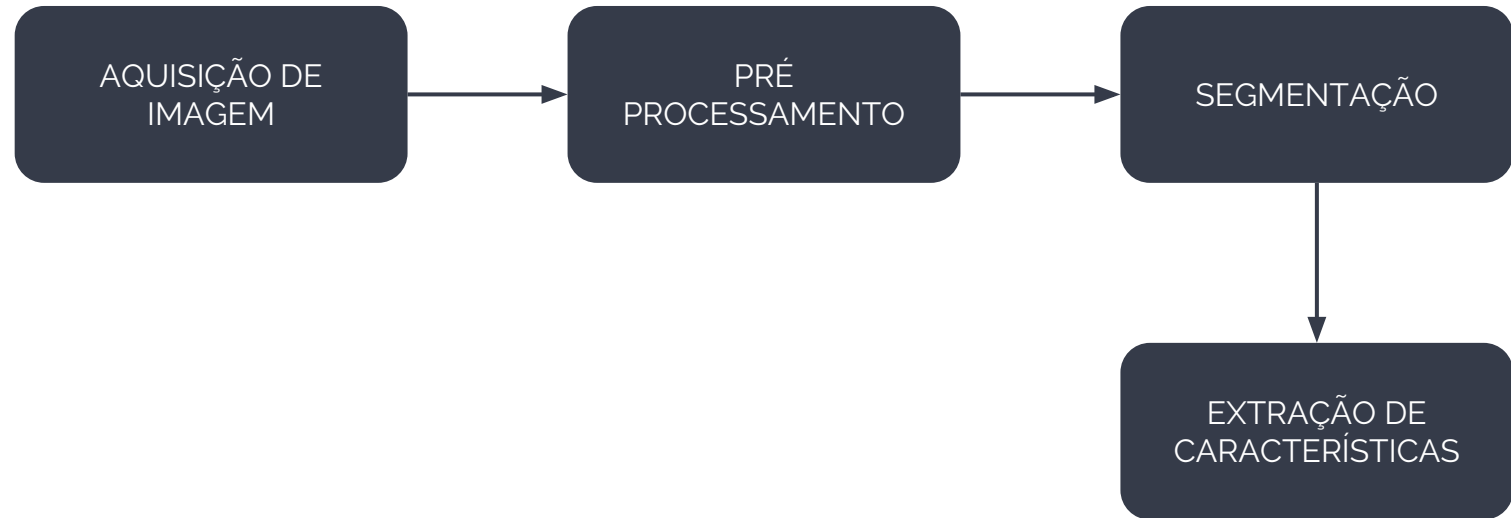
# Fluxo comum desses sistemas



# Fluxo comum desses sistemas

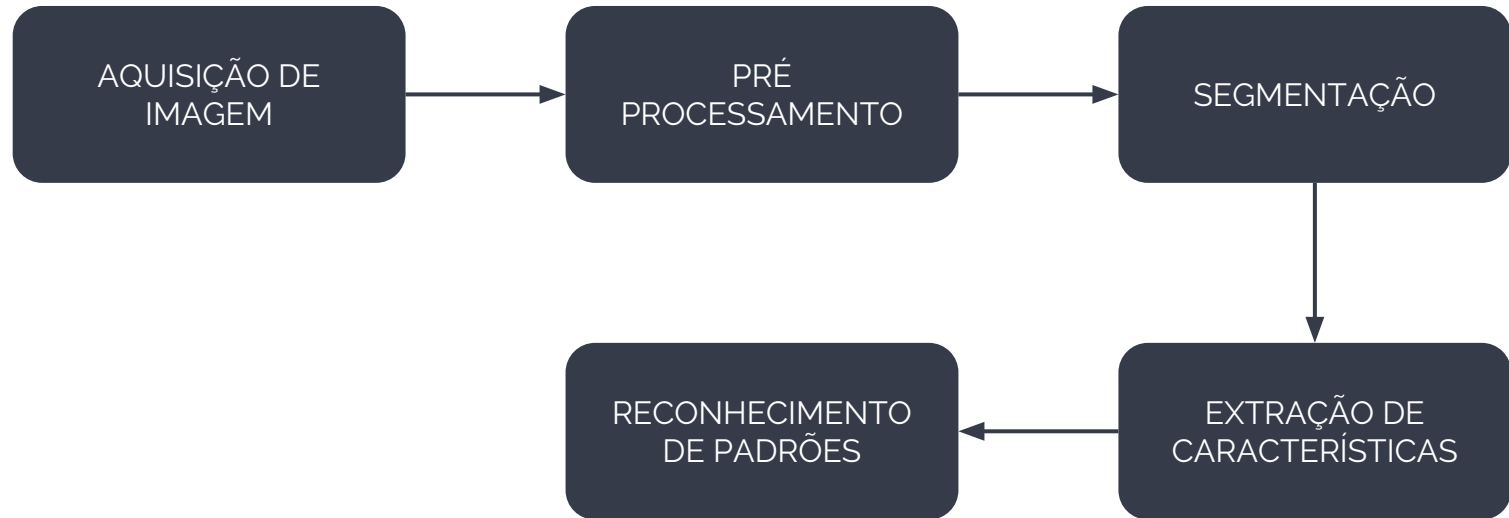


# Fluxo comum desses sistemas

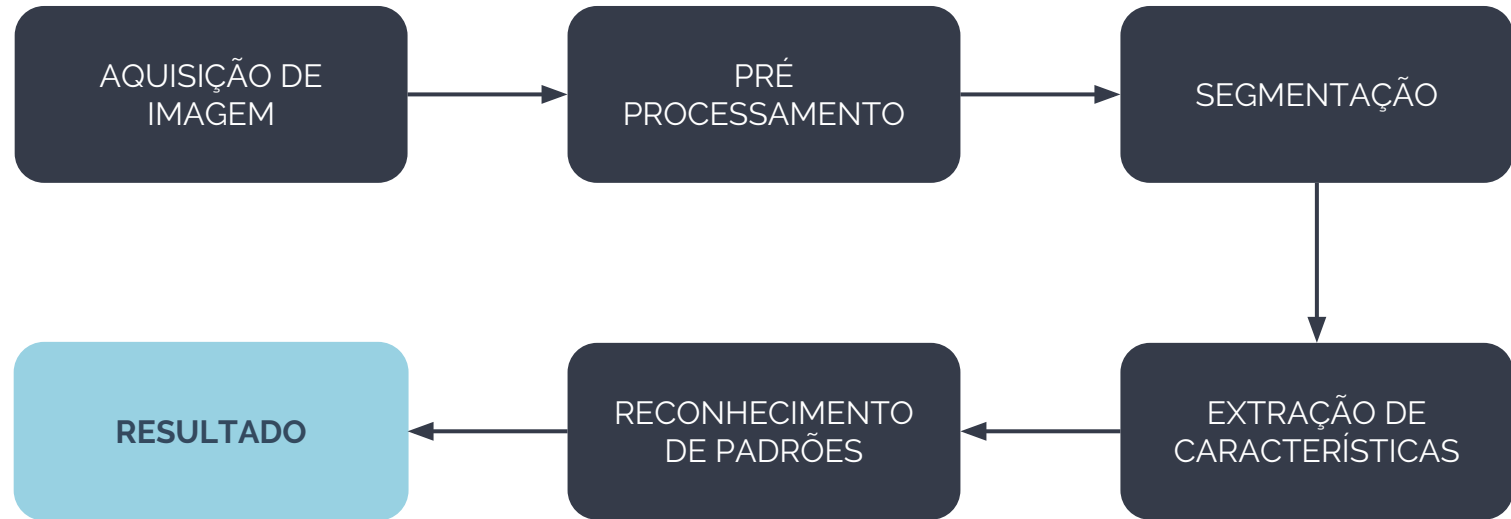




# Fluxo comum desses sistemas

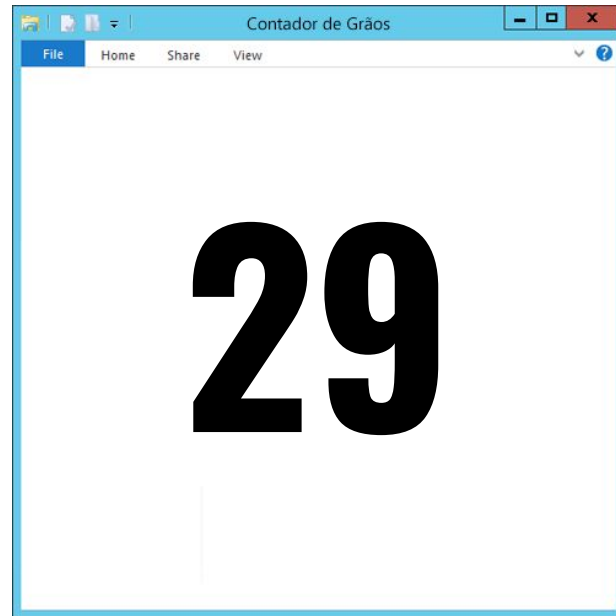


# Fluxo comum desses sistemas



# Sistema Contador de Grãos

# Sistema Contador de Grãos



# Fluxo do sistema de contagem de grãos



# Aquisição de Imagem

# Capturar cena



# Pré-processamento



# Converter imagem RGB para Tons de Cinza



# Imagem em Tons de Cinza



107	085	076	079
121	102	090	093
134	123	110	105
114	110	095	089

# Redimensionar imagem



# Segmentação do objeto de interesse

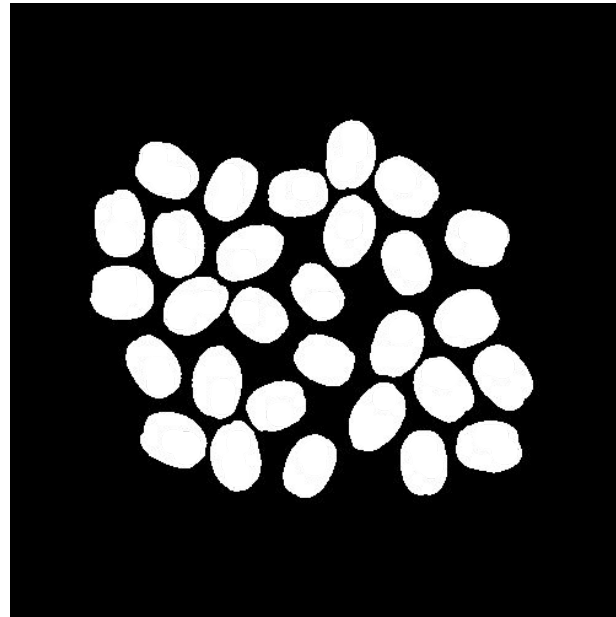
# Segmentação por binarização

- ❑ Considerando o valor 180 como limiar.

153	153	153	153	153	153
153	255	255	153	217	153
153	255	255	153	217	153
153	153	153	153	153	153
153	153	183	153	255	153
153	153	183	153	153	153

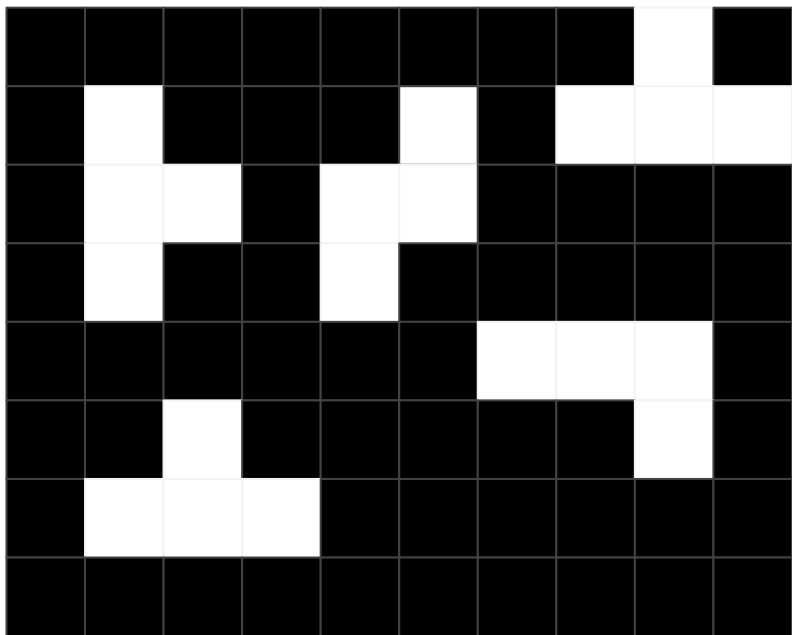
0	0	0	0	0	0
0	1	1	0	1	0
0	1	1	0	1	0
0	0	0	0	0	0
0	0	1	0	1	0
0	0	1	0	0	0

# Segmentação por binarização



# Obtendo o total de objetos em imagens binárias

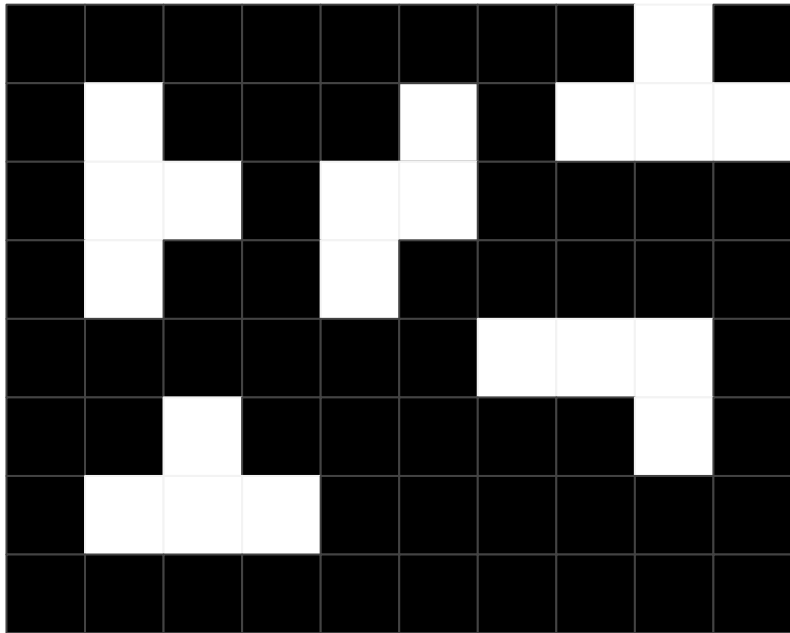
# Algoritmo baseado na área do objeto



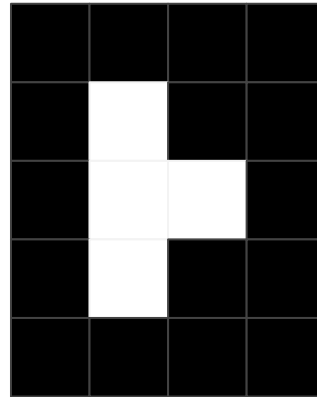
□ Total de pixels brancos: 20



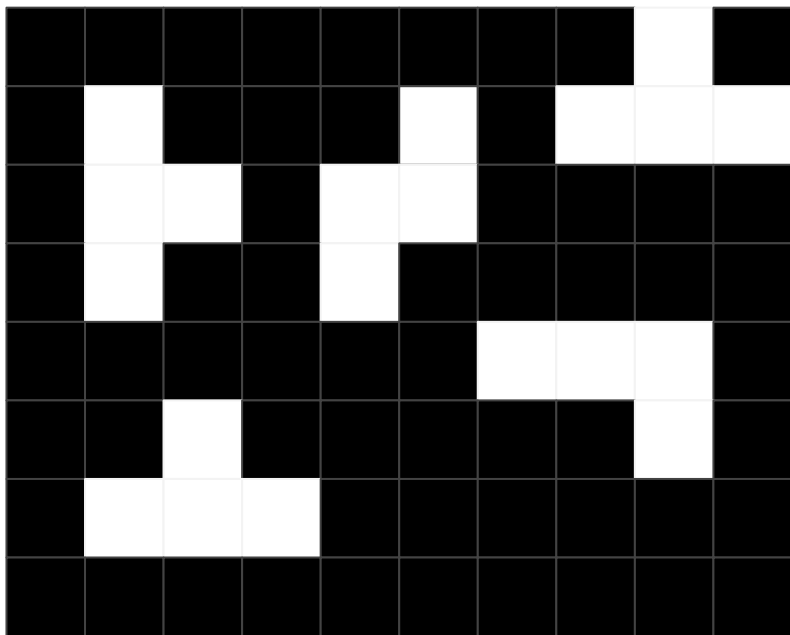
# Algoritmo baseado na área do objeto



- ❑ Total de pixels brancos: 20
- ❑ Área de um objeto:



# Algoritmo baseado na área do objeto



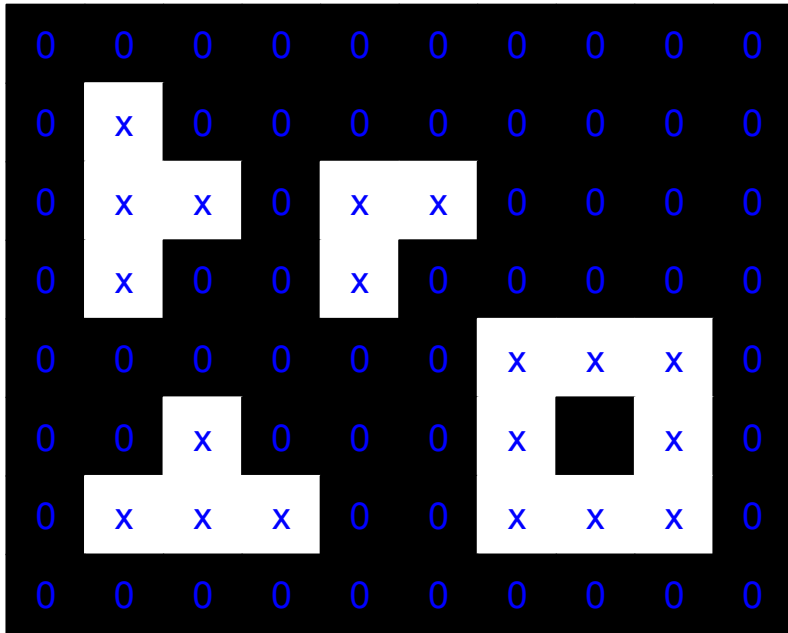
- ❑ Total de pixels brancos: 20
- ❑ Área de um objeto: 4
- ❑ Total de objetos:  
 $20 / 4 = 5$  objetos

# Vamos pensar um pouco?

- ❑ Qual seria uma limitação desse algoritmo de contar objetos em imagens a partir do valor médio de pixels brancos que os representam?

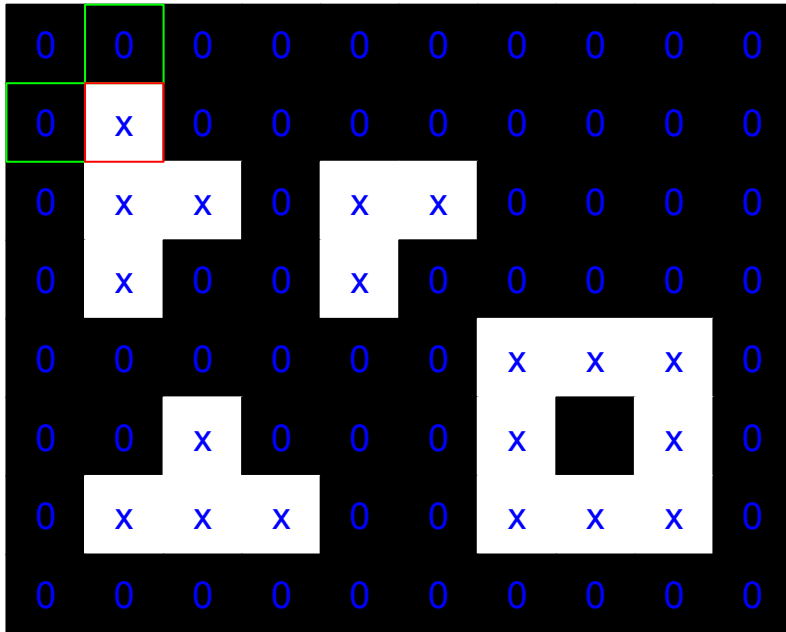


# Componentes conectados



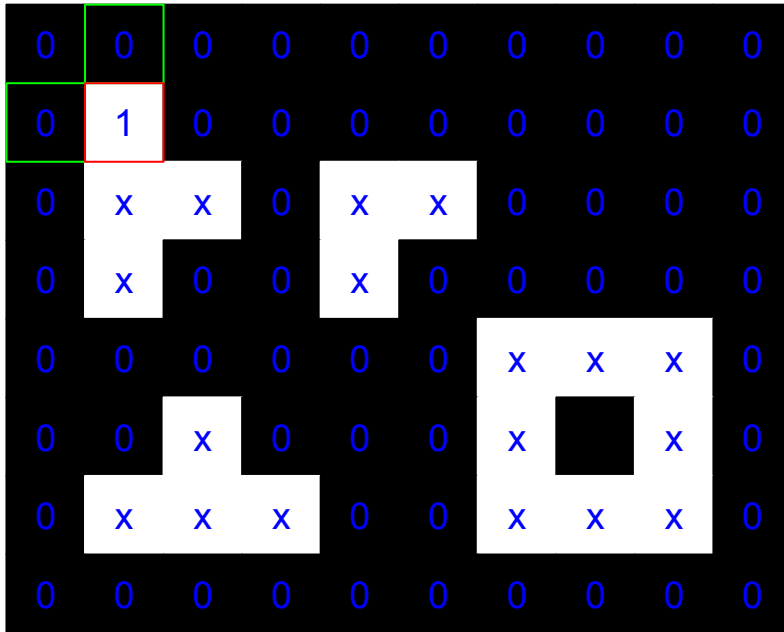
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



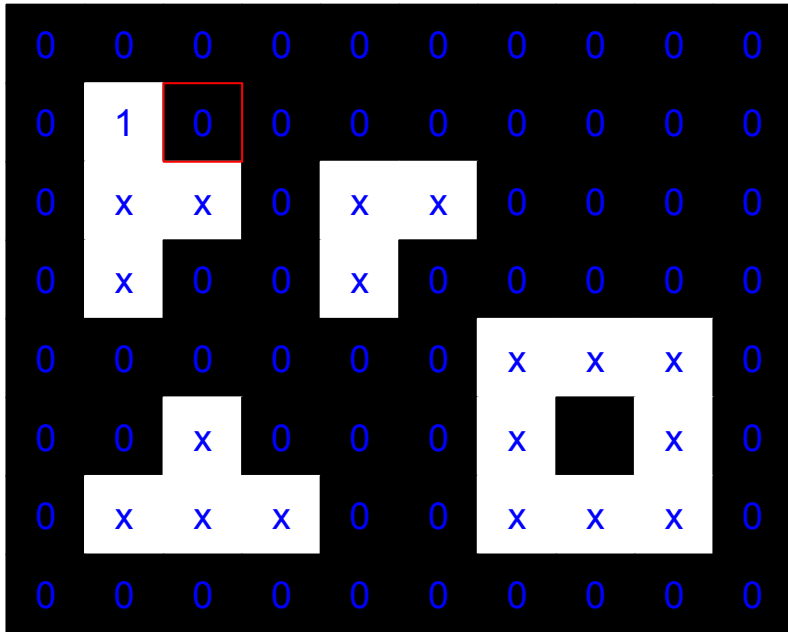
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



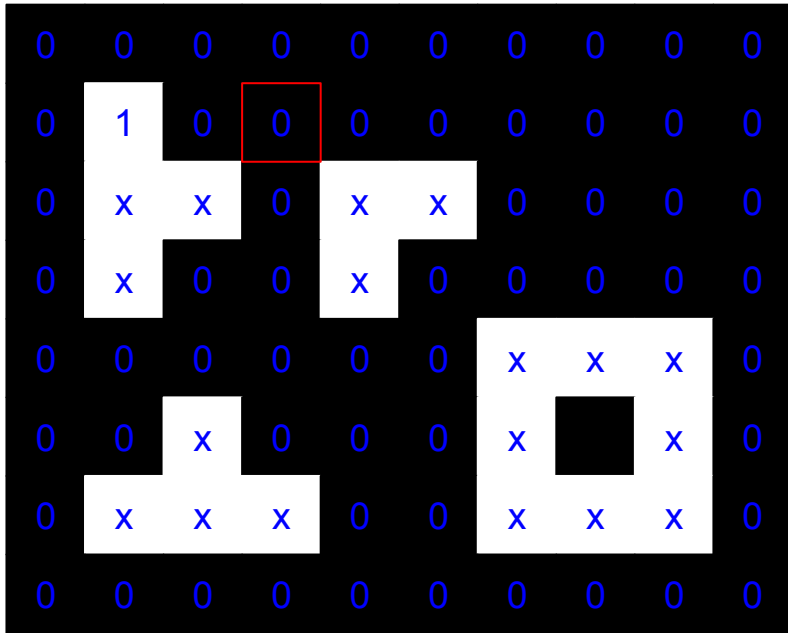
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

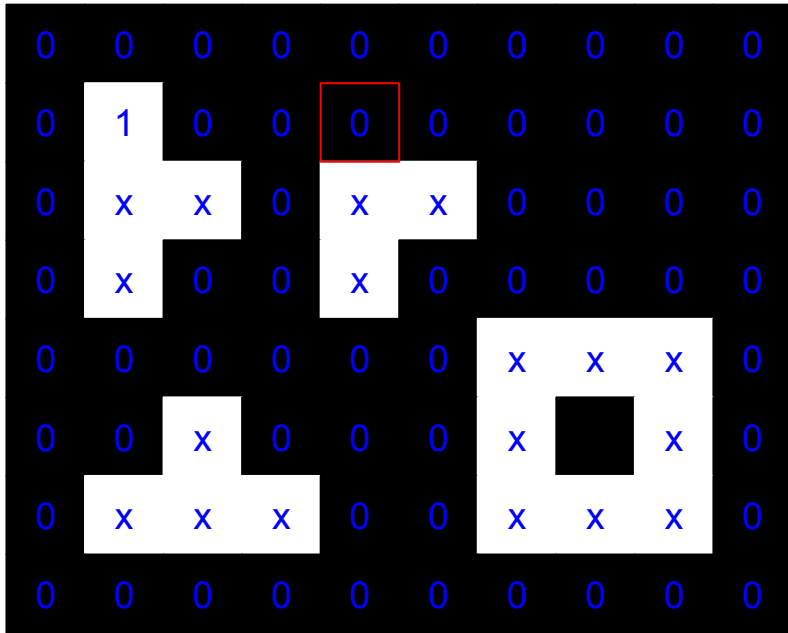
# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

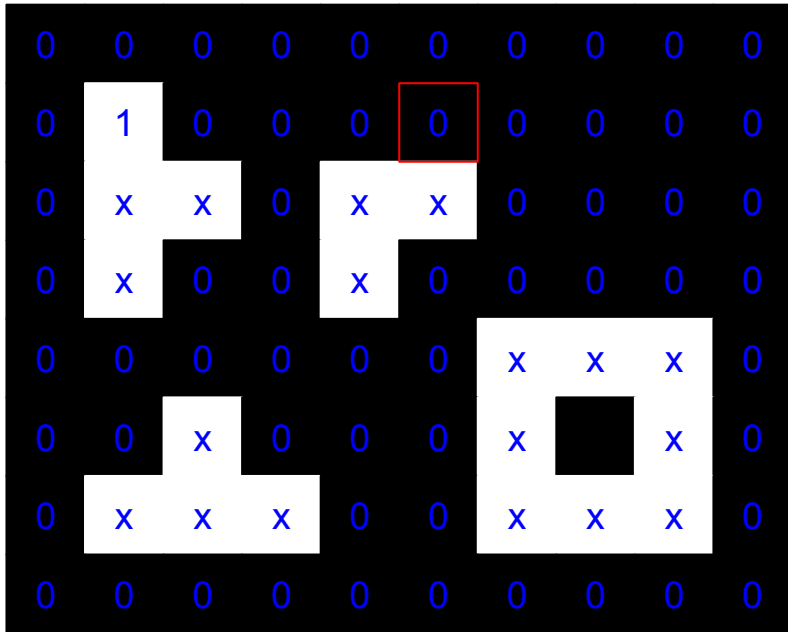


# Componentes conectados



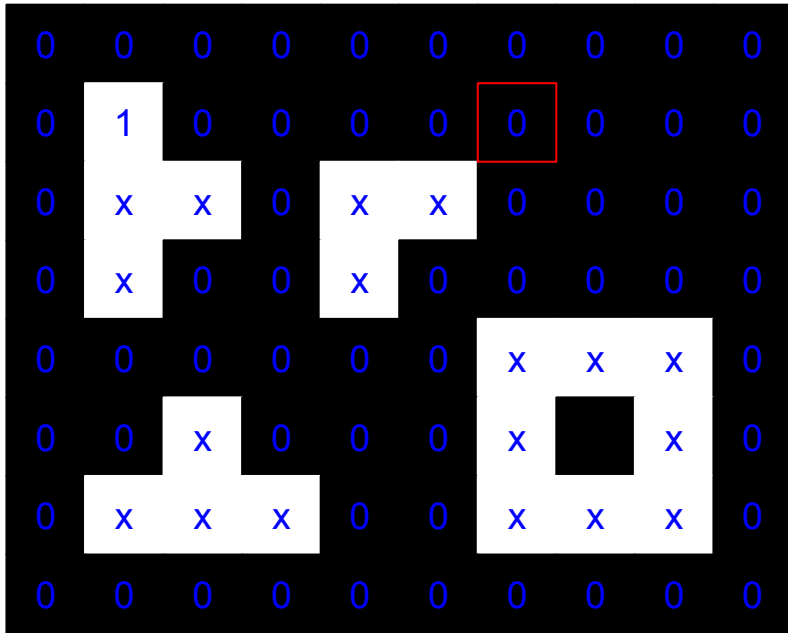
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



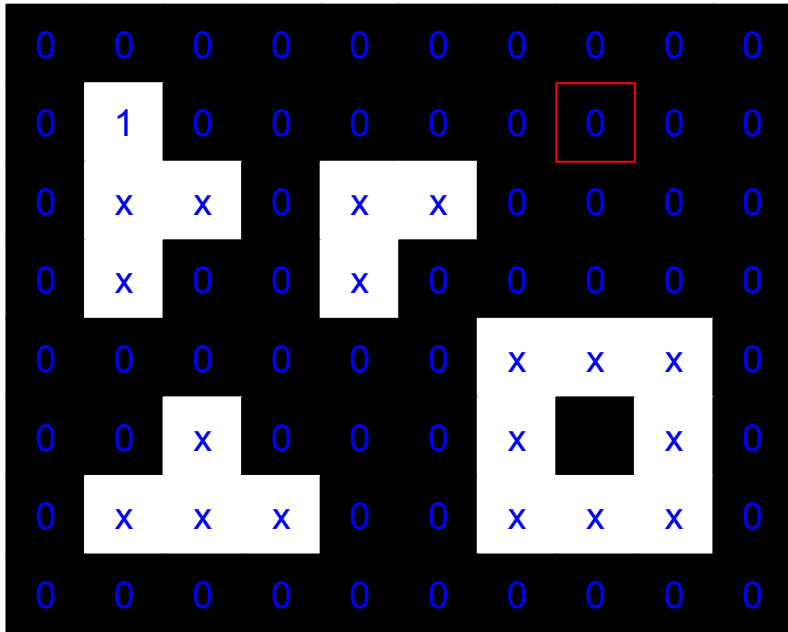
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



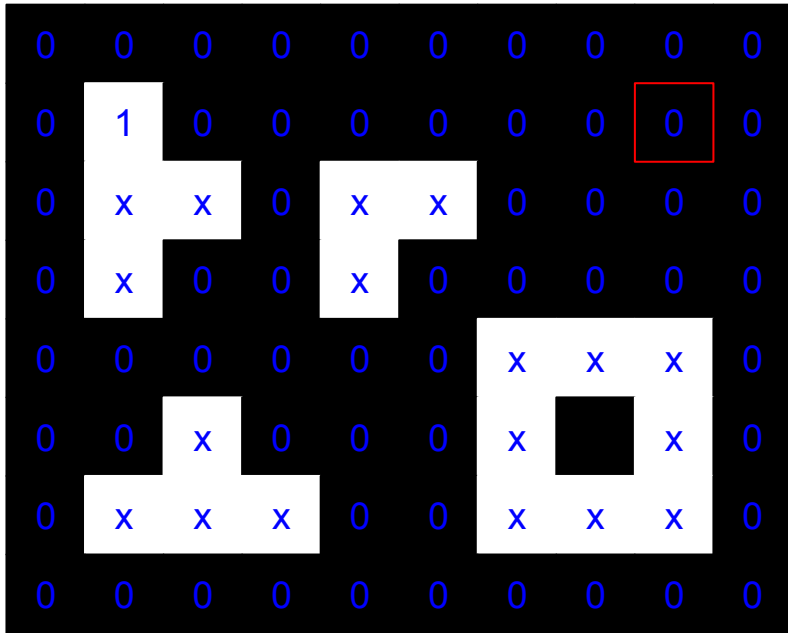
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



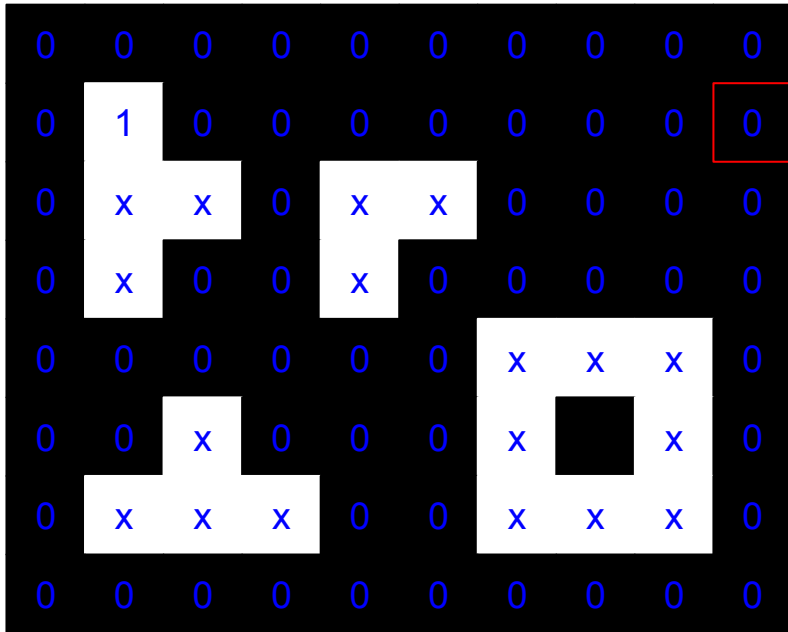
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



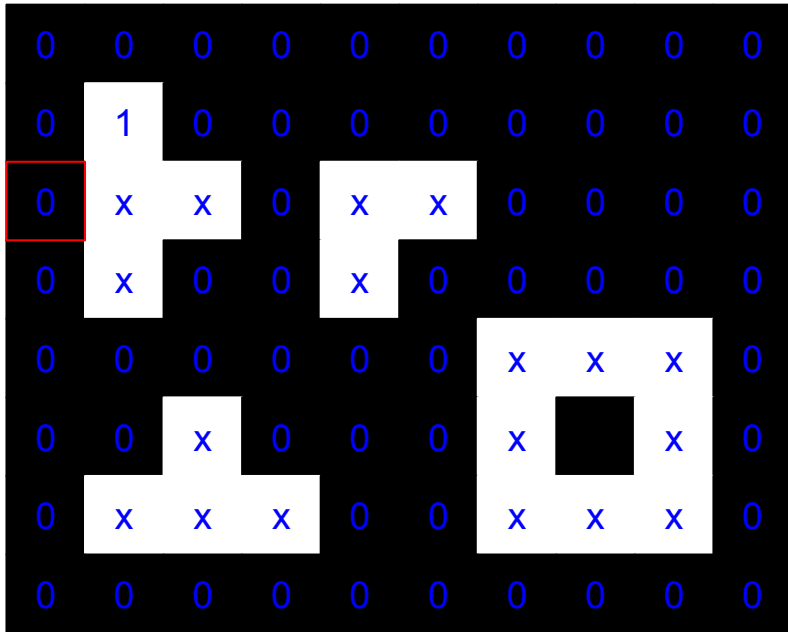
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



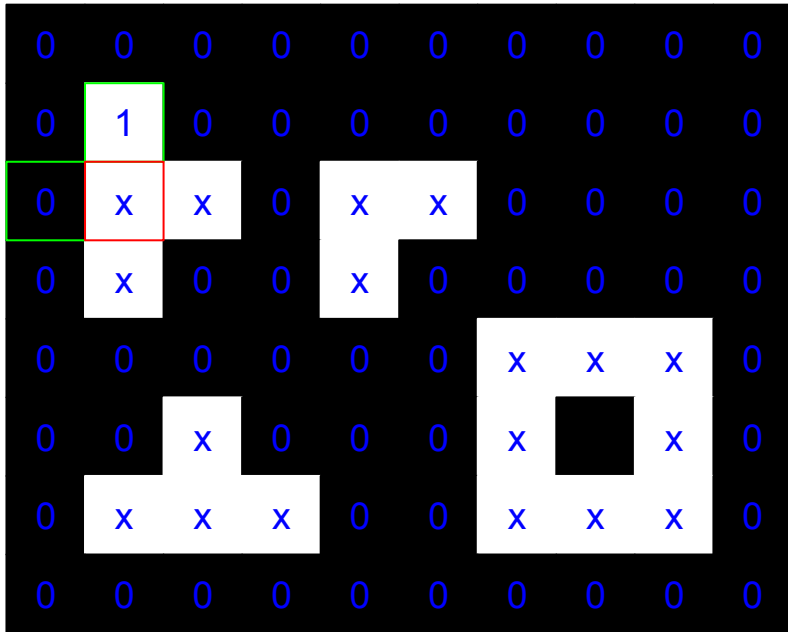
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

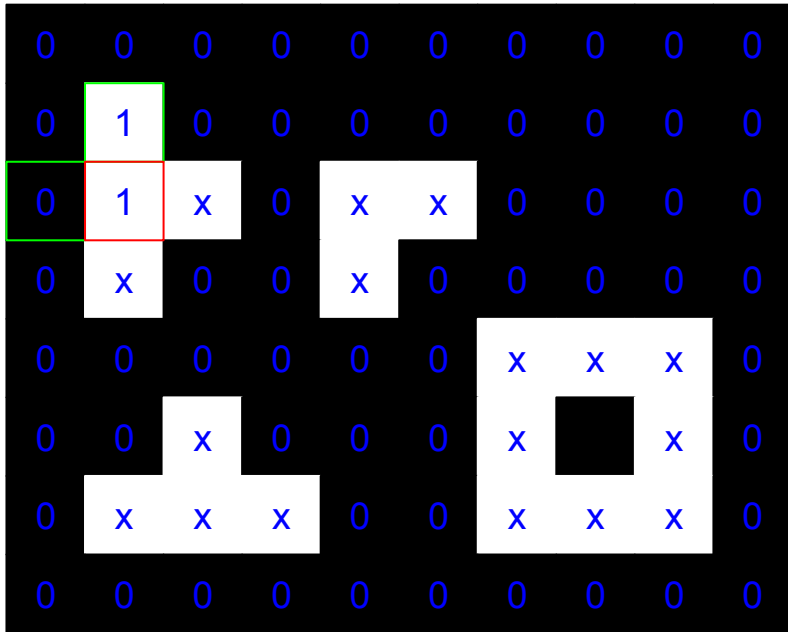
# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

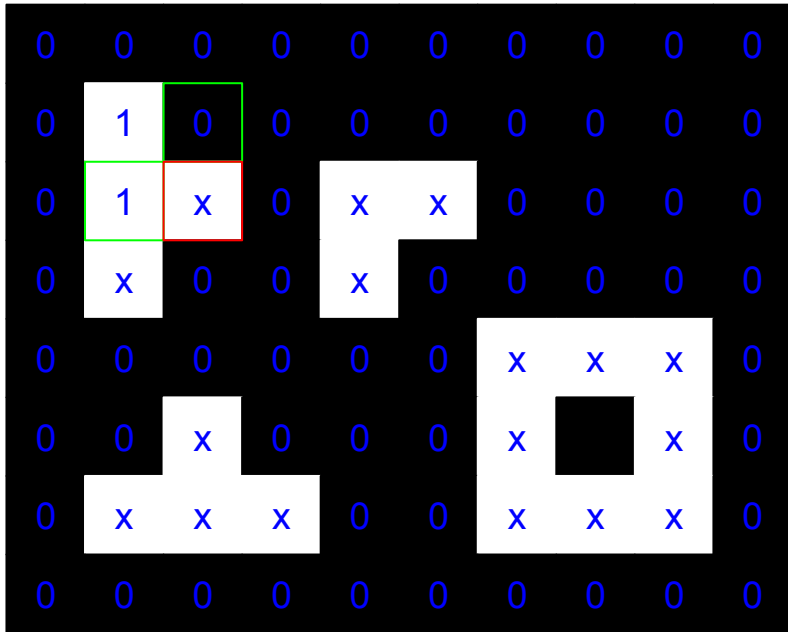


# Componentes conectados



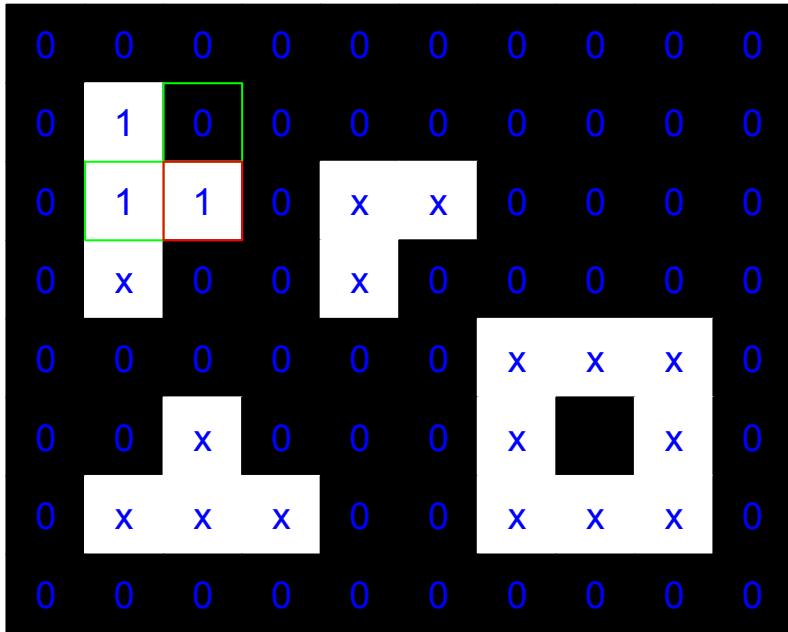
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



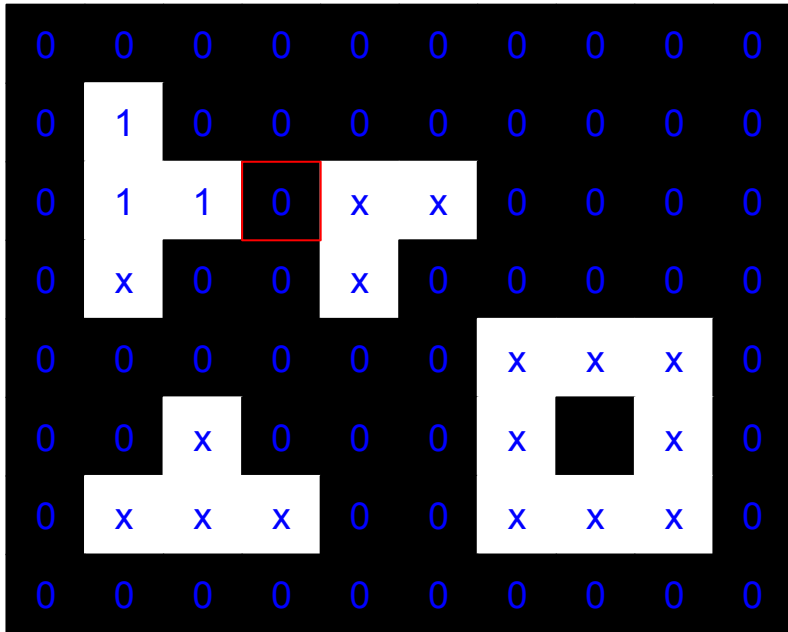
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



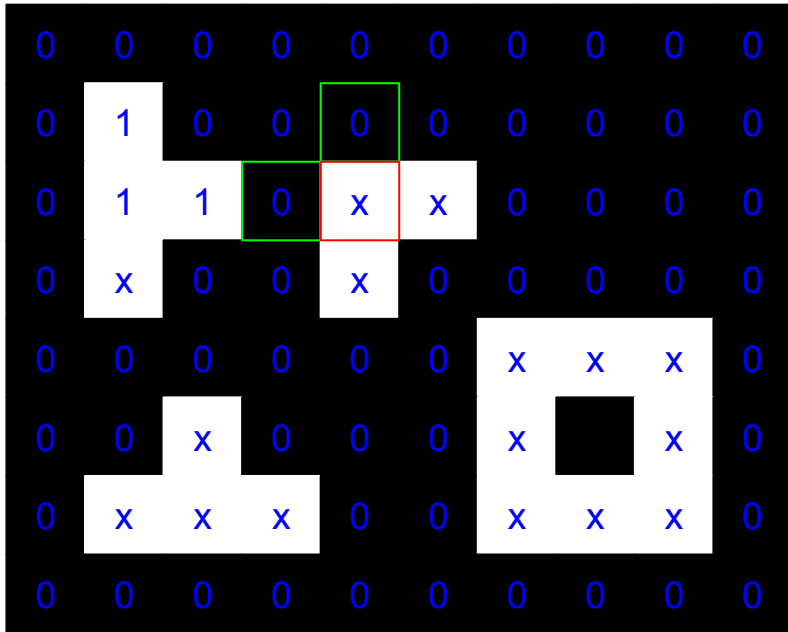
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



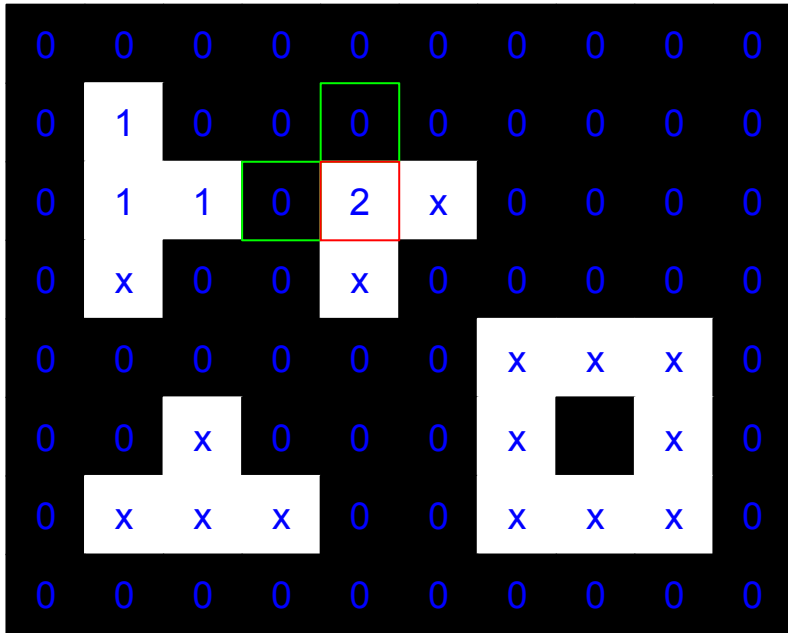
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



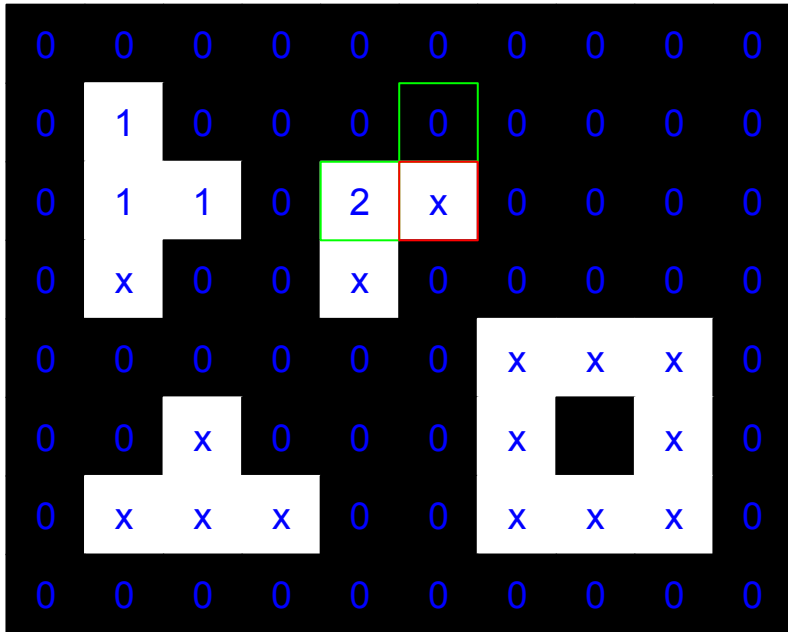
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



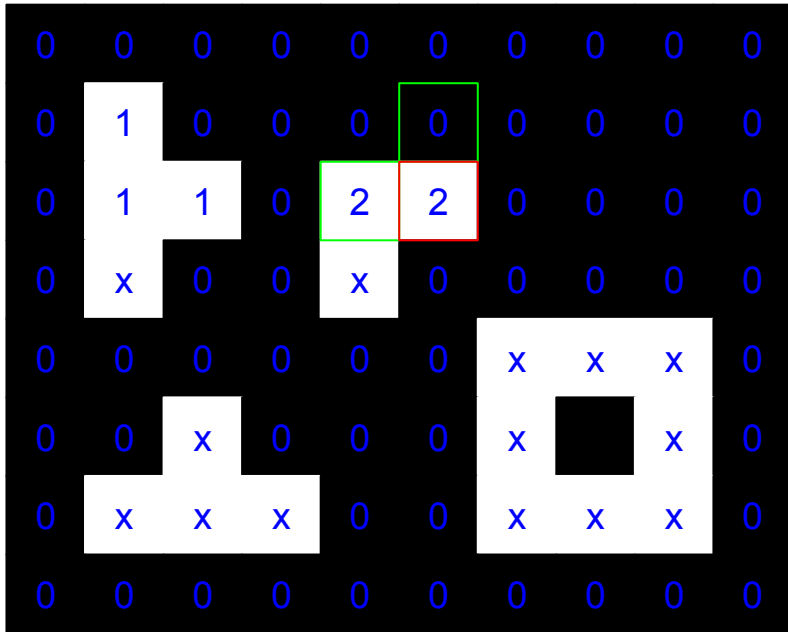
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

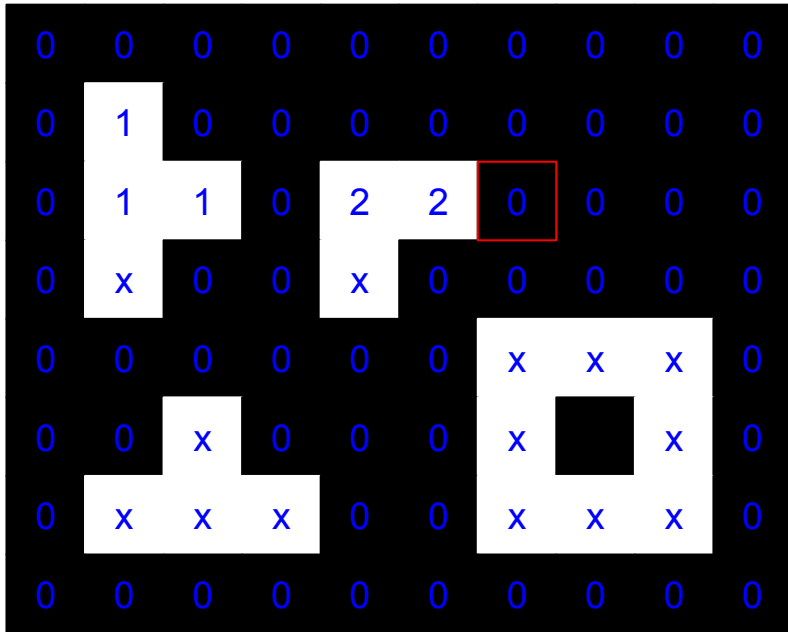
# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

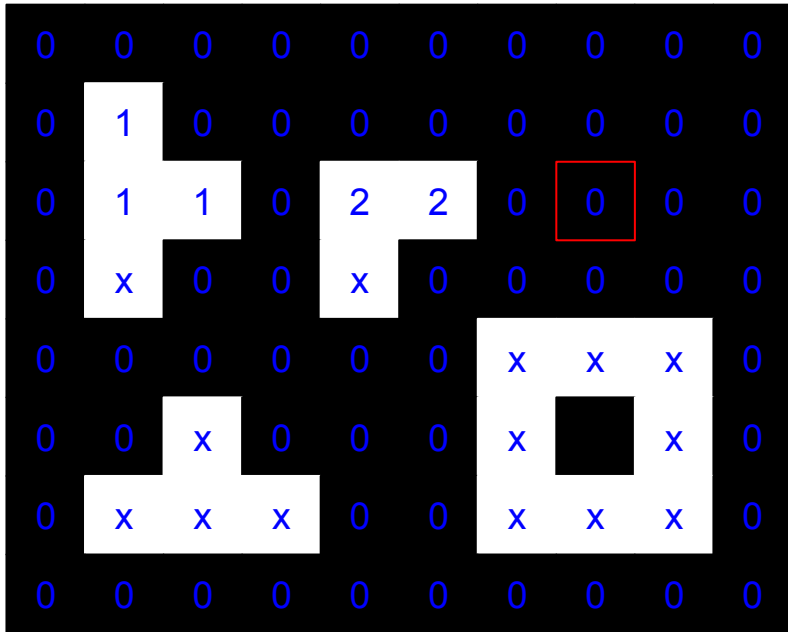


# Componentes conectados



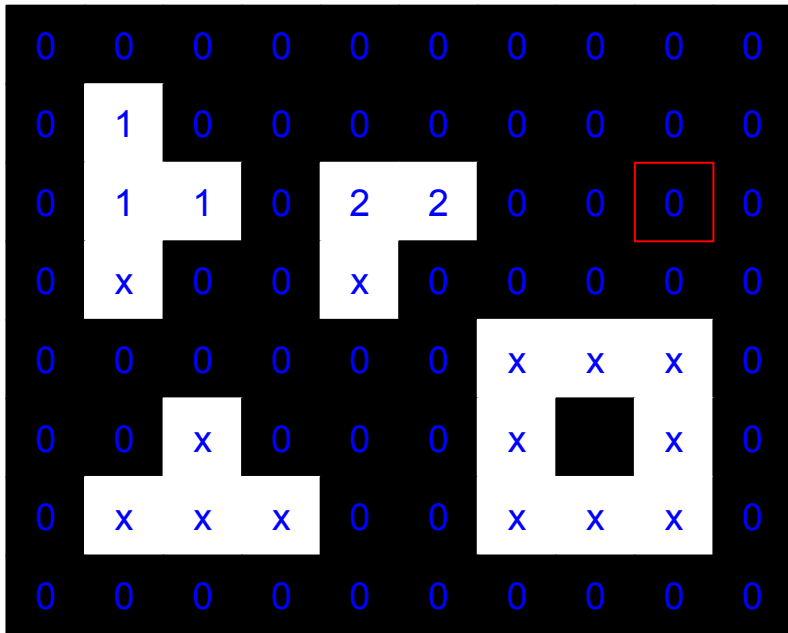
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



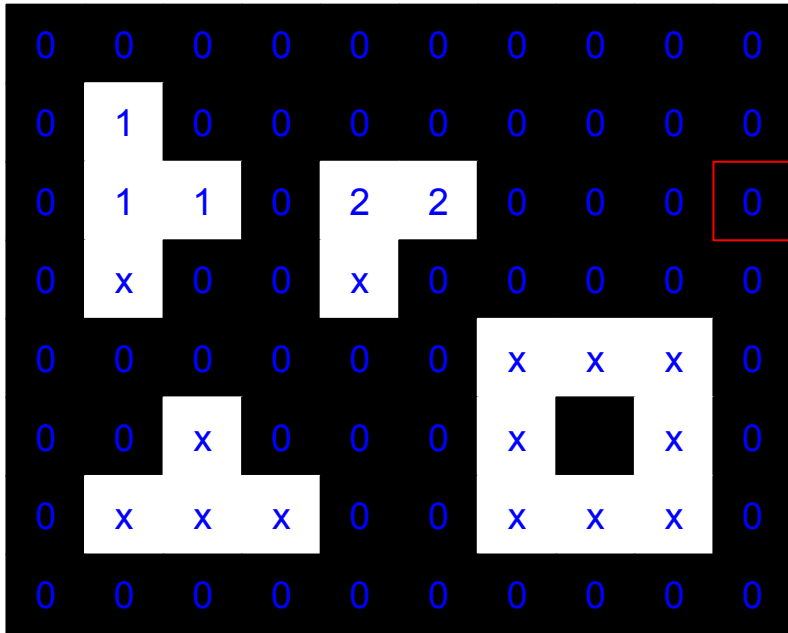
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



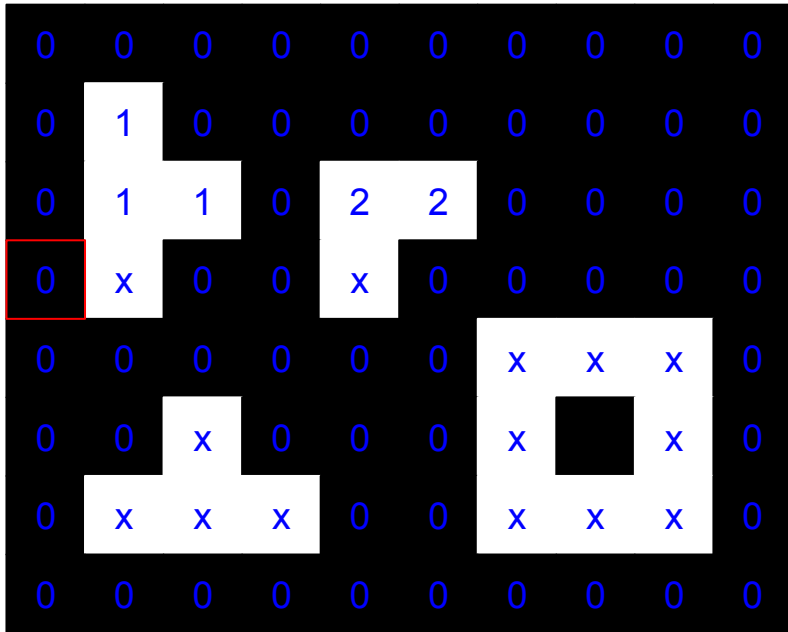
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



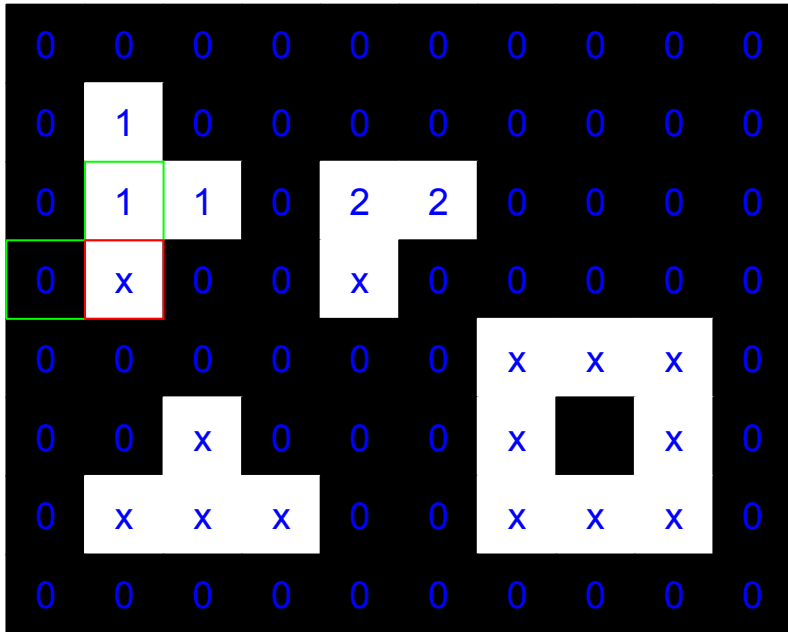
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



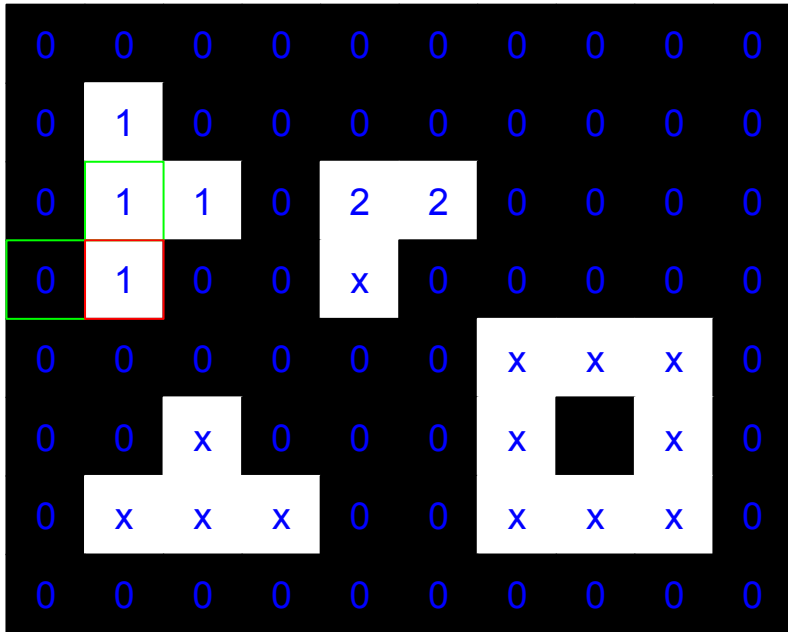
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



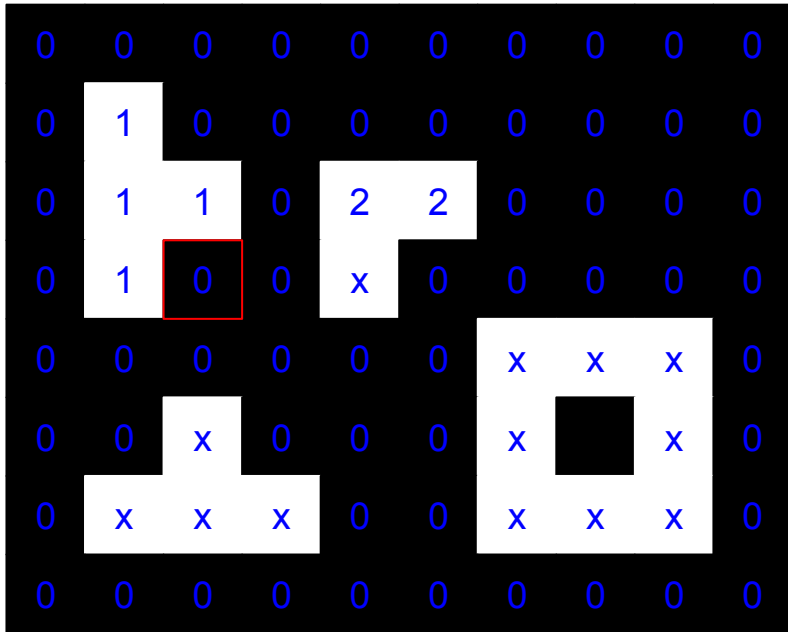
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

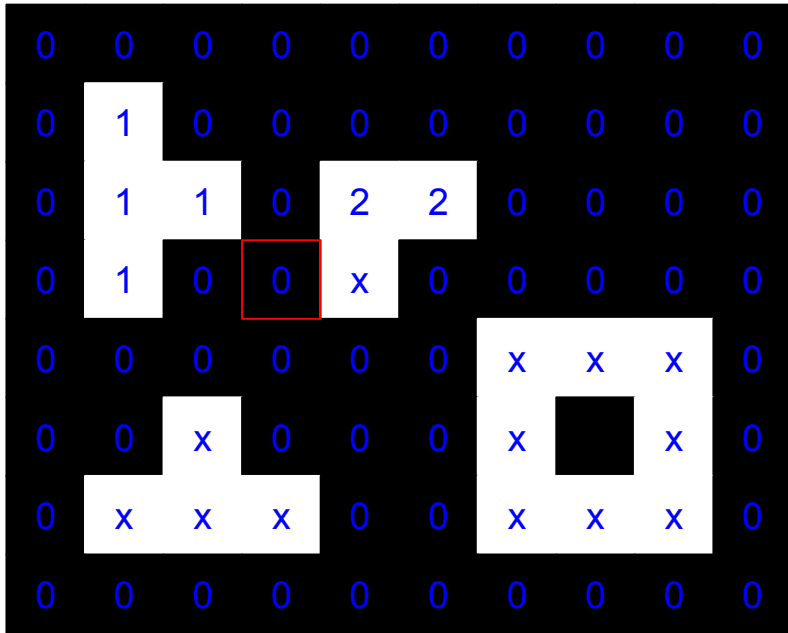
# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

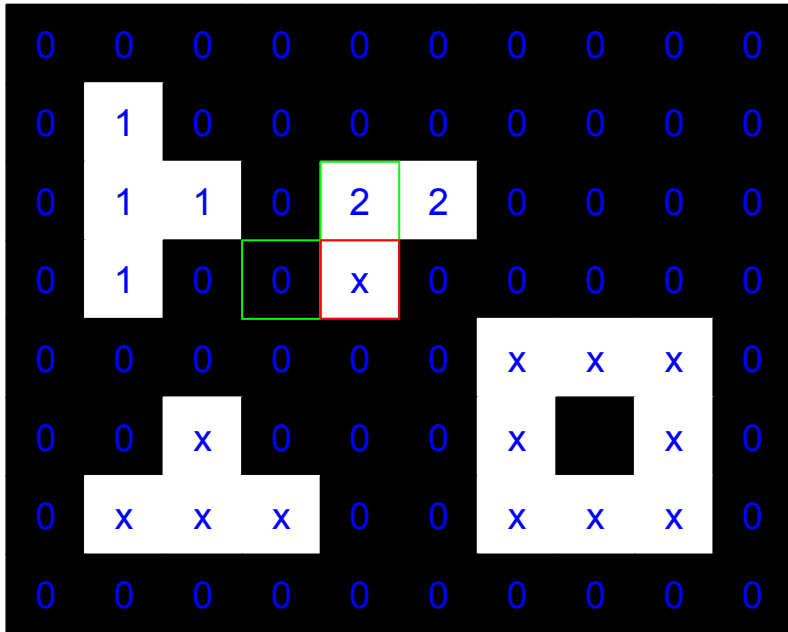


# Componentes conectados



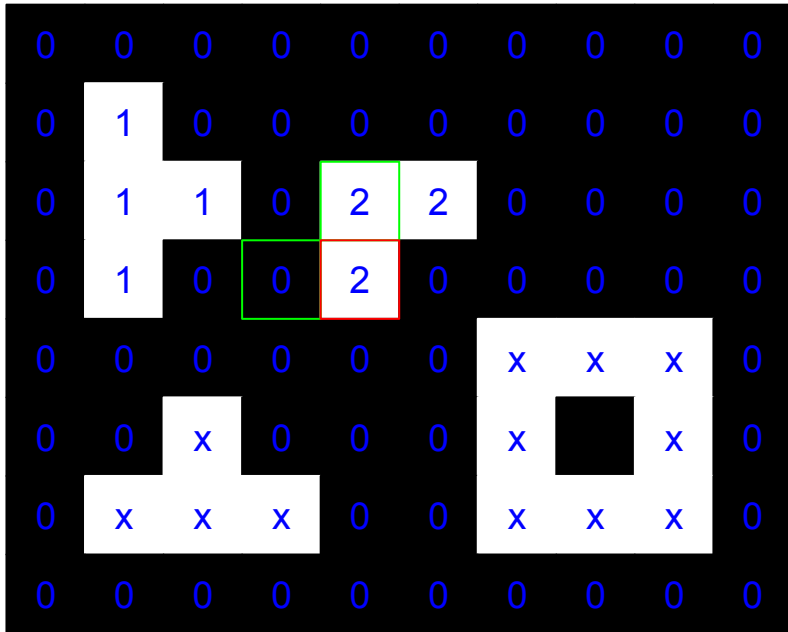
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



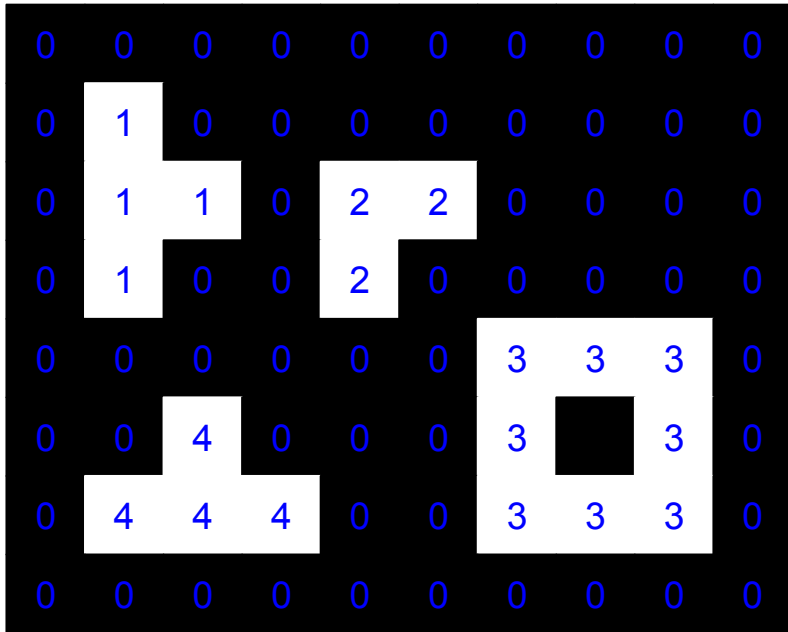
- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados



- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.

# Componentes conectados

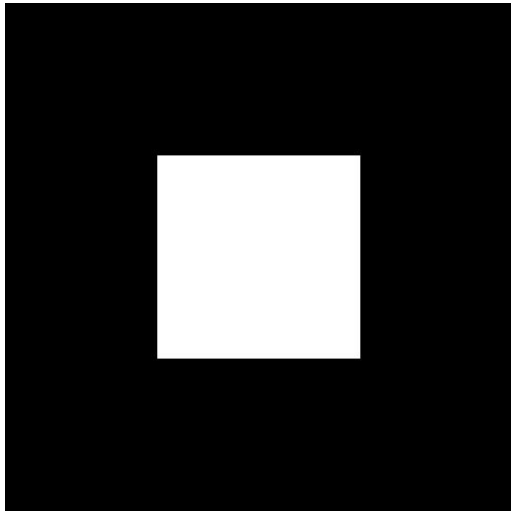


- ❑ Percorrer toda a imagem e verificar as conexões entre os pixels brancos.
- ❑ Definir um rótulo para cada grupo de pixels brancos conectados.
- ❑ 4 rótulos distintos.

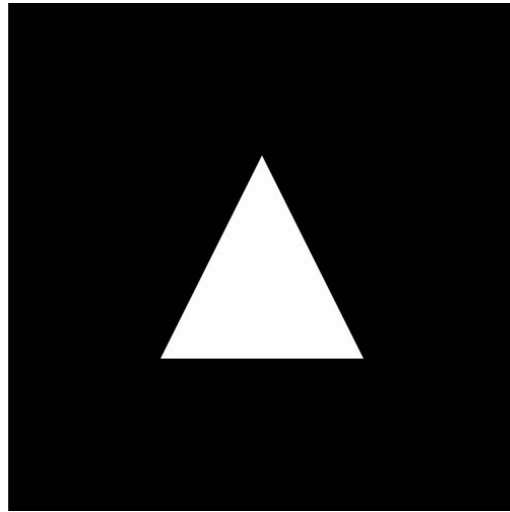
# Extração de características

# Características dimensionais

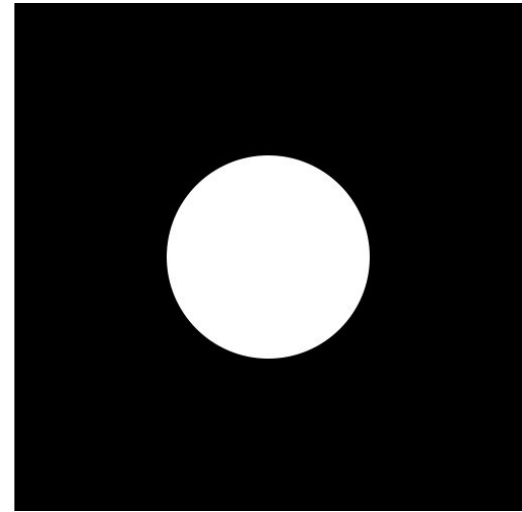
## □ Área do objeto



40.000 pixels



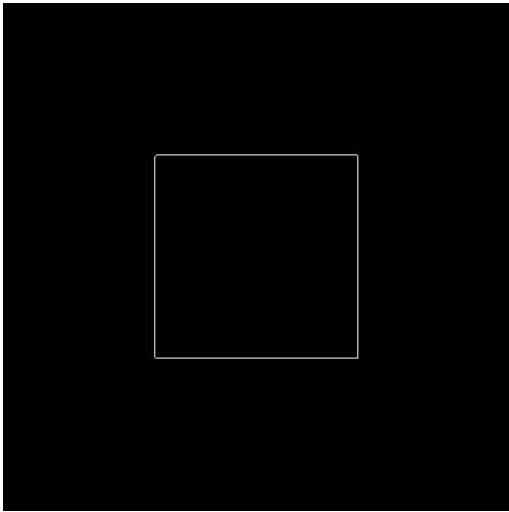
20.000 pixels



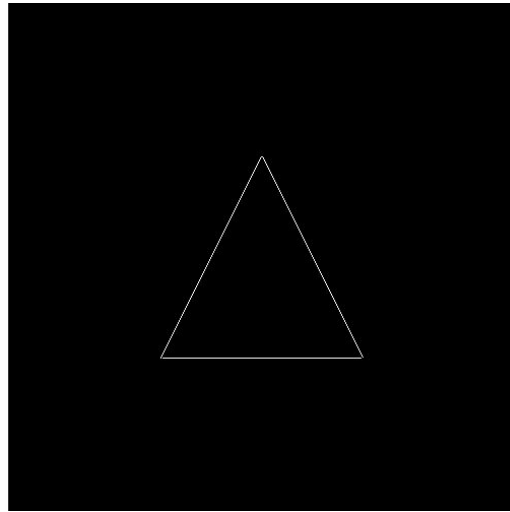
31.432 pixels

# Características dimensionais

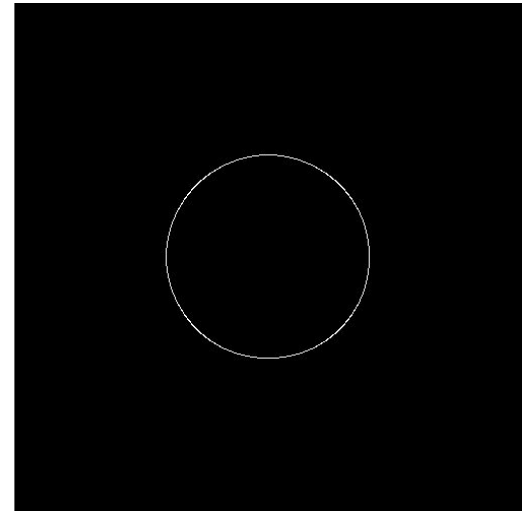
## □ Perímetro do objeto



796 pixels



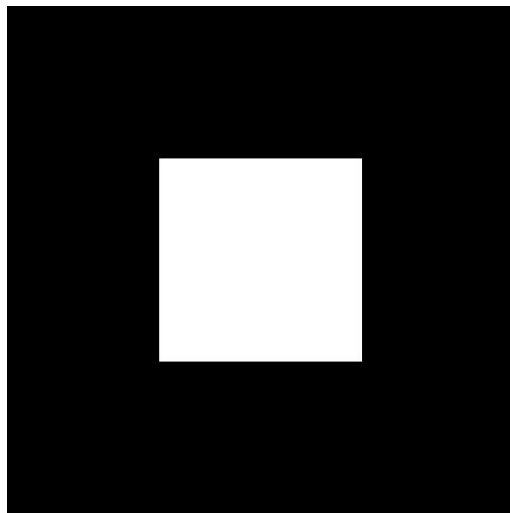
593 pixels



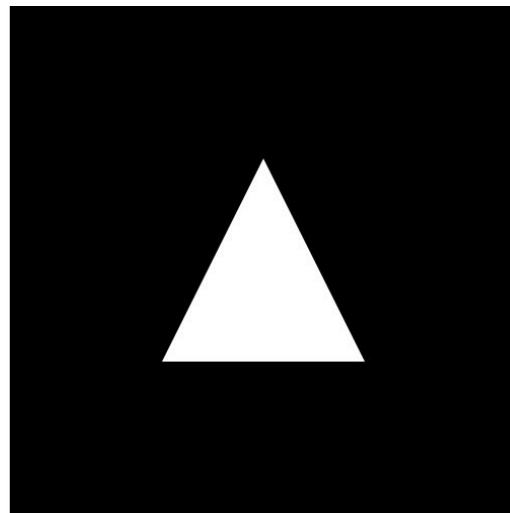
660 pixels

# Características topológicas

## □ Número de vértices



4 vértices

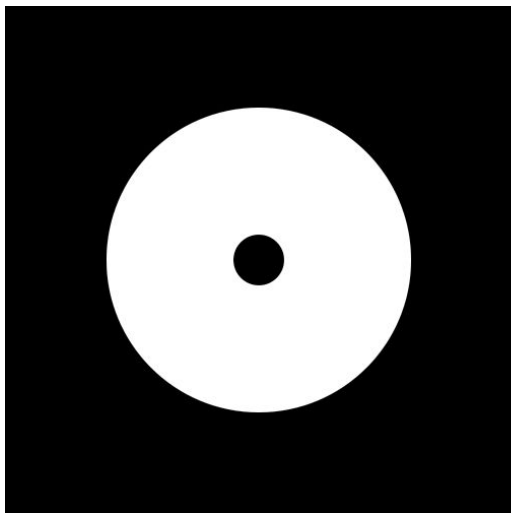


3 vértices

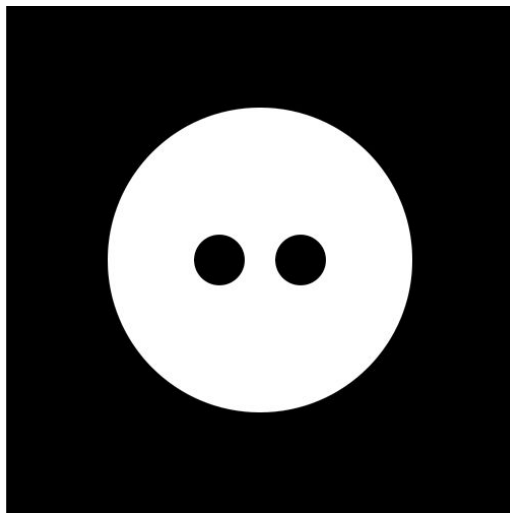


# Características topológicas

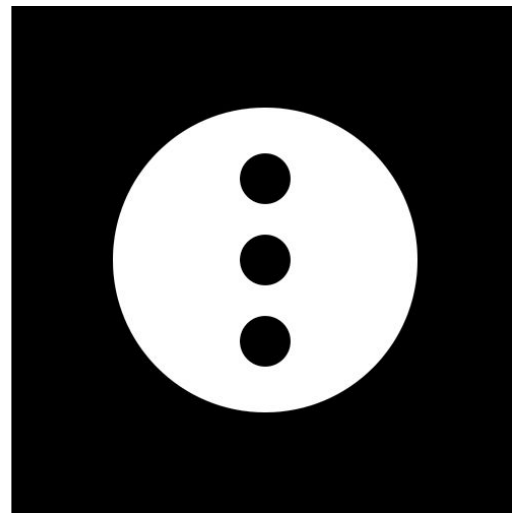
## □ Número de furos



1



2



3

# Características topológicas

- ❑ Número de furos
- ❑ Indústria utiliza sistemas de Visão Computacional para inspecionar se peças metálicas foram cortadas corretamente e não possuem falhas.



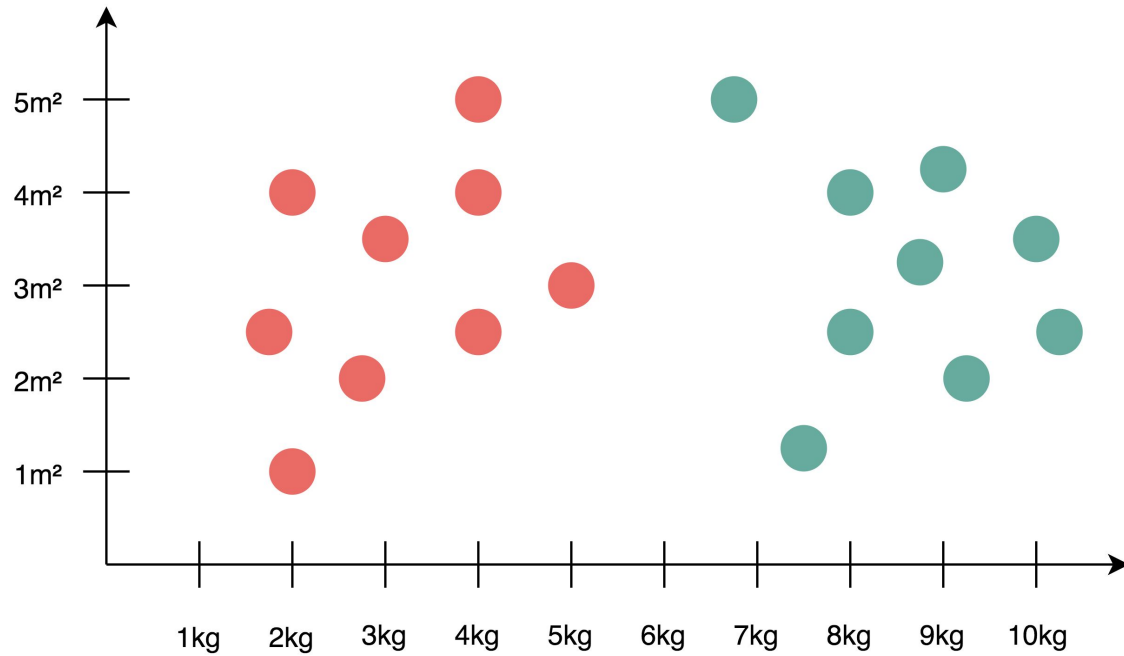
# Reconhecimento de padrões

# Reconhecimento de padrões

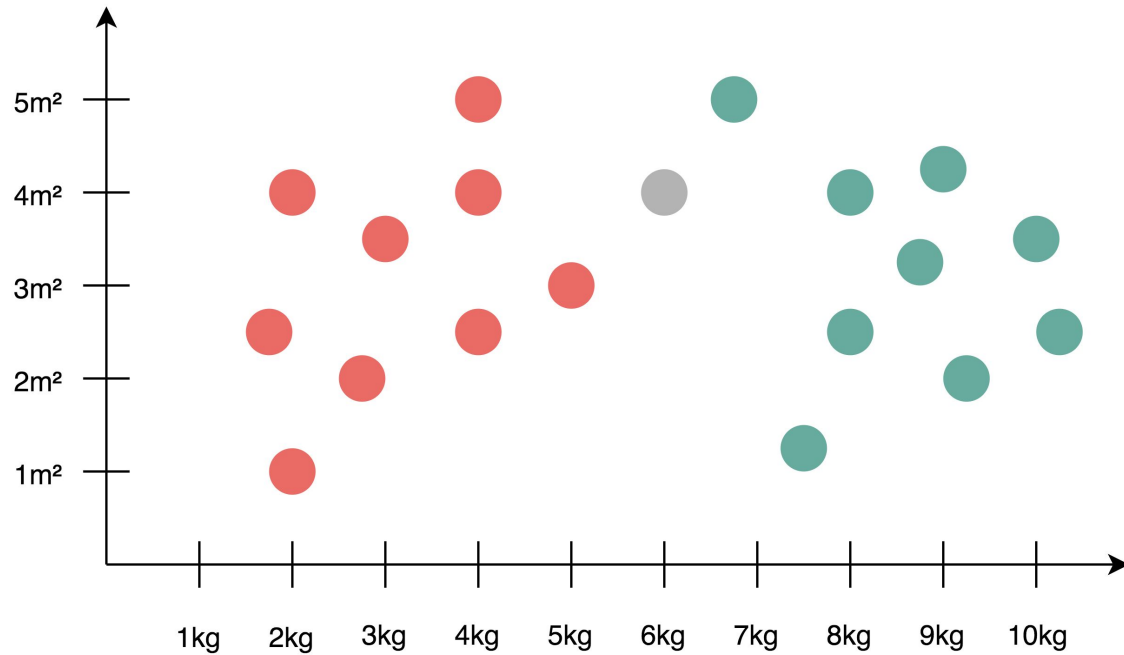
- ❑ O Reconhecimento de Padrões é o campo da ciência que estuda técnicas para descrever padrões de objetos a fim de classificá-los.
- ❑ Nos sistemas baseados em Visão Computacional, as técnicas de reconhecimento de padrões são essenciais, pois possibilitam a classificação automática de um objeto de interesse.

# Aprendizagem Supervisionada

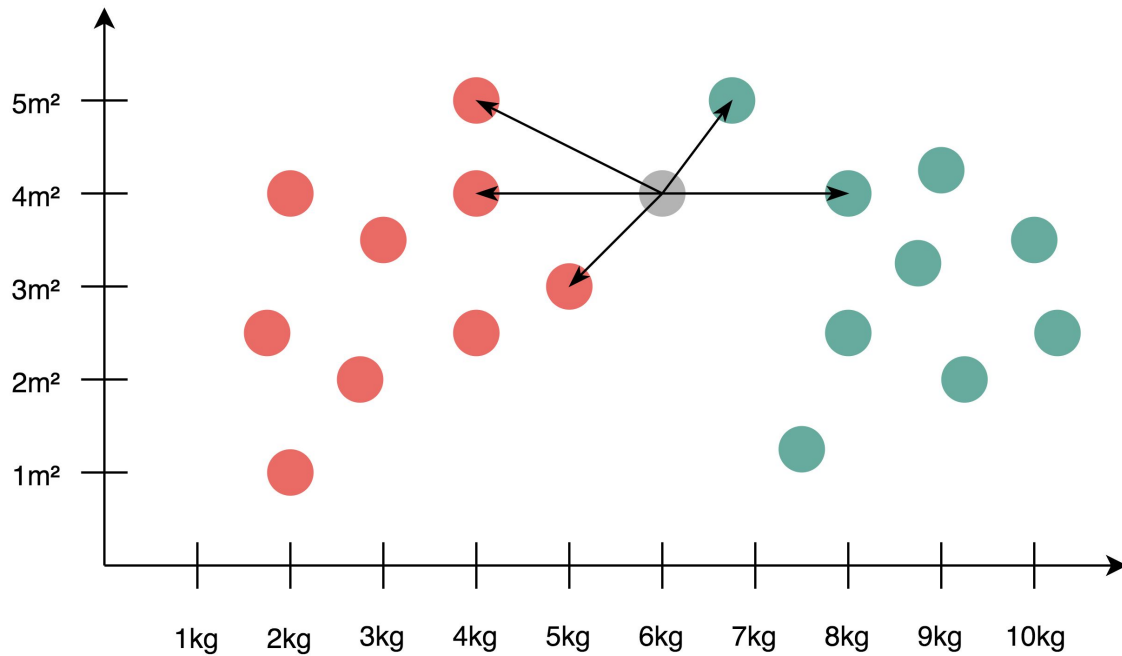
# Classificador K-NN



# Classificador K-NN



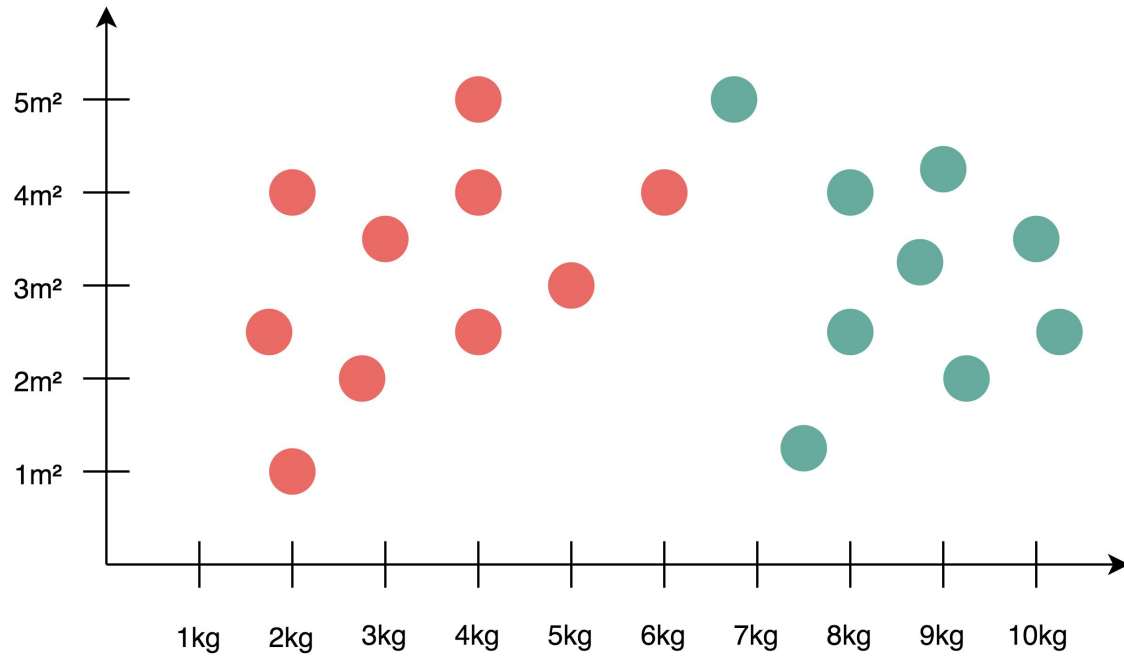
# Classificador K-NN



$K = 5$

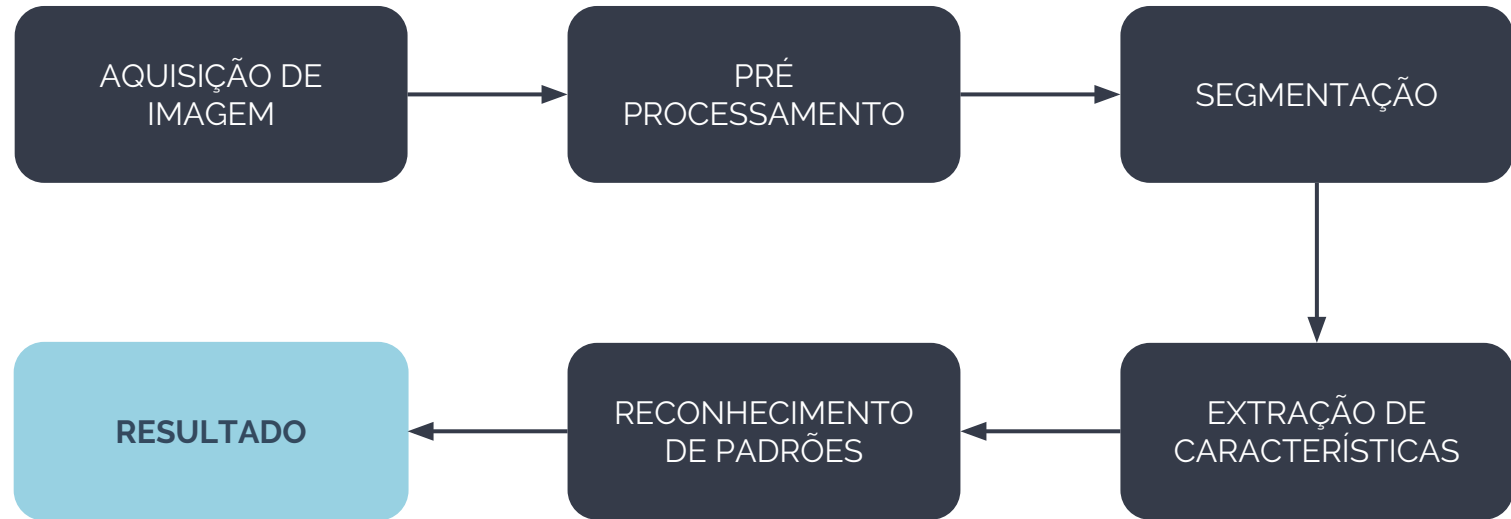


# Classificador K-NN



$K = 5$

# Revisando



# Ferramentas para desenvolver sistemas de Visão Computacional

# Google & Microsoft Vision API

- ❑ **Serviços em nuvem com API para C#, Java, Python.**
  - ❑ Descrever e rotular imagem.
  - ❑ Detectar conteúdo adulto.
  - ❑ Detectar faces.
  - ❑ Análisar sentimento em imagem.

# Microsoft Vision API




FEATURE NAME:	VALUE
Description	{ "tags": [ "train", "platform", "station", "building", "indoor", "subway", "track", "walking", "waiting", "pulling", "board", "people", "man", "luggage", "standing", "holding", "large", "woman", "yellow", "suitcase" ], "captions": [ { "text": "people waiting at a train station", "confidence": 0.833099365 } ] }
Tags	[ { "name": "train", "confidence": 0.9975446 }, { "name": "platform", "confidence": 0.995543063 }, { "name": "station", "confidence": 0.9798007 }, { "name": "indoor", "confidence": 0.927719653 }, { "name": "subway", "confidence": 0.838939846 }, { "name": "pulling", "confidence": 0.431715637 } ]
Image format	"Jpeg"

# Microsoft Vision API




Black and white	false
Adult content	false
Adult score	0.0147124995
Racy	false
Racy score	0.0162802152
Categories	[ { "name": "trans_trainstation", "score": 0.98828125 } ]
Faces	[]
Dominant color	■ "Black"

# Google Vision API


Labels	Web	Document	Properties	Safe Search	JSON																
																					
senai.jpg																					
				<table><tbody><tr><td>Training</td><td>74%</td></tr><tr><td>Classroom</td><td>74%</td></tr><tr><td>Institution</td><td>72%</td></tr><tr><td>Education</td><td>65%</td></tr><tr><td>Communication</td><td>62%</td></tr><tr><td>Course</td><td>62%</td></tr><tr><td>Class</td><td>53%</td></tr><tr><td>Student</td><td>51%</td></tr></tbody></table>	Training	74%	Classroom	74%	Institution	72%	Education	65%	Communication	62%	Course	62%	Class	53%	Student	51%	
Training	74%																				
Classroom	74%																				
Institution	72%																				
Education	65%																				
Communication	62%																				
Course	62%																				
Class	53%																				
Student	51%																				

# Google Vision API

Labels	Web	Document	Properties	Safe Search	JSON
		 <p>senai.jpg</p>	<ul style="list-style-type: none"><li>J E S E N</li><li>+Block 3</li><li>E S E N A I</li><li>+Block 4</li><li>S E N A I</li><li>+Block 5</li><li>S E N A I</li></ul>		



# Google Vision API

Labels	Web	Document	Properties	Safe Search	JSON															
				<table><tbody><tr><td>Adult</td><td>■</td><td>Very Unlikely</td></tr><tr><td>Spoof</td><td>■</td><td>Very Unlikely</td></tr><tr><td>Medical</td><td>■</td><td>Very Unlikely</td></tr><tr><td>Violence</td><td>■</td><td>Very Unlikely</td></tr><tr><td>Racy</td><td>■ ■</td><td>Unlikely</td></tr></tbody></table> <p>Likelihood values are Unknown, Very Unlikely, Unlikely, Possible, Likely, and Very Likely</p>	Adult	■	Very Unlikely	Spoof	■	Very Unlikely	Medical	■	Very Unlikely	Violence	■	Very Unlikely	Racy	■ ■	Unlikely	
Adult	■	Very Unlikely																		
Spoof	■	Very Unlikely																		
Medical	■	Very Unlikely																		
Violence	■	Very Unlikely																		
Racy	■ ■	Unlikely																		
senai.jpg																				

# Câmera Intel REALSENSE SR300



The image shows a screenshot of the Intel Developer Zone website for the Intel RealSense SR300 camera. The page features a blue header with the Intel Software Developer Zone logo, a search bar, and navigation links for support, login, and language selection. The main content area has a dark blue background with the product name 'CÂMERA INTEL® REALSENSE™ SR300' in large white letters. Below the name is a descriptive paragraph and a 'Comece a usar' button. A 3D rendering of the camera is shown on the right, with the 'CREATIVE' logo on its side. The Intel RealSense Technology logo is at the bottom right. A footer contains icons and links for 'Para começar', 'Documentação', and 'Fórum'.

intel Software Developer Zone

Pesquisa em todo o conteúdo 🔍

🔗 Suporte

👤 Iniciar sessão

🌐 Português

☰ **TECNOLOGIA INTEL® REALSENSE™**  
Periféricos da câmera

**CÂMERA INTEL® REALSENSE™ SR300**

Uma câmera sensora de profundidade que usa metodologia de luz codificada para percepção de profundidade em curto alcance.

[Comece a usar](#)

[Compartilhar](#)

intel REALSENSE TECHNOLOGY

📖 Para começar

📄 Documentação

👥 Fórum

# Câmera Intel REALSENSE SR300

- ❑ **Kit de desenvolvedor Intel® RealSense™**
  - ❑ Ferramentas para reconhecimento facial.
  - ❑ Ferramentas para reconhecimento de gestos.
  - ❑ API para análise de sentimento na imagem.



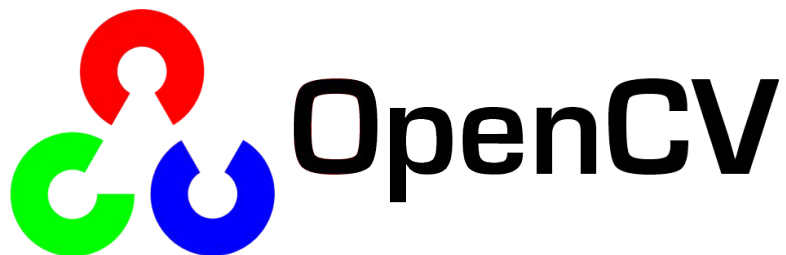
# MatLab

- ❑ Software proprietário, desenvolvido pela MathWorks Inc.
- ❑ Multiplataforma, com versões para Windows, MacOS e Linux.
- ❑ Foi desenvolvida nas linguagens de programação C e Java.
- ❑ É um software destinado a fazer cálculos com matrizes.
  - ❑ MATLAB = MATrix LABoratory

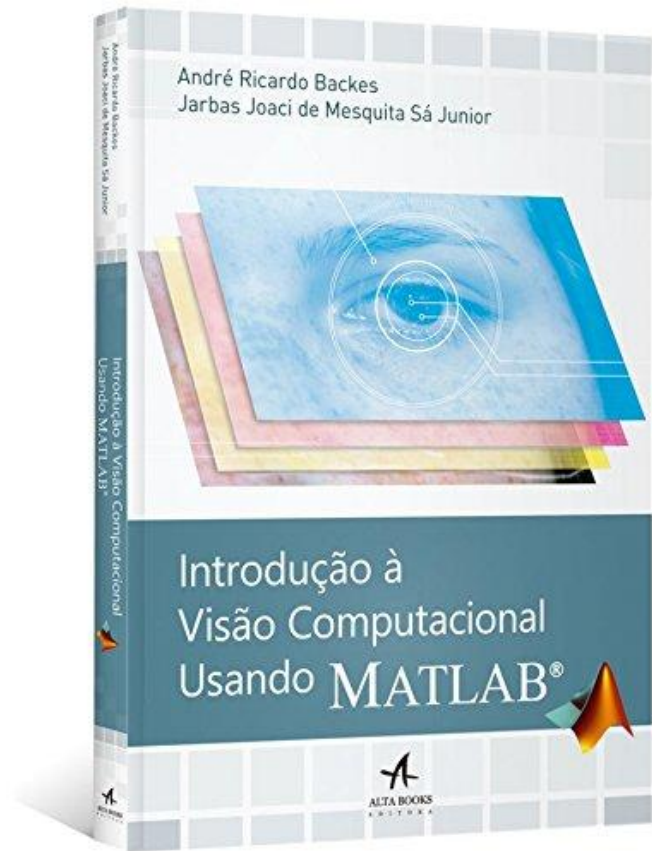


# OpenCV

- ❑ Originalmente, desenvolvida pela Intel, em 2000.
- ❑ Biblioteca multiplataforma, totalmente livre ao uso acadêmico e comercial.
- ❑ Foi desenvolvida nas linguagens de programação C/C++.
- ❑ Foi desenvolvida também para Java e Python.



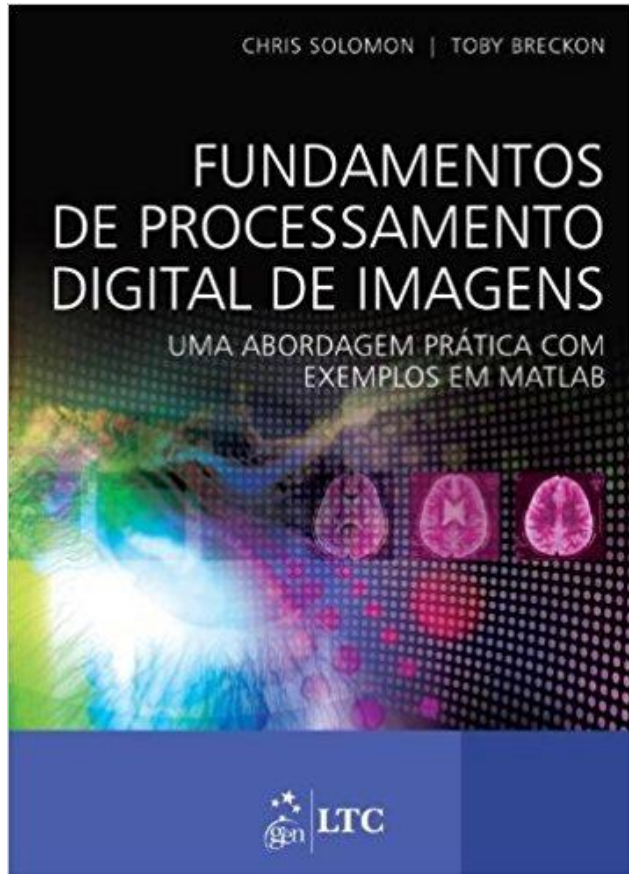
# Leituras recomendadas



# Introdução à Visão Computacional Usando MATLAB

André Ricardo Backes  
Jarbas Joaci de Mesquita Sá Junior

Editora Alta Books

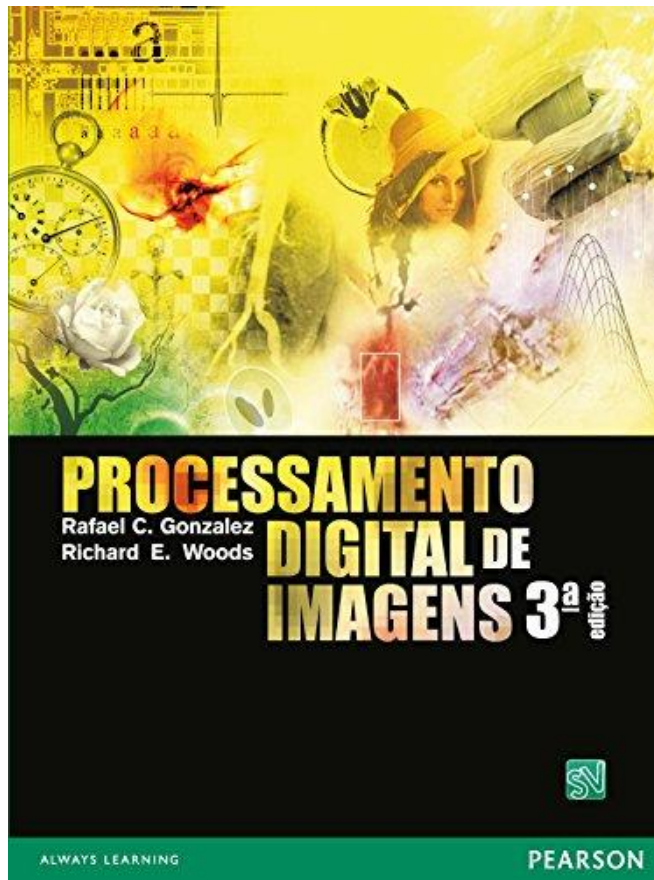


Fundamentos de Processamento  
Digital de Imagens: Uma Abordagem Prática  
com Exemplos em Matlab

Chris Solomon  
Toby Breckon

Editora LTC





## Processamento Digital de Imagens

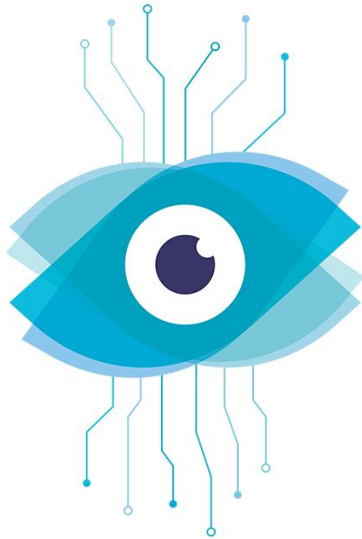
Rafael C. Gonzalez

Richard E. Woods

Editora Pearson

# Introdução à Visão Computacional

Uma abordagem prática com Python e OpenCV



 Casa do  
Código

FELIPE BARELLI

Introdução à Visão Computacional:  
Uma abordagem prática com Python e OpenCV

Felipe Barelli

Editora Casa do Código

# FIM

## Partiu sorteio!



**@felipecbarelli**

+55 27 99227-4710

felipecbarelli@outlook.com