

FIAP

# 1. dlib

# Objetivos da aula:

- apresentar e aplicar o dlib para detecção de face

Atualmente temos disponíveis para uso outras diversas redes já treinadas para detecção não apenas de face mas também de mãos, corpo, objetos.....

Vou destacar algumas redes:

Dlib C++

MTCNN

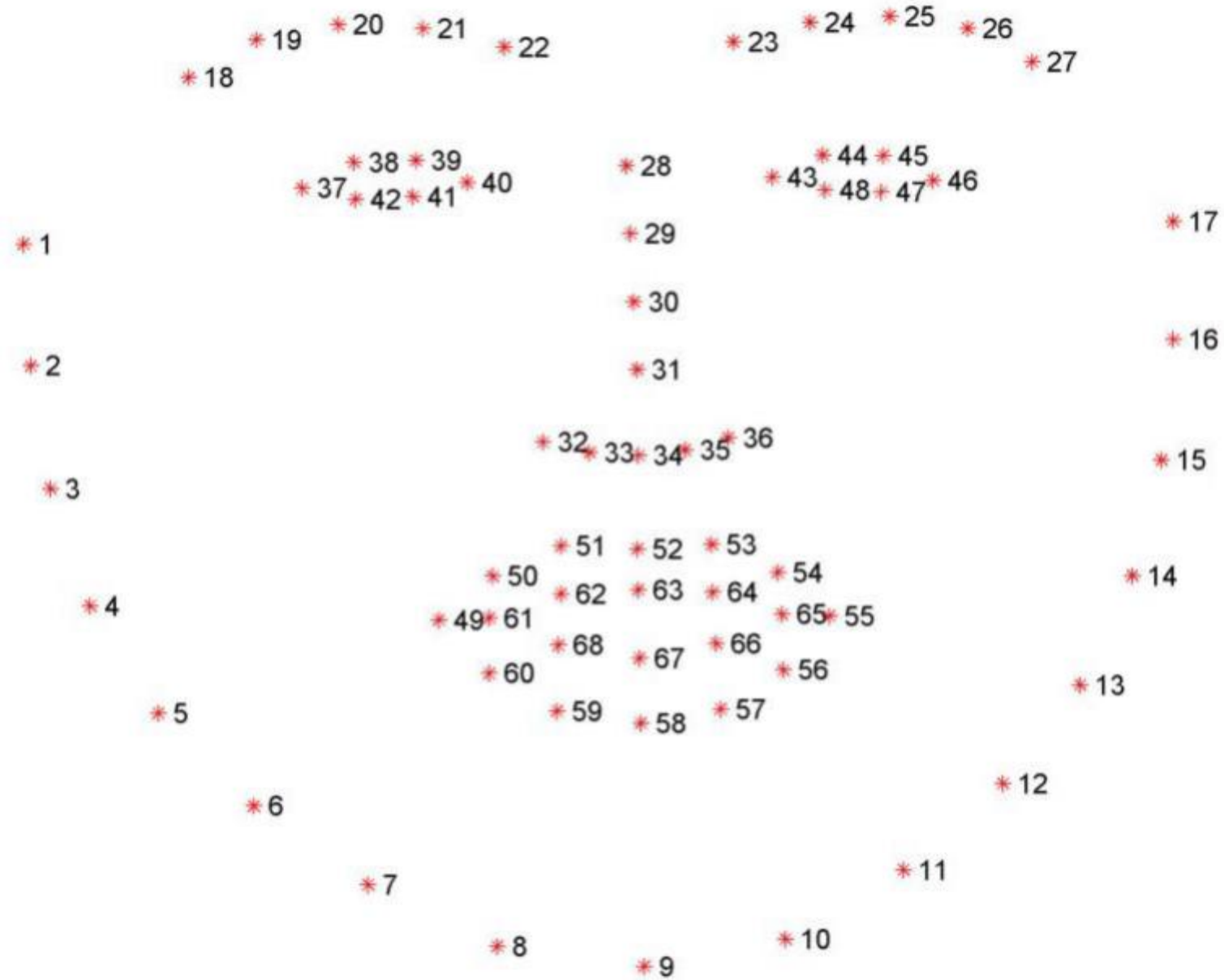
Media Pipe

Cada uma das redes acima aplica usam técnicas de machine learning na etapa de treinamento para realizar a detecção dessas redes dentre outras coisas conseguem realizar a detecção de face. Hoje vamos dar destaque especial para a rede Dlib. Como sugestão leia a documentação e aprenda a usar as redes MTCNN e Media Pipe para descobrir qual é mais rápida, mais leve ou mais acurada.

# Dlib

A Dlib, além de realizar a predição para detectar uma face, ela consegue ressaltar alguns pontos da face. Esses pontos de destaque são chamados de **\*\*landmarks\*\***, a Dlib consegue encontrar 68 pontos da face como cantos dos olhos, sobrancelhas, boca e ponta do nariz.

Cada landmark devolve uma coordenada da posição (x,y) da imagem.



# Instalação Dlib

- `pip install dlib`



# Deteção de faces com o dlib

- Vamos fazer o nosso "Hello World" usando a Dlib, neste caso, vamos implementar um detector de face simples.
- Note que o método é bem similar ao que já conhecemos, a lista de faces detectadas é um pouco diferente.

```
%matplotlib inline
import cv2
from matplotlib import pyplot as plt

import dlib

# carrega uma imagem para detectar o rosto
img1 = cv2.imread('lena.png')
img1_gray = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)

# Inicializa o detector dlib
detector = dlib.get_frontal_face_detector()

# Faz detecção das faces
faces = detector(img1_gray)

print(faces)

for face in faces:
    x,y = face.left(), face.top() # topo esquerda
    x1,y1 = face.right(), face.bottom() #baixo direita

    cv2.rectangle(img1, (x, y), (x1, y1), (0, 255, 0), 1)

# Exibe imagem
plt.figure(figsize = (10,10))
plt.imshow(cv2.cvtColor(img1,cv2.COLOR_BGR2RGB)); plt.show();
```



# Detecção dos landmarks da face

- Para predição dos landmarks uma rede neural já treinada é utilizada, serão preditos 68 pontos da face.
- Para carregar os pesos da rede é necessário fazer o download de:

hape\_predictor\_68\_face\_landmarks.dat

- Dlib positions
- ("mouth", (48, 68)),
- ("right\_eyebrow", (17, 22)),
- ("left\_eyebrow", (22, 27)),
- ("right\_eye", (36, 42)),
- ("left\_eye", (42, 48)),
- ("nose", (27, 35)),
- ("jaw", (0, 17))

```

%matplotlib inline
import cv2
from matplotlib import pyplot as plt

import dlib

# carrega uma imagem para detectar o rosto
img1 = cv2.imread('lena.png')
img1_gray = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)

# Inicializa o detector dlib
detector = dlib.get_frontal_face_detector()

# Inicializa o identificador de Landmark identifier. Você ter esse arquivo na pasta do projeto
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")

# Faz detecção das faces
faces = detector(img1_gray)

print(faces)

# para todas faces detectadas
for face in faces:
    # Faz a predição dos landmarks
    shape = predictor(img1_gray, face)
    for i in range(1,68): #São 68 landmark em cada face
        # Desenha um circulo e exibe o indice do landmark
        # shape.part(i).x devolve o valor x da coordenada
        cv2.circle(img1, (shape.part(i).x, shape.part(i).y), 1, (0,255,0), thickness=-1)
        # Escreve o indice de cada landmark na imagem
        cv2.putText(img1, str(i), (shape.part(i).x,shape.part(i).y), fontFace=cv2.FONT_HERSHEY_SCRIPT_SIMPLEX, fontScale=0.3, color=(0, 0, 255))

# Exibe imagem
plt.figure(figsize = (10,10))
plt.imshow(cv2.cvtColor(img1,cv2.COLOR_BGR2RGB)); plt.show();

```





# Desafio 1

- Implemente um código que faz o crop de uma das regiões preditas.
- Exemplo: crop da boca, sobrancelha, olho, nariz ou boca.

# Desafio 2

- Faça um código capaz de detectar em tempo real as landmarks face pela webcam.



## Desafio 3

- Inspirado na solução do detector de fadiga, implemente um código que faz a detecção de emoção. Ou seja, detecta se a pessoa está sorrindo ou não.



**Copyright © 2024 Prof. Arnaldo Jr/Yan  
Coelho**

**Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).**