

COMPUTAÇÃO GRÁFICA

SANTOS, M. R (marcos.santos@imed.edu.br)

COMPUTAÇÃO GRÁFICA

Objetivo da Disciplina:

- Possibilitar aos alunos uma visão tecnológica da área da Computação Gráfica, com ênfase em Análise e Síntese de Imagens.

Ementa da Disciplina:

- Fundamentos: percepção visual e formação da imagem, amostragem e quantização, relações entre elementos da imagem, operações aritméticas e lógicas básicas. Conceitos de Transformações de imagens: Transformada de Fourier. Realce de Imagens: Domínios Espaço e Frequência. Segmentação de Imagens. Representação e Descrição de Imagens. Reconhecimento e Interpretação. Fundamentos de Processamento de Vídeo. Tópicos especiais em processamento de Imagens e Vídeos.

Ferramentas

Abaixo é listado as principais ferramentas utilizadas na construção desta disciplina.



<https://www.python.org/>



<https://atom.io/>

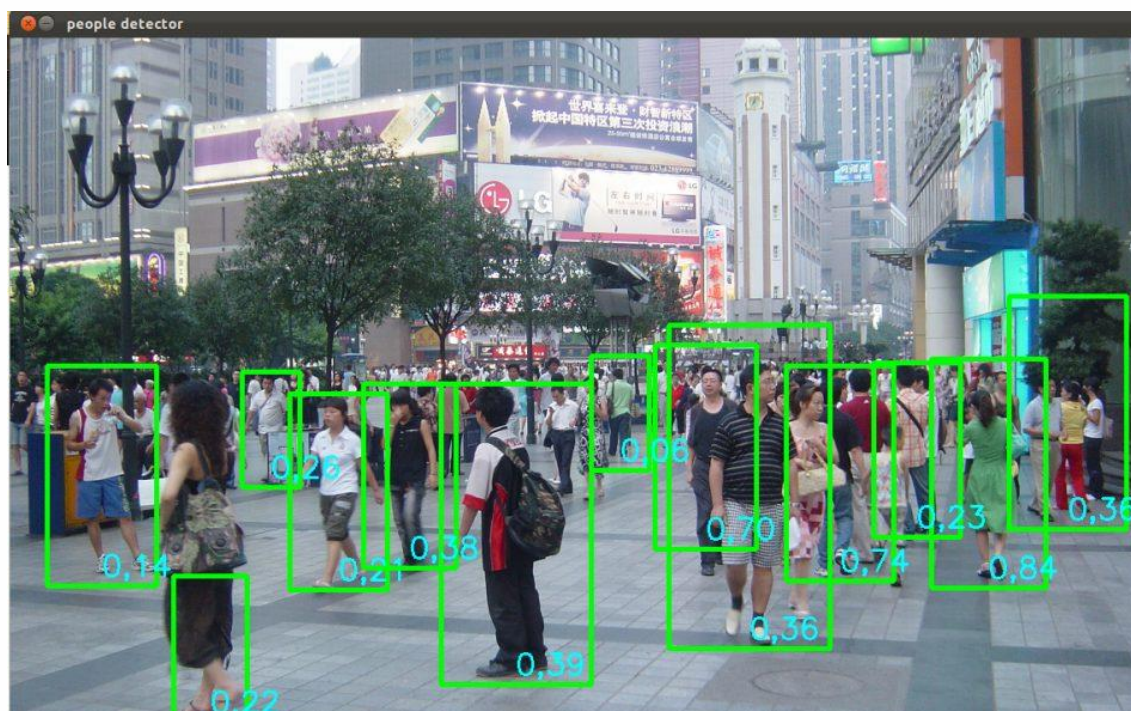


<https://opencv.org/releases.html>

Visão Computacional

Visão computacional ou computer vision (CV) é a ciência responsável pela forma com que o computador enxerga o meio a sua volta, extraindo informações úteis através de imagens capturadas por câmeras de vídeo, satélites, sensores, scanners, entre outros. A utilização destes métodos tem aumentado consideravelmente nos últimos anos em diversas áreas de aplicação, como na indústria, na lavoura, no trânsito, entre outros, pois fornecem quantidades significativas de informação. Além disso, esta tecnologia possibilita analisar informações do espectro eletromagnético em que o olho humano não é sensível, tal como ultravioleta (UV) ou regiões espectrais de infravermelho.

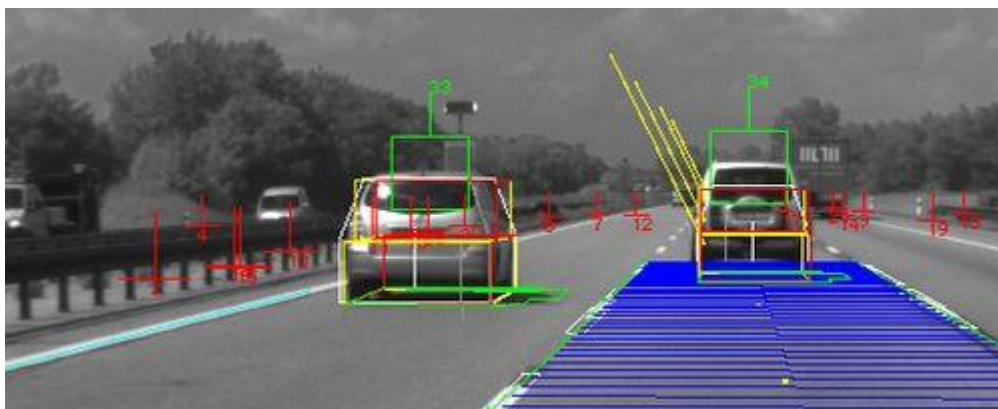
A seguir serão abordados métodos e técnicas que correlacionam com as teorias e aplicações de CV e a extração de informações úteis para o processamento dos dados. Ainda será explanado sobre a biblioteca de código aberto OpenCV e sua possível aplicação em sistemas de visão computacional.



Captura de Imagens

A aquisição de imagens é o primeiro passo para se aplicar técnicas de visão computacional e justapor o devido tratamento para a obtenção de informações. A captura de imagens depende de dois componentes. O primeiro é um dispositivo físico que seja sensível a uma banda do espectro de energia eletromagnética e que produza um sinal elétrico de saída proporcional a um nível de energia percebida.

Como exemplos, têm-se equipamentos de ultrassom, radiografia, microscópios eletrônicos, radares, equipamento de ultrassom, câmeras digitais. O segundo é um dispositivo digitalizador para a conversão da saída elétrica para a forma digital.



Ainda a aquisição é o processo de obtenção da imagem por meio de um sensor de imageamento capaz de digitalizar o sinal produzido. Para tanto, algumas técnicas de pré-processamento são necessárias para melhorar a qualidade da imagem, com o intuito de facilitar a aplicação das etapas subsequentes, que serão elucidadas nos itens a seguir.

Iluminação

Dentre as principais características que determinam a qualidade que uma imagem digital pode apresentar, a iluminação destaca-se por estar relacionada à clara definição dos itens de uma cena. O termo imagem refere-se a uma função de intensidade luminosa bidimensional, denotada por $f(x,y)$, em que o valor da amplitude de f nas coordenadas espaciais (x,y) dá a intensidade (brilho) da imagem naquele ponto. Como a luz é uma forma de energia, $f(x,y)$ deve ser uma quantia positiva e finita, ou seja, $0 < f(x,y) < 1$.

A importância da iluminação é que uma boa prática para capturar imagens é colocar os itens fotografados em um ambiente controlado, a fim de evitar variações de iluminação entre as imagens coletadas. A Figura abaixo retrata um exemplo de ambiente controlado para a coleta das imagens.

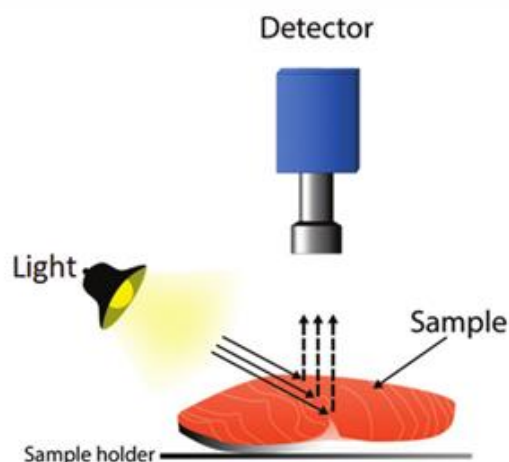


Figura: Esquema para coleta de imagens com iluminação controlada.

As imagens que as pessoas percebem em atividades visuais corriqueiras consistem de luz refletida dos objetos. A natureza básica de $f(x,y)$ pode ser caracterizada por dois componentes: (1) a quantidade de luz incidindo na cena observada e (2) a quantidade de luz refletida pelos objetos da cena. Esses componentes são chamados de iluminação e refletância, respectivamente, e são representados por $i(x,y)$ e $r(x,y)$, onde o produto das funções $i(x,y)$ e $r(x,y)$ resulta $f(x,y)$.

Esta última equação indica que a refletância é limitada entre 0 (absorção total) e 1 (refletância total). A natureza de $i(x,y)$ é determinada pela fonte de luz, e $r(x,y)$ é determinada pelas características dos objetos observados.

Imagem de Reamostragem

Uma imagem digital pode passar por transformações em relação a sua resolução espacial antes de ser processada e analisada. A essas mutações dá-se o nome de imagem de reamostragem, que tem por finalidade o aumento ou a redução da resolução. No caso de redução tem-se uma subamostragem e no caso de aumento de resolução tem-se uma interpolação.

Esta técnica é utilizada para ampliar, reduzir e rotacionar imagens digitais, eliminando informações irrelevantes e indesejáveis presente na imagem. Na prática, este método permite que algoritmos de processamento de imagens, baseado na exploração de pixels (px), tenham um tempo de resposta relativo ao tamanho da imagem ou área foco que necessita ser analisada.



Uma imagem de reamostragem é realizada pelas aplicações de técnicas como detecção de linhas, armazenamento das linhas detectadas, eliminação de pontos inválidos, redimensionamento, equalização de contraste, entre outros.

Construção de Imagem Base

Imagem base é criada a partir da imagem de reamostragem, aplicando-se métodos de detecção de bordas. A finalidade da detecção de borda, em geral, é reduzir significativamente a quantidade de dados em uma imagem, enquanto preserva as propriedades estruturais a serem utilizadas para o processamento da gravura.

A Figura abaixo demonstra o exemplo de uma imagem base (imagem área de uma lavoura) processada por meio de um algoritmo de detecção de bordas.

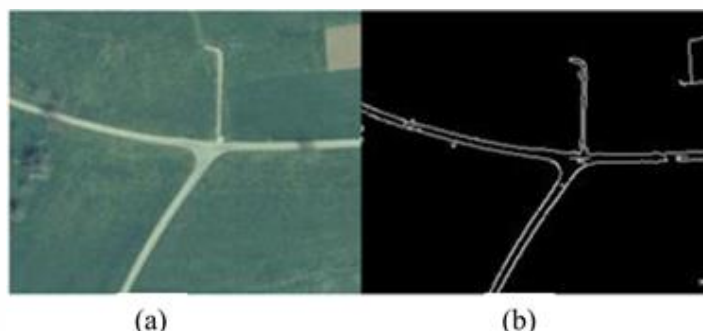


Figura: (a) Imagem de Entrada e (b) Imagem Base.

Para a execução de algoritmos de detecção de bordas, destacam-se dois filtros espaciais lineares que podem ser aplicados: baseado no gradiente da função luminosidade da imagem e baseados no operador laplaciano.

Armazenamento de Linhas Detectadas

As informações geradas pelos métodos de detecção de bordas possuem coordenadas dos pixels necessários para os processamentos subsequentes. O armazenamento depende de uma estrutura de dados onde cada cadeia de pixels contíguos é armazenada em estruturas distintas do tipo array (vetor), representando segmentos da imagem. Na Figura abaixo, são apresentados diversos segmentos de objetos na imagem, resultado de um pré-processamento.



Figura: Segmentação de objetos de uma imagem.

Uma análise detalhada do objeto em questão pode ser exemplificada na Figura abaixo, incluindo o pixel correspondente de cada marcação:

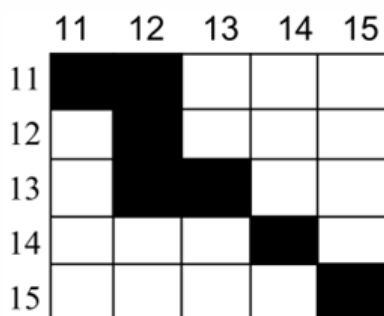


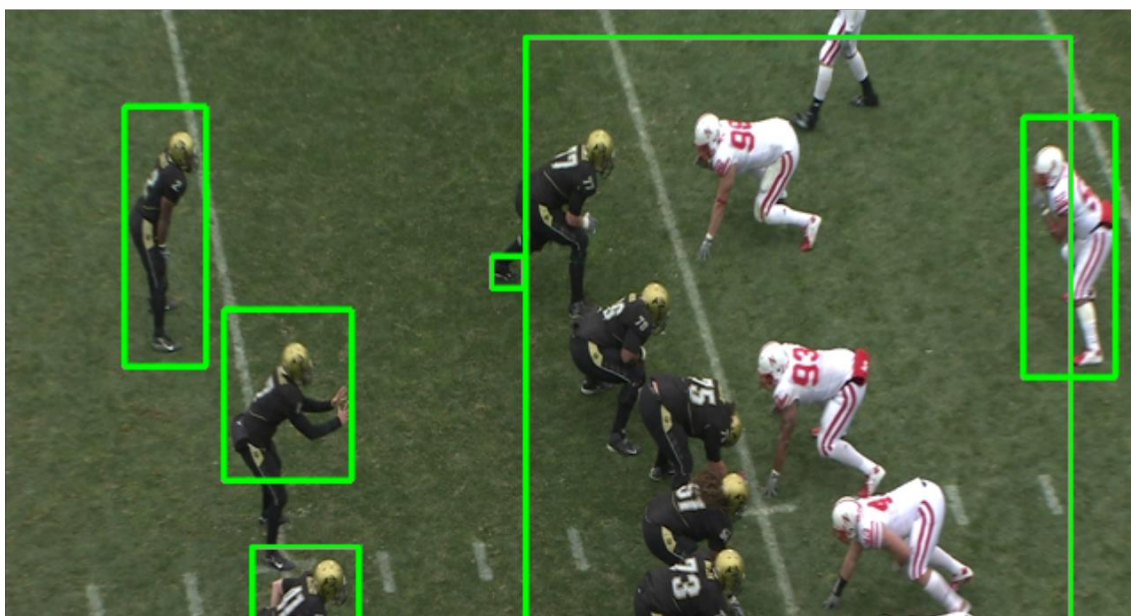
Figura: Detalhamento de objeto pixel a pixel.

Propondo a estrutura de dados para armazenar as informações do objeto selecionado, tem-se o seguinte array de inteiros: [11,11], [11,12], [12,12], [13,12], [13,13], [14,14], [15,15].

PROCESSAMENTO DE IMAGENS DIGITAIS

Uma imagem digital pode ser considerada uma matriz cujos índices de linhas (N) e de colunas (M) identificam um ponto na imagem, e o correspondente valor do elemento da matriz identifica a cor naquele ponto. Os elementos básicos dessa matriz digital são chamados de elementos da imagem, elementos da figura ou "pixels"(abreviação de "picture elements").

Cada pixel representa uma medida que dependem de variáveis como cor, profundidade e tempo. É o menor ponto que forma uma imagem digital.



Pode-se entender uma imagem digital como um agrupamento de sensores que, quando atingidos por raios de luzes, gravam a tonalidade de cor que o atingiu. Logo, quanto mais pixels (sensores) uma imagem tem, melhor é sua qualidade e maior é a sua resolução.

As próximas subseções explicarão as principais etapas aplicadas ao processamento de imagens digitais, como técnicas que realçam as características da figura e diminuem imperfeições, bem como operações matemáticas utilizadas para extrair informações úteis do conjunto final de pixels.

Etapas de Processamento de Imagens

O principal objetivo do processamento de imagens é a remoção de barreiras inerentes ao sistema visual humano, para facilitar a extração de informações. O reconhecimento de padrões encontrados em imagens pode ser dividido em cinco etapas:

- aquisição da imagem,
- pré-processamento,
- segmentação,
- representação/descrição e
- reconhecimento/ interpretação,

conforme o esquema apresentado na Figura abaixo:

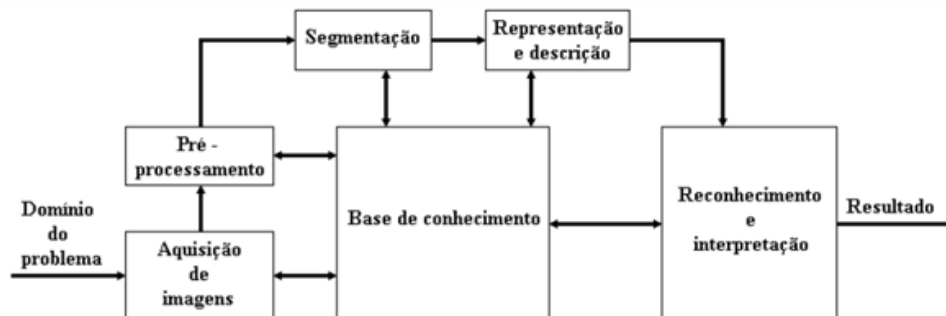


Figura: Etapas fundamentais do processamento de imagens [12].

Tamanho e Formato de Imagens Digitais

O tamanho de uma imagem está altamente correlacionado com o tempo de processamento e o custo computacional em aplicações com sistemas de visão computacional. Para se obter estimativas do tamanho real de objetos em uma cena é necessário fazer a relação entre a média em milímetros deste objeto e o número de linhas e colunas de pixels que este possui na imagem digital. Com essa técnica é possível analisarmos os feedbacks dos sistemas de visão computacional com unidades de medida entendíveis pelos seres humanos como centímetros e ou metros.

Ainda na detecção do tamanho de objetos na cena, um dos principais modelos utilizados é localizar o primeiro pixel que corresponde ao objeto, assim o algoritmo varre recursivamente sua vizinhança buscando por pixels de valores iguais. Esse processo acaba apenas quando não há mais pixels de valor igual adjacentes ao ruído.

Conforme os formatos de arquivos de imagens, é definido que existe necessidade de se designar padrões de armazenamento de imagens de forma que se possa conseguir a interação destas entre diferentes sistemas. Outro fator importante é a codificação das imagens, uma vez que estas normalmente ocupam muito espaço de memória e por isso, necessitam do emprego de alguma forma de compressão de dados para o seu armazenamento. Existem diferentes formatos de arquivos para o armazenamento de imagens, e os principais são apresentados na Tabela abaixo.

Tabela 1. Formatos de arquivos para armazenamento de imagens

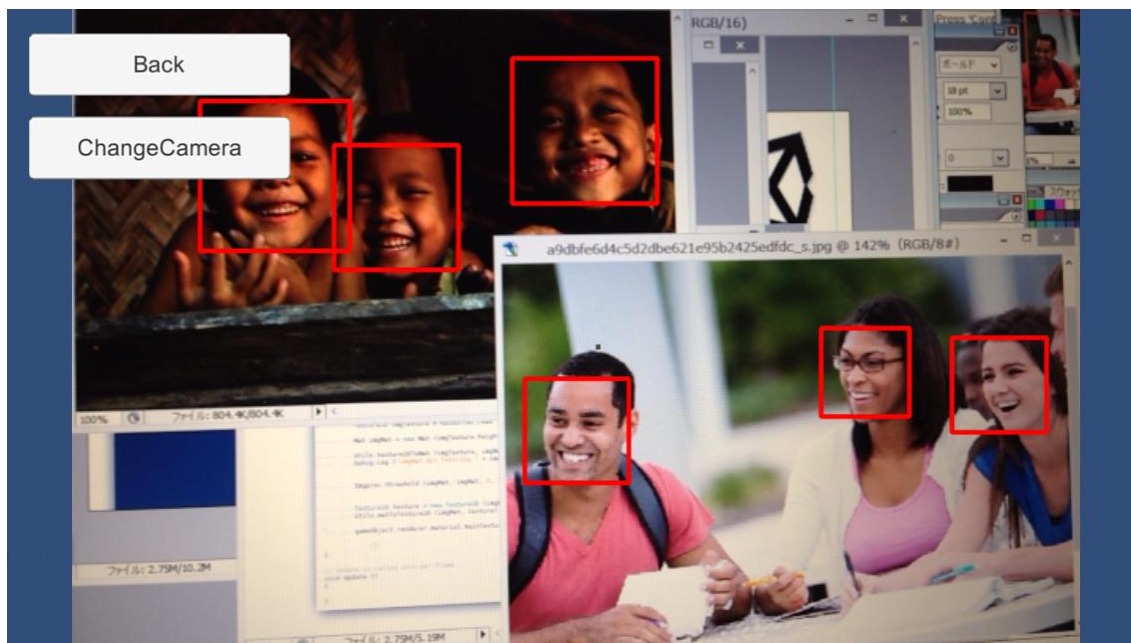
Formato / Nome	Descrição	Reconhecimento Extensão
TIFF	Tagged Image File Format	.tif; .tiff
JPEG	Joint Photographic Experts Group	.jpg; .jpeg
GIF	Graphics Interchange Format	.gif
BMP	Windows Bitmap	.bmp
PNG	Portable Network Graphics	.png
XWD	XWD	.xwd %

Opencv

Open Source Computer Vision Library. Originalmente, desenvolvida pela Intel, em 2000, é uma biblioteca multiplataforma, totalmente livre ao uso acadêmico e comercial, para o desenvolvimento de aplicativos na área de Visão computacional, bastando seguir o modelo de licença BSD Intel.

O OpenCV possui módulos de Processamento de Imagens e Video I/O, Estrutura de dados, Álgebra Linear, GUI (Interface Gráfica do Usuário) Básica com sistema de janelas independentes, Controle de mouse e teclado, além de mais de 350 algoritmos de Visão computacional como: Filtros de imagem, calibração de câmera, reconhecimento de objetos, análise estrutural e outros. O seu processamento é em tempo real de imagens.





Esta biblioteca foi desenvolvida nas linguagens de programação C/C++. Também, dá suporte a programadores que utilizem Java, Python e Visual Basic e desejam incorporar a biblioteca a seus aplicativos. A versão 1.0 foi lançada no final de 2006.

Áreas de aplicação

- Humano-Computador Interface (HCI)
- Identificação de objetos
- Sistema de reconhecimento facial
- Reconhecimento de movimentos
- Gravação de vídeos
- Robôs móveis
- Reconstrução 3D
- Realidade virtual
- Realidade aumentada
- Realidade mista

Estrutura do OpenCV

- cv — Módulo das principais funcionalidades e algoritmos de Visão Computacional do OpenCV.
- cvaux — Módulo com algoritmos de Visão, ainda está em fase experimental.
- cxcore — Módulo de Estrutura de Dados e Álgebra Linear.
- highgui — Módulo de Controle de Interface e dispositivos de entrada.
- ml — Módulo de “Machine Learning” é um módulo processador de imagem que utiliza algoritmos com aprendizagem de máquina.
- dnn — Módulo de “Deep Learning” é um módulo processador de imagem que utiliza algoritmos com aprendizagem de máquina profundo.
- ed — Manual de estrutura de dados e operações.

Instalação das Ferramentas



Para executar os comandos abaixo, execute o terminal como administrador. Em alguns casos, é necessário instalar o c++ sob o link abaixo:

<https://www.microsoft.com/en-us/download/details.aspx?id=53587>

```
pip install numpy
```

O NumPy é o pacote básico da linguagem Python que permite trabalhar com arranjos, vetores e matrizes de N dimensões, de uma forma comparável e com uma sintaxe semelhante ao software proprietário Matlab, mas com muito mais eficiência, e com toda a expressividade da linguagem. Provê diversas funções e operações sofisticadas, incluindo (mas não se limitando a):

- Objeto array para a implementação de arranjos multidimensionais
- Objeto matrix para o cálculo com matrizes
- Ferramentas para álgebra linear
- Transformadas de Fourier básicas
- Ferramentas sofisticadas para geração de números aleatórios

Além disso tudo, as classes criadas podem ser facilmente herdadas, permitindo a customização do comportamento (por exemplo, dos operadores típicos de adição, subtração, multiplicação, etc.). O módulo é implementado em linguagem C, o que dá uma grande velocidade às operações realizadas.

Página Oficial: <http://pyscience-brasil.wikidot.com/module:numpy>

```
pip install opencv_python
```

Verificando a versão

```
import cv2  
print (cv2.__version__)
```

Testando funcionalidades

A imagem clouds.jpg está disponível na pasta da disciplina (images)

```
import cv2
print (cv2.__version__)

image = cv2.imread("clouds.jpg")
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
cv2.imshow("Acima das nuvens", image)
cv2.imshow("Acima das nuvens - Cinza", gray_image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```