

Prática 4 - Reconhecimento de Pessoas em Vídeos

Universidade Federal de São João del Rei

Curso de Ciência da Computação - Departamento de Computação

Rodrigo José Zonzin Esteves

212050002

10 de dezembro de 2023

Introdução

A Visão Computacional é uma disciplina interdisciplinar na interseção da Computação e Matemática. Seu propósito é capacitar sistemas artificiais a extrair informações de imagens. Essa área utiliza matemática, estatística e aprendizado de máquina para processar o conteúdo visual.

Técnicas como convolução, diferença de frames e algoritmos de inteligência artificial são fundamentais para o desenvolvimento da área. A convolução extrai características e identifica padrões em imagens, enquanto a diferença de frames rastreia movimento e detecta mudanças em vídeos. Algoritmos como o k-means segmenta e identifica agrupamentos em imagens.

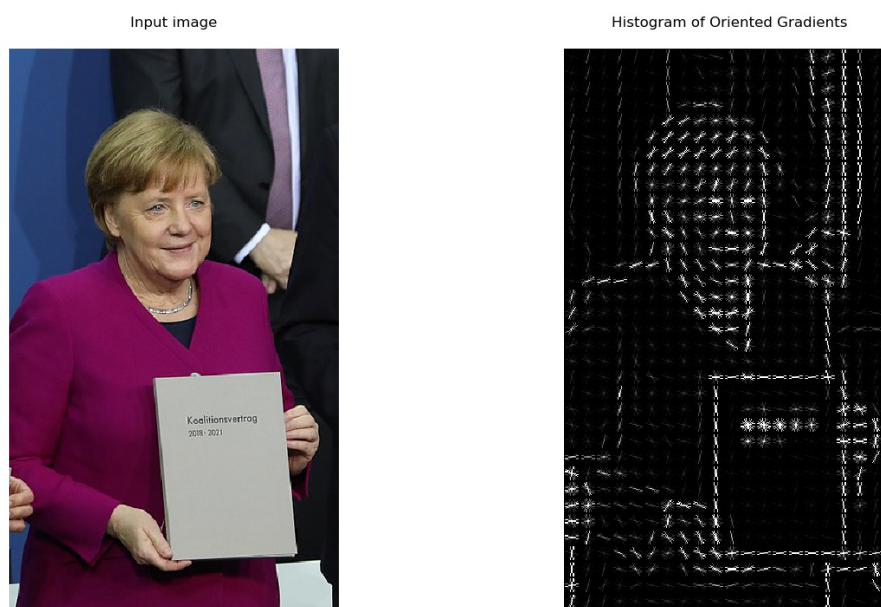
O reconhecimento de pessoas é essencial, especialmente em sistemas de segurança. Identificar indivíduos em imagens proporciona avaliar a quantidade de pessoas em um ambiente, quais ações estão tomando e abordagens mais complexas podem tentar prever os movimentos para que determinadas decisões sejam tomadas.

Além disso, a detecção de pedestres é essencial para o desenvolvimento de carros autônomos. A capacidade de identificar e rastrear outros veículos, bem como pedestres e obstáculos, é vital para garantir a segurança e a eficiência desses sistemas de transporte autônomos. A Visão Computacional possibilita a interpretação do ambiente ao redor do veículo, ajudando na tomada de decisões em tempo real para navegação e deslocamento seguros.

Metodologia

Histogram of Oriented Gradients

Esse algoritmo opera diretamente sobre os pixels em torno de cada ponto na imagem. Seu propósito é avaliar a diferença de intensidade entre o pixel atual e seus vizinhos, buscando identificar a direção na qual a imagem se torna mais escura. O processo consiste em desenhar setas indicativas dessa direção, criando gradientes que representam a transição de luz na imagem. Essencialmente, cada pixel é substituído por uma seta correspondente ao gradiente detectado. Esses gradientes tornam possível analisar o fluxo de luz na imagem, oferecendo informações cruciais para a execução de análises mais aprofundadas por parte de outros algoritmos.



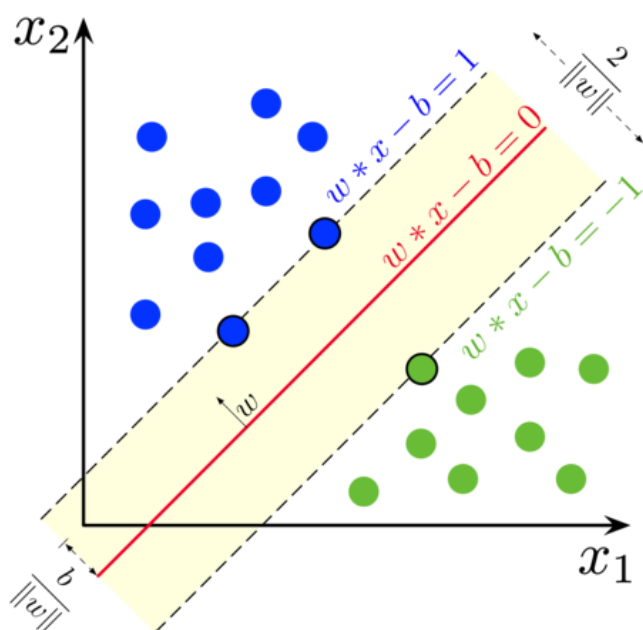
Support Vector Machine

Um dos métodos consagrados na ciência da computação para aprendizado supervisionado é a Máquina de Vetores de Suporte (SVM, do inglês Support-Vector Machine). Esta abordagem é utilizada para análise de dados, reconhecimento de padrões e desempenha um papel fundamental na classificação e análise de regressão. A natureza do SVM reside na sua capacidade de lidar com conjuntos de dados, prever a qual classe pertence cada entrada e atuar como um classificador linear binário não probabilístico.

Ao receber um conjunto de dados como entrada, o SVM realiza previsões sobre a categoria à qual cada entrada pertence. Isto é feito mediante a construção de um modelo a partir de exemplos de treinamento, cada um associado a uma de duas categorias. O algoritmo de treinamento do SVM cria um modelo que pode classificar novos exemplos dentro dessas categorias. Este modelo é representado visualmente como pontos no espaço, sendo mapeados de forma a garantir que os exemplos de cada categoria sejam separados por um espaço claro, maximizando a margem de separação entre elas.

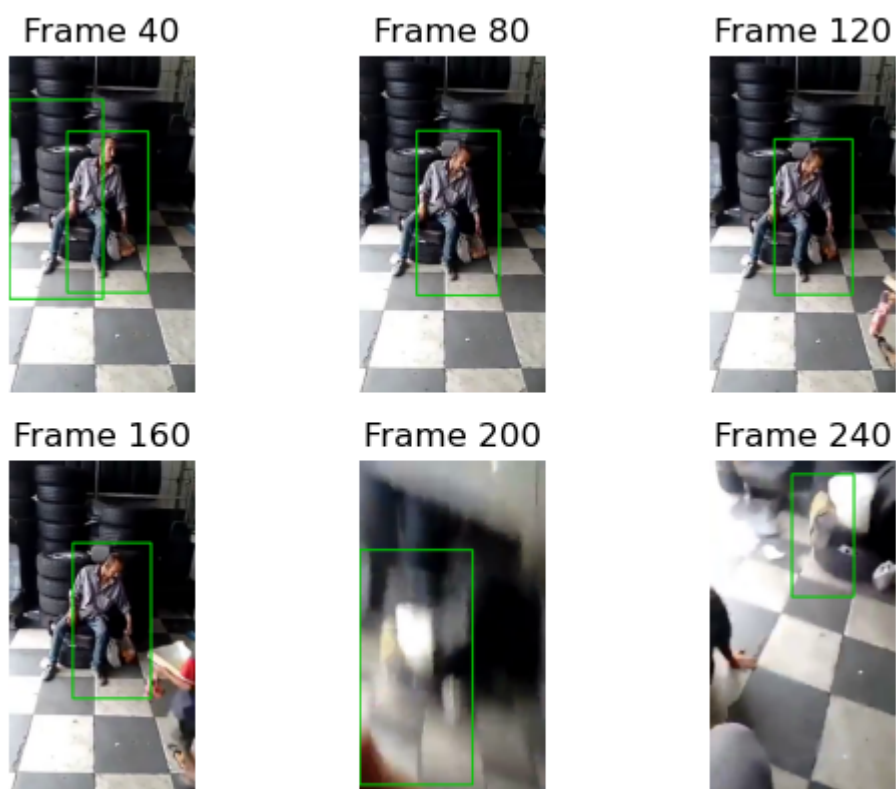
A ideia central é mapear os exemplos em um espaço multidimensional de modo que diferentes categorias sejam divididas por um hiperplano. Dessa forma, novos exemplos

podem ser mapeados nesse mesmo espaço e serem previstos como pertencentes a uma categoria específica baseada em qual lado do espaço eles são alocados. Este método se destaca por sua capacidade de lidar eficazmente com conjuntos de dados complexos e encontrar fronteiras de decisão ótimas, maximizando a generalização para novos dados.



Resultados

Vídeo 1 - Um vídeo viral no Twitter/X



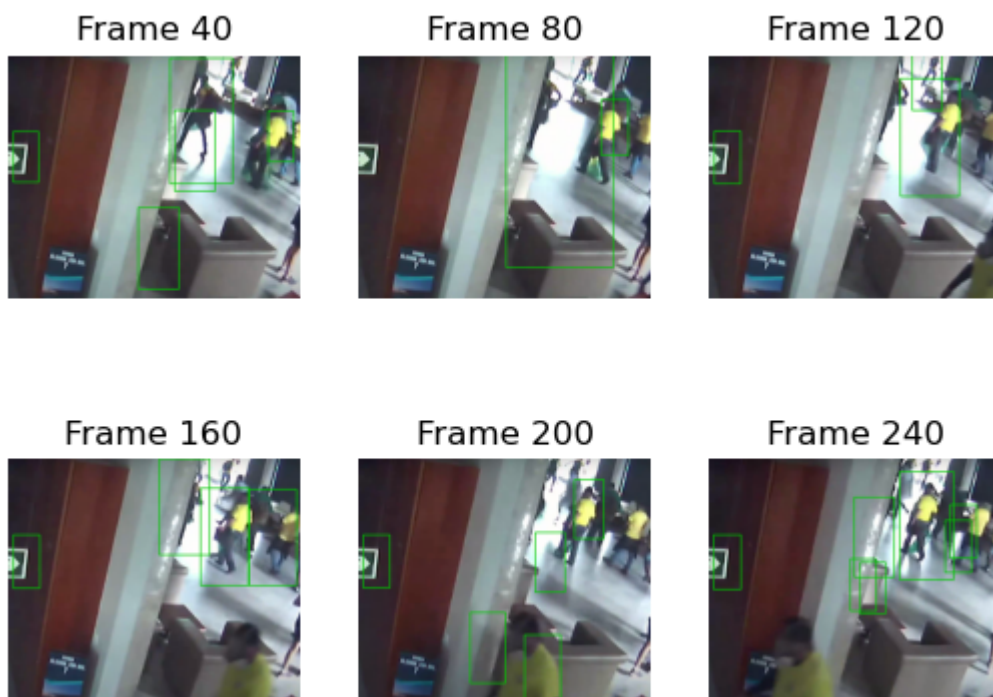
Fonte: <https://twitter.com/i/status/1323758776349020160>

Vídeo 2 - Uma briga de rua



Fonte: <https://twitter.com/i/status/1686060717626081294>

Vídeo 3 - Câmeras de segurança do Supremo Tribunal Federal em 08/01



Fonte: https://www.youtube.com/watch?v=_5bzM3m8uac

Código:

```
import numpy as np
import cv2
import matplotlib.pyplot as plt
```

```
import imutils
from imutils.object_detection import non_max_suppression
import sys
import time
```

```
HOGCV = cv2.HOGDescriptor()
HOGCV.setSVMdetector(cv2.HOGDescriptor_getDefaultPeopleDetector())
```

```
def lerVideo(caminho):
    cap = cv2.VideoCapture(caminho)
    frames = []

    if not cap.isOpened():
        raise("Erro abrindo o arquivo!")

    while True:
        ret, frame = cap.read()
        if not ret:
            print("Último frame ou erro")
            break
        frames.append(frame)
    if cv2.waitKey(1) == ord('q'):
        break

    print("Leitura completa")
    cap.release()
    return np.array(frames)
```

```
def reproduzirVideo(frames):
    for frame in frames:
        cv2.imshow("Sem fundo", frame)
        if cv2.waitKey(1) == ord('q'):
            break

        time.sleep(1/30)
    cv2.destroyAllWindows()
```

```
def Detector(frame):
    rects, weights = HOGCV.detectMultiScale(frame, winStride=(4, 4),
padding=(8, 8), scale=1.03)
    rects = np.array([[x, y, x + w, y + h] for (x, y, w, h) in rects])
    pick = non_max_suppression(rects, probs=None, overlapThresh=0.65)
    c = 1
```

```
for x, y, w, h in pick:
    cv2.rectangle(frame, (x, y), (w, h), (0, 200, 0), 2)
    c += 1

return frame, c


def main():
    frames = lerVideo('anao.mp4')
    frames_processado = []
    for i, frame in enumerate(frames):
        framecorrente, c = Detector(frame)
        frames_processado.append(framecorrente)

    frames_processado = np.array(frames_processado)
    #reproduzirVideo(frames_processado)


if __name__ == "__main__":
    main()
```