

# ESZA019-17 - Visão Computacional - NA

## Prática 01 - Experimentos Básicos - v02

Prof. Celso Kurashima

Sala 904 - Bloco B - [celso.kurashima@ufabc.edu.br](mailto:celso.kurashima@ufabc.edu.br)

Prof. Filipe Ieda Fazanaro

Sala 929 - Bloco B - [filipe.fazanaro@ufabc.edu.br](mailto:filipe.fazanaro@ufabc.edu.br)

12 de junho de 2019

CONSIDERAÇÕES GERAIS	
Data limite para entrega do relatório:	27/06/2019
Entrega do relatório após data limite:	Rel01 = 0,0
Nota final dessa prática:	R1 = Rel01

## Descrição

É necessário que os pacotes básicos estejam instalados. Use o apêndice disponibilizado na pasta da disciplina ([aqui](#)). Ou se estiver usando o seu computador (laptop provavelmente), use um dos tutoriais descritos no plano de ensino.

# Atividades em Aula - Parte 01

## 1. Leitura de imagem em arquivo:

- (a) O programa “**lab01.py**” descrito no roteiro-apêndice realiza a leitura de uma imagem gravada em disco e mostra a imagem em uma nova janela no Linux (ou no Windows ou em qualquer outro sistema operacional que você estiver usando). Além disso, o programa permite salvar a imagem em arquivo.

## 2. Leitura de vídeo de um arquivo:

- (a) Dentro da subpasta “lab01”, crie o código “**lab01tocavideo.py**”, descrito a seguir. Esse código realiza a leitura de um vídeo gravado em arquivo e mostra o vídeo em uma nova janela Linux.

```
import time
import numpy as np
import cv2 as cv

cap = cv.VideoCapture('test.mp4')

while(cap.isOpened()):
    ret, frame = cap.read()

    if ret==True:
        # show the frame
        cv.imshow('frame',frame)

        #wait next frame by 40ms - 25fps
        time.sleep(1/25.0)

        if cv.waitKey(1) & 0xFF == ord('q'):
            break
    else:
        break

cap.release()
cv.destroyAllWindows()
```

Após digitar e salvar o código, dentro do ambiente virtual, execute-o com a seguinte linha de comando:

```
1 python lab01tocavideo.py
```

### 3. Leitura de imagem de câmera:

- (a) Dentro da subpasta “lab01”, crie o código “**lab01camera.py**”, descrito a seguir. Esse código realiza a leitura de imagens da câmera e mostra a imagem capturada em uma nova janela Linux.

```
import numpy as np
import cv2 as cv

cap = cv.VideoCapture(0)

while(True):
    # Capture frame-by-frame
    ret, frame = cap.read()

    # Our operations on the frame come here
    gray = cv.cvtColor(frame, cv.COLOR_BGR2GRAY)

    # Display the resulting frame
    cv.imshow('frame',gray)

    if cv.waitKey(1) & 0xFF == ord('q'):
        break

# When everything done, release the capture
cap.release()
cv.destroyAllWindows()
```

Após digitar e salvar o código, **dentro do ambiente virtual**, execute-o com a seguinte linha de comando:

```
1 python lab01camera.py
```

#### 4. Gravação de vídeo da câmera:

- (a) Dentro da subpasta “lab01”, crie o código “**lab01capvideo.py**”, descrito a seguir. Esse código realiza a leitura de imagens da câmera, mostra a sequência de imagens em uma nova janela do Linux e, ao digitar a tecla “q”, salva essa sequência em um arquivo de formato AVI de vídeo.

```
import numpy as np
import cv2 as cv

cap = cv.VideoCapture(0)

# ----- #
# Define the codec and create VideoWriter object
# ----- #
fourcc = cv.VideoWriter_fourcc(*'XVID')
# ----- #
out = cv.VideoWriter('saidavideo.avi',fourcc, 20.0, (640,480))

while(cap.isOpened()):
    ret, frame = cap.read()
    if ret==True:
        # write the frame
        out.write(frame)

        cv.imshow('frame',frame)

        if cv.waitKey(1) & 0xFF == ord('q'):
            break
    else:
        break

# Release everything if job is finished
cap.release()
out.release()
cv.destroyAllWindows()
```

Após digitar e salvar o código, **dentro do ambiente virtual**, execute-o com a seguinte linha de comando:

```
1 python lab01capvideo.py
```

# Atividades para Entregar

**Observação:** as atividades descritas nessa seção deverão ser contempladas no relatório.

## 1. Obtenção de fotos:

- (a) Obtenha fotos de você mesmo, preferencialmente em uma pose de frente, sem óculos, com fundo claro e boa iluminação.
- (b) Salve as imagens em cores.
- (c) Converta para tons de cinza a partir de duas soluções. A primeira emprega a relação (1)

$$I_{\text{gray}} = \alpha I_{\text{RGB}}(:, :, R) + \beta I_{\text{RGB}}(:, :, G) + \gamma I_{\text{RGB}}(:, :, B) \quad (1)$$

onde  $I_{\text{RGB}}$  refere-se à imagem colorida que você salvou. Os índices R, G e B indicam as camadas de cores.  $\alpha = 0.2989$ ,  $\beta = 0.5870$ ,  $\gamma = 0.1140$  indicam as ponderações para essa conversão. A segunda solução consiste em empregar funções específicas do OpenCV-Python. Pesquise a respeito.

- (d) Em todos os casos, salve as imagens nos formatos “bmp” e “jpg”. Compare as diferenças de tamanho. **Todas essas imagens serão utilizadas na próxima aula prática. Utilize a sua página no dropbox para salvá-las.**
- (e) Estude os programas “lab01.py” e “lab01camera.py”. Crie um novo programa para realizar o que foi solicitado.

## 2. Obtenção de vídeos:

- (a) Capture dois vídeos, sendo o primeiro com movimentos lentos e o segundo com movimentos rápidos, de um objeto contendo uma ou várias cores, e tendo o fundo uma cor diferente do objeto (e.g. uma caneta preta em um fundo branco). É importante que o objeto esteja bem iluminado. Anote a taxa de quadros por segundo utilizada durante a captura.
- (b) Salve cinco quadros de cada vídeo, no formato “bmp”, representando instantes diferentes. Observe - e relate - se é possível notar diferenças entre imagens do mesmo vídeo e entre vídeos diferentes.

## Observações

- 1) Os relatórios deverão ser entregues única e exclusivamente pelo Dropbox. Coloque todos os arquivos lá (imagens, vídeos, relatório, códigos em Python).
- Cada código deve estar comentado, contendo um cabeçalho com a sua identificação, breve descrição do que o código faz e comentário pertinentes ao seu completo entendimento. Veja o código ilustrado na Figura 1 apresentado como exemplo de como isso deve ser feito. Na dúvida, pergunte.

```
1 # ESZA019-17 – Visao Computacional – NA – 2Q2019
2 # PRATICA 01
3 #
4 # RA: <seu RA>
5 # NOME: <seu nome>
6 #
7 # E-MAIL: <o email que voce estah utilizando para comunicacao>
8 #
9 # DESCRICAO:
10 #   – Descreva em linhas gerais, o que o seu codigo esta fazendo;
11 #   – No esqueca de inserir comentarios pertinentes ao completo
12 #   entendimento do codigo.
13 import numpy as np
14 import cv2 as cv
15
16 img = cv.imread('messi5.jpg',0)
17
18 cv.imshow('image',img)
19
20 k = cv.waitKey(0)
21
22 if k == 27: # wait for ESC key to exit
23     cv.destroyAllWindows()
24
25 elif k == ord('s'): # wait for 's' key to save and exit
26     cv.imwrite('/home/esza019/Desktop/SeuPrimeiroNome/lab01/messigray.png',img)
27     cv.destroyAllWindows()
```

Figura 1: Exemplo de como os códigos realizados devem estar comentados.

- 2) O relatório deverá estar em formato .pdf. Arquivos enviados em quaisquer outros formatos não serão considerados.
- 3) Os seguintes cuidados devem ser levados em consideração:
  - Deve-se adotar o seguinte padrão: esza019\_<seu RA>\_<seu nome>\_rel01.pdf
    - Exemplo: esza019\_001\_Ozzy\_Osborne\_rel01.pdf
    - Arquivos fora dessa padronização não serão aceitos.
- 4) Envie uma e-mail para mim confirmando que você subiu os arquivos na sua pasta no Dropbox.
  - Título do e-mail: [ESZA019-17] <RA> - <Nome> - Prática 01

- No dia seguinte após a finalização do prazo de entrega, você receberá uma resposta confirmando se o relatório foi recebido e se o mesmo respeita as especificações supracitadas.

## Sobre o relatório

O relatório deve estar formatado no padrão IEEE de conferência (especificado no plano de ensino e discutido em aula) e deve estar organizado da seguinte maneira:

- Título com a sua identificação (nome da instituição, nome do aluno e o respectivo e-mail de contato);
- Resumo:
  - Autoexplicativo. Resumo das atividades realizadas e considerações sobre os resultados obtidos.
- Metodologia:
  - Explicação da metodologia empregada.
- Resultados:
  - Imagens obtidas, tabelas de análises comparativas de resultados;
  - **Análise dos resultados (interpretação).**
- Conclusões e discussões;
- Bibliografia (se houver);
- Após a seção de Bibliografia, insira uma seção “Biografia” que deverá conter uma foto sua.

## Notas da Versão

**v01:** versão inicial, disponibilizada em 11/06/2019.

**v02:** versão disponibilizada em 25/06/2019. Alteração do nome da seção “Atividades em Aula - Parte 02” para “Atividades para Entregar”. Explicita quais as atividades devem ser contempladas no relatório de atividades referente à presente prática.

## Referências

MINICHINO, J., HOWSE, J.. Learning OpenCV 3 Computer Vision with Python, 2nd Edition, Packt Publishing, 2015.

Tutorial: [OpenCV-Python Tutorials](#)

Tutorial: [Getting Started with Images](#)