

Machine Learning Day - DIO #####
22/06/2022 Prof. Diego Bruno

Live no YouTube: <https://www.youtube.com/watch?v=DAaTjE1hhsw>

GitHub: <https://github.com/rosacarla/Deteccao-de-faces-OpenCV-ResNet>

Ferramenta online para rotular imagens: <http://labelme.csail.mit.edu/Release3.0/>

** Este material contém anotações sobre conteúdo resumido da live e cópias de slides utilizados pelo professor Diego.



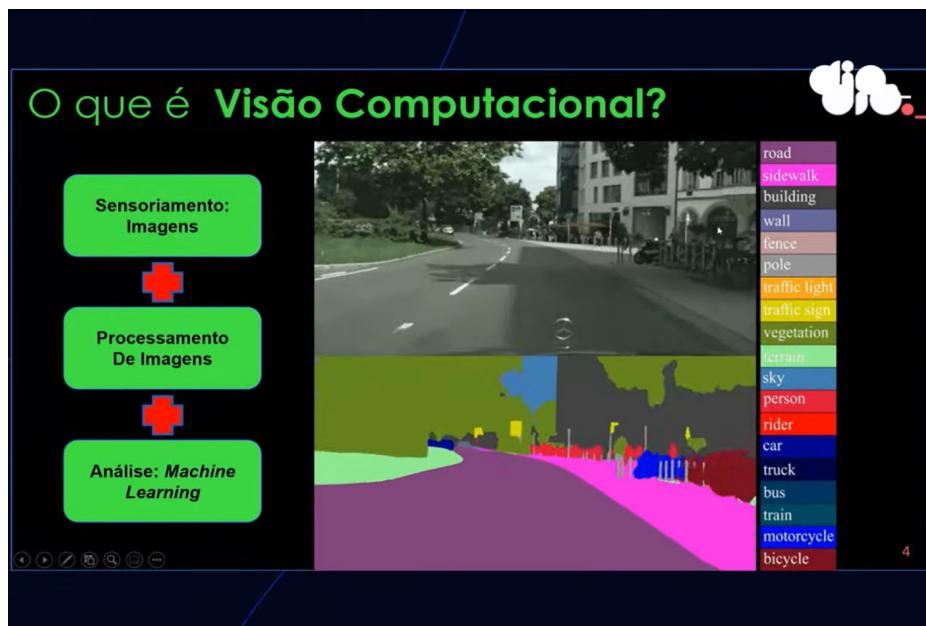
O objetivo do hands on é mostrar a evolução do reconhecimento facial, o que é a área de Visão Computacional envolvendo machine learning e focar no processamento de visão computacional.

Como exemplos temos os filmes que exploram os recursos de IA nos seus roteiros, porém a realidade tecnológica da IA ainda é bastante distante do que se imagina, por isso não cabe sentir medo de supostos malefícios.



IA Geral replica o que os seres humanos são capazes de fazer, ou seja, cozinhar, dirigir um carro, cantar etc.

IA Restrita é restrita para resolver certo tipo de problema, como em sistema de reconhecimento facial, realiza automações com base em inteligência artificial para retirar tarefas humanas que sejam prejudiciais à saúde, como em fábricas, ou facilitar tarefas do dia a dia: recomendar produto, desbloquear celular com imagem da face em substituição ao contato físico exigido pela biometria, o que oferece segurança. Pesquisas sobre reconhecimento da íris são de IA restrita. Machine Learning gera a tecnologia de IA.



Visão Computacional ocorre por meio de imagens de entrada que são capturadas por uma câmera qualquer, não tem nada de IA nessa parte, há apenas a leitura das imagens por meio dos sensores da câmera, armazenamento delas em matrizes e depois na memória do dispositivo. A visão computacional interpreta as imagens, detecta o que é cada parte para, no caso de veículos autônomos, diferenciar o que é obstáculo e tomar decisões de movimento. Esse tipo de sistema utiliza a IA para treinar e aprender o que é cada elemento das imagens. Trabalha-se com sensoriamento, processamento e análise das imagens com ML.



Exemplo de filme Robocop, cujo personagem principal tinha uma IA. Hoje temos recursos de visão computacional como foram representados em filmes de ficção científica no passado.



Quanto ao posicionamento da câmera, percebe-se que não é tão fácil desbloquear o celular com a imagem do rosto se for colocado em posição pouco amigável em relação às imagens com que foi treinado, as quais são feitas pelo usuário quando inicia o uso do aparelho para treinamento da IA. O sistema de reconhecimento não funciona se o aparelho estiver em posição ruim ou sob pouca iluminação. Mesmo com mudança de visual (cortar ou pintar cabelo, retirada ou acréscimo de barba etc.), o sistema continua com os reconhecimentos, por causa das variáveis incluídas (distância entre os olhos, comprimento do rosto na vertical e horizontal etc.).



A classificação dá certo por haver um objeto definido na imagem (imagem estática de um gato onde nada atrapalha), enquanto que na detecção é feita quando existem outros objetos que atrapalham a visão da imagem. Na imagem do gato com outros animais a detecção é mais complexa, porque é preciso detectar o objeto e classificá-lo dentro da área detectada, requer algoritmo que detecte o contorno

do objeto com fundo eliminado.



8

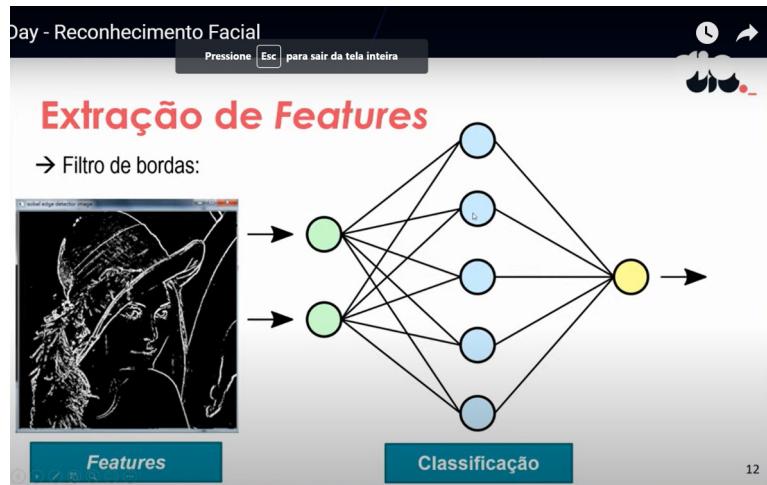


9

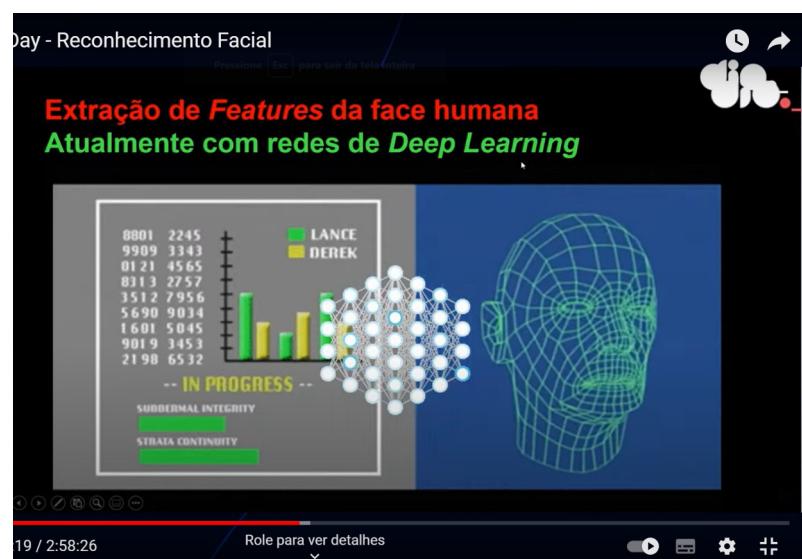


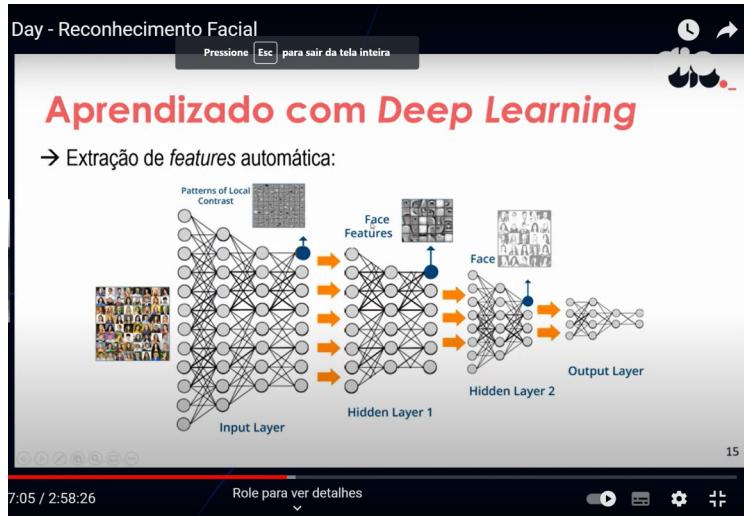
Imagen da Lena foi a primeira imagem escaneada por computador.

Imagen em preto e branco na verdade é composta por tons de cinza, contém variação de 0 a 255 cores, facilita o trabalho por ser representada por única matriz, enquanto que a imagem colorida é tripla. Sendo cinza, obtém-se menor quantidade de dados da imagem para processar. Para sistema de reconhecimento às vezes é preciso apenas o contorno da imagem.

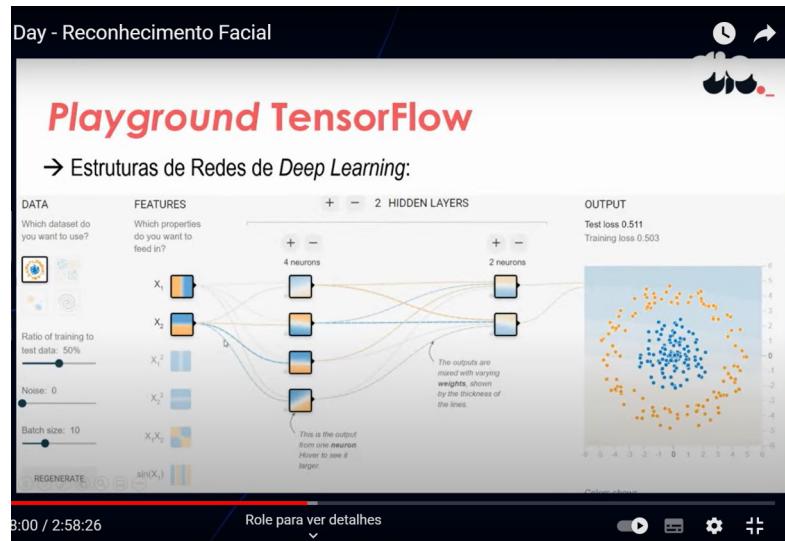


No passado a extração de features era feita manualmente, hoje as redes de deep learning fazem a tarefa com facilidade e maior precisão.

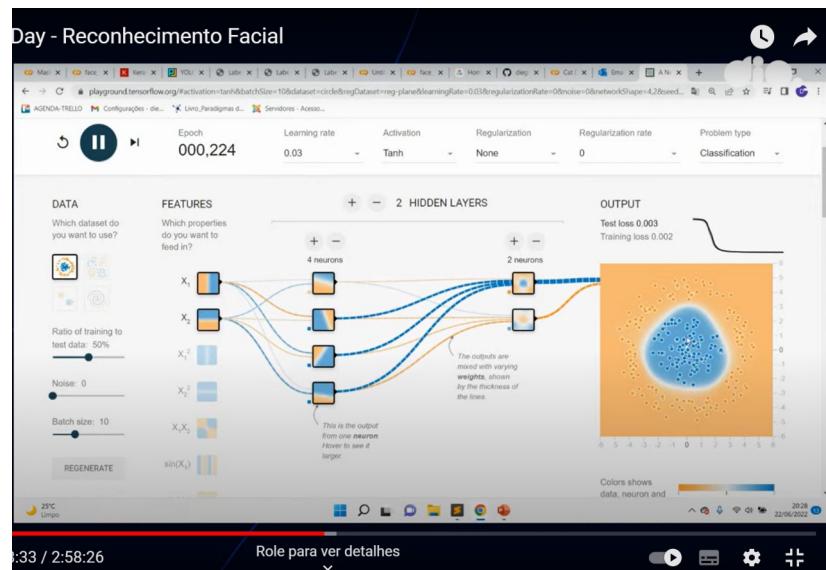




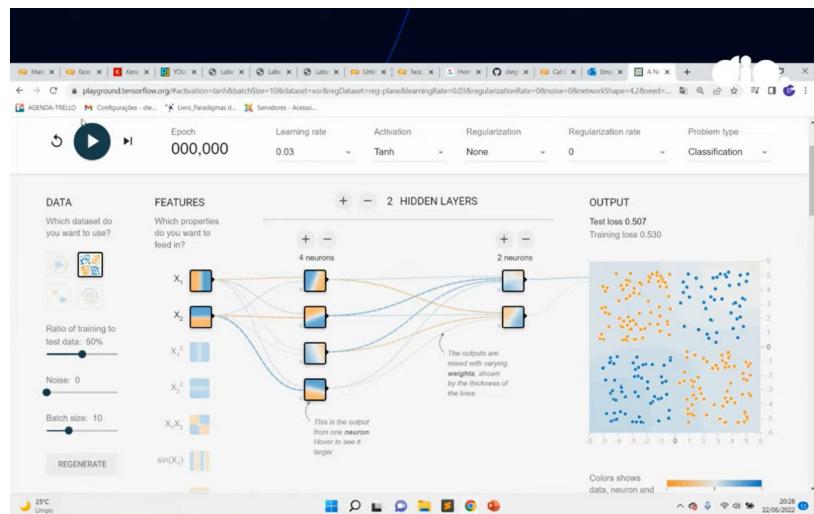
A rede tem capacidade de aprendizado profunda, por exemplo, consegue definir se determinada característica da imagem é relevante ou não para o reconhecimento, como um sinal no rosto de alguém. Por esse motivo chama-se deep learning.



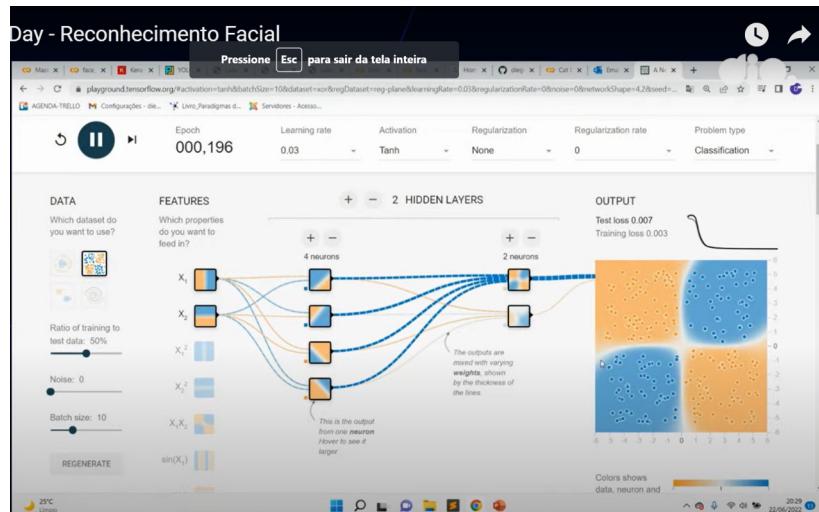
Exemplo de rede de deep learning no TensorFlow, com problema para resolver: classificar pontos azuis e laranjas da imagem. Resultado:



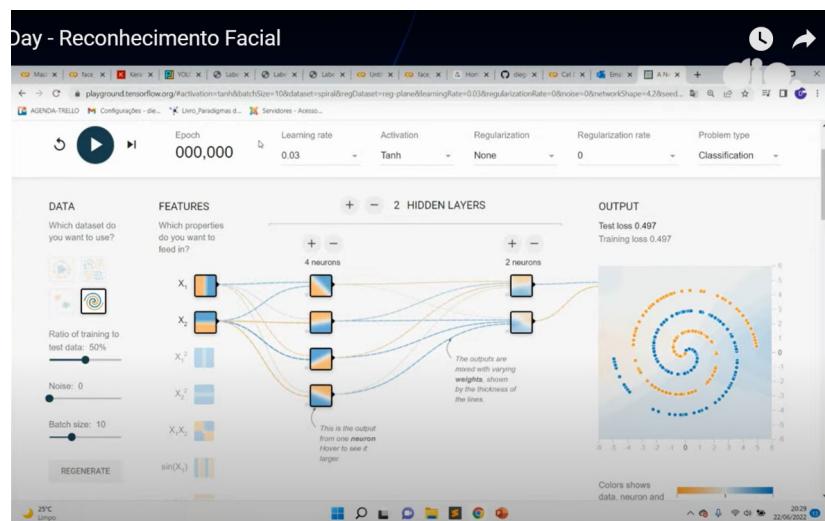
Problema mais complexo, demora mais para processar.



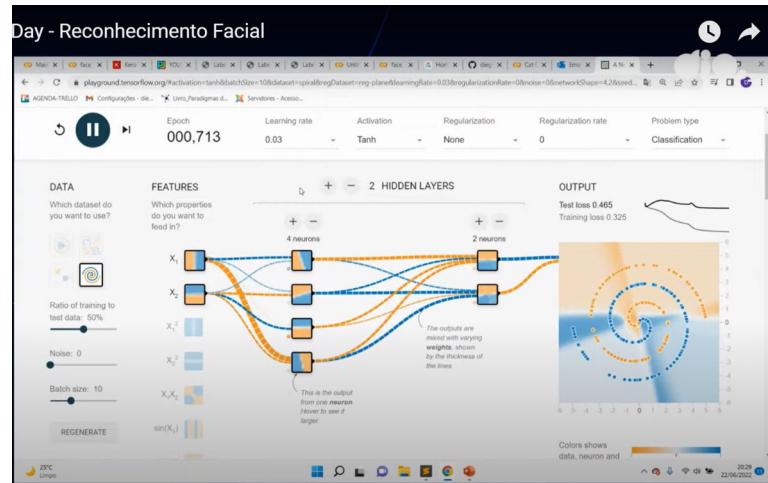
Resultado:



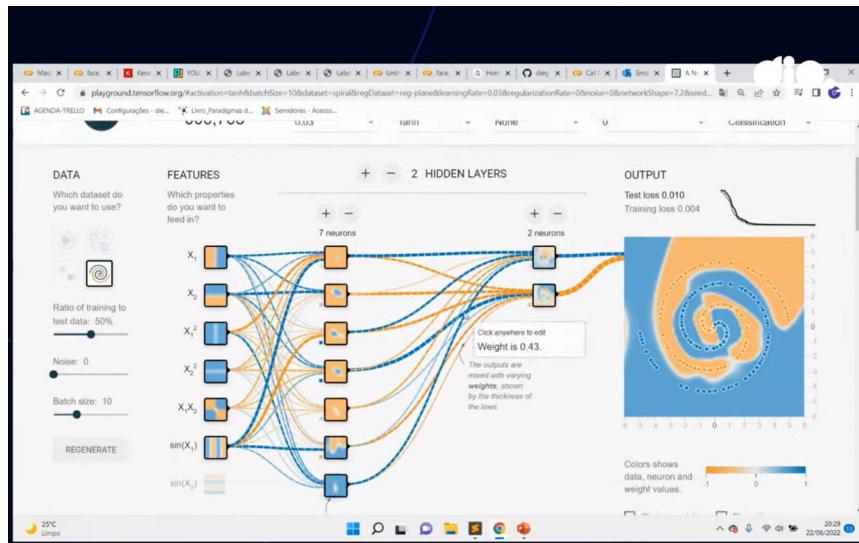
Problema clássico de espiral

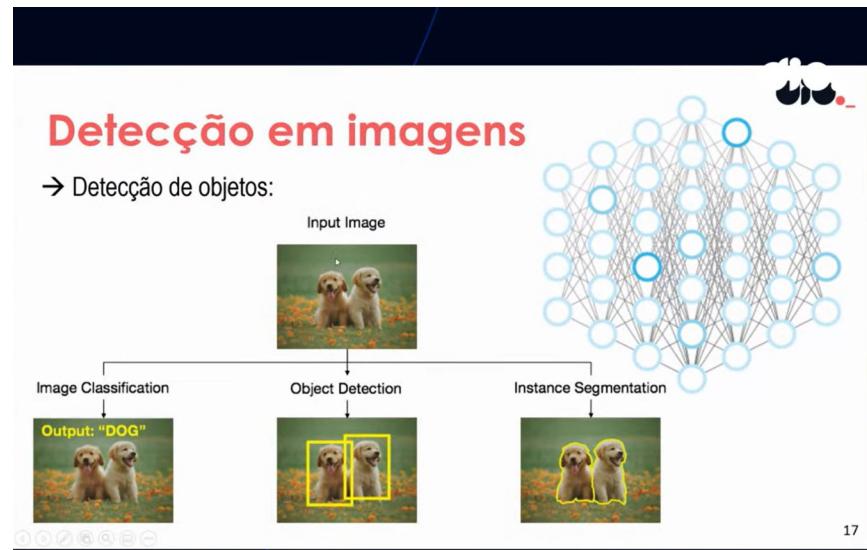


Resultado sem resposta, requer inclusão de mais neurônios e entradas na camada oculta da rede.



Resultado após acréscimo de neurônios:





Day - Reconhecimento Facial

Detecção de imagens

People with no idea about AI, telling me my AI will destroy the world Me wondering why my neural network is classifying a cat as a dog..

:26 / 2:58:26 Role para ver detalhes

Meme ilustra IA restrita que classifica incorretamente um gato como cachorro.

Day - Reconhecimento Facial

Como criar o Dataset?

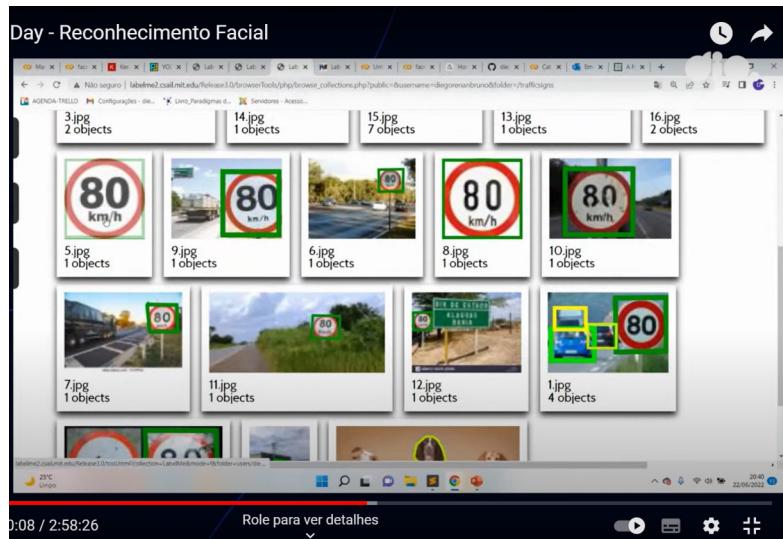
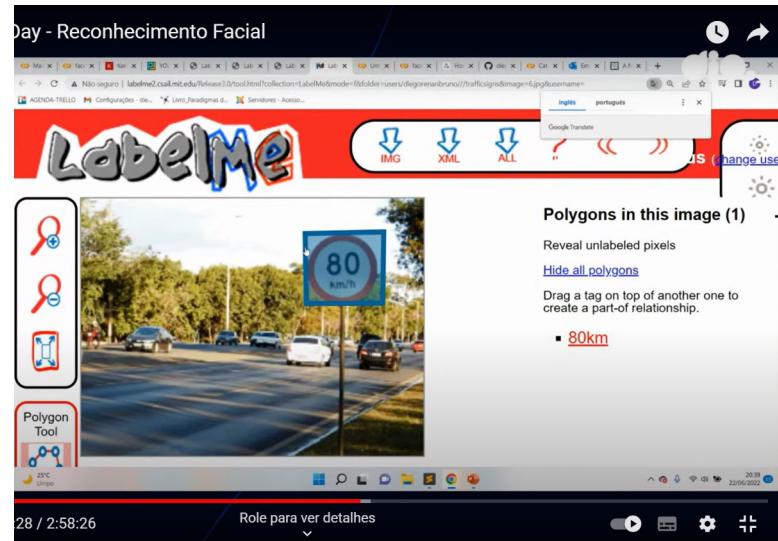
→ Criando sua base de dados:

:20 / 2:58:26 Role para ver detalhes

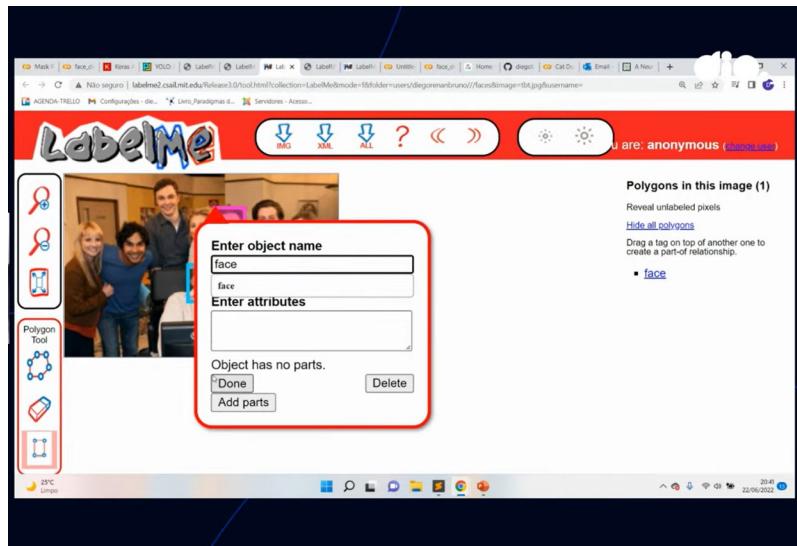
Machine Learning precisa de uma base de dados para o aprendizado. No caso de imagens, é preciso marcar cada pessoa para a IA aprender e reconhecer. Exemplo da base de dados do professor, composta por imagens de placas de trânsito.



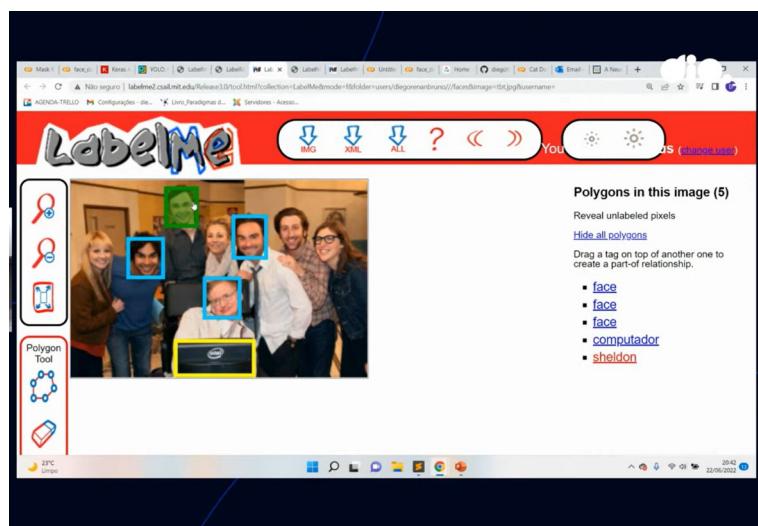
Para mostrar a placa para IA, é feito o contorno da placa e é apontada a localização da placa com certa característica (de 80 km/h). Não devemos mostrar toda a imagem, porque a IA não consegue aprender se houver muitos elementos.



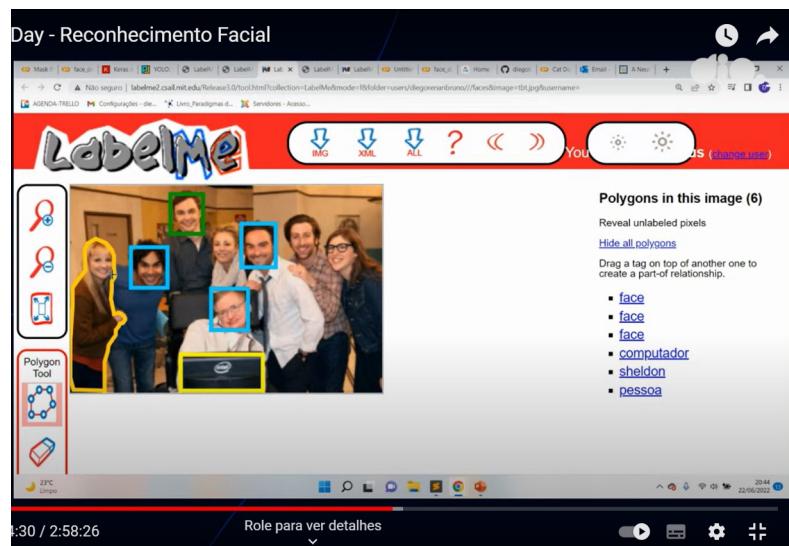
Não adianta mostrar só as placas recortadas, porque a IA precisa da informação de contexto da cena (posição, localização, formatos, semáforos com 3 focos coloridos etc.). Com ferramenta de rotulação, os rostos são rotulados e ensina-se à IA que é uma face (classe do objeto).



Criação de nova classe computador. É bom rotular todas as pessoas da imagem, pois quanto mais exemplos, melhor é para o aprendizado. É possível criar classe com nome da pessoa para reconhecimento específico.



No Google Street View, as faces das pessoas são ocultadas para preservar as identidades e manter a ética. Na segmentação de imagens, contorna-se toda a silhueta de uma pessoa e cria-se a classe pessoa para a IA reconhecer.



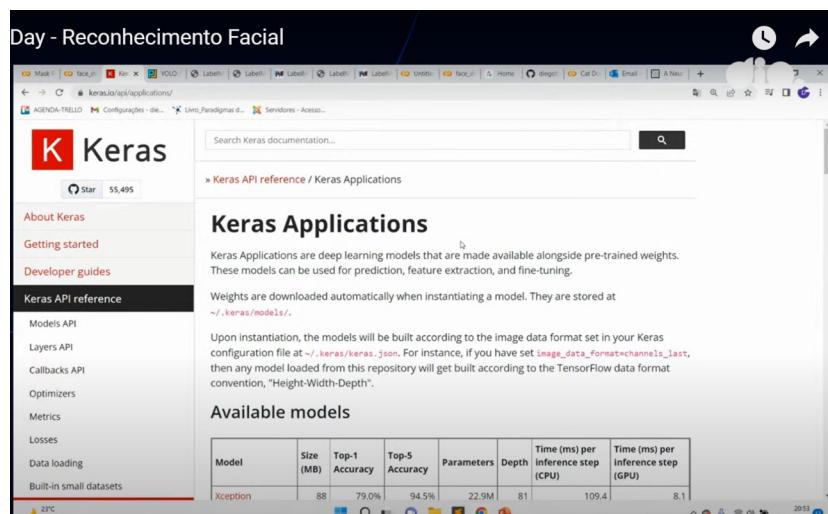
Para sistema de classificação é possível colocar imagens recortadas.



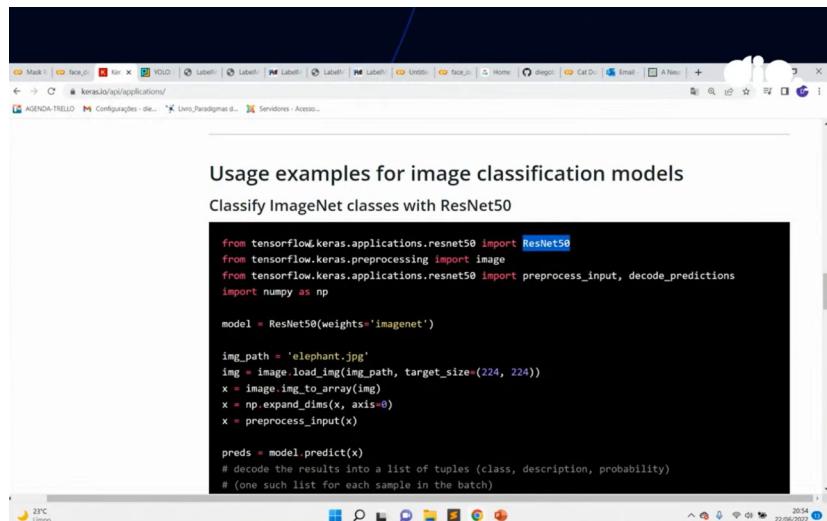
Para detecção de objeto, é necessário ter informações de contexto.



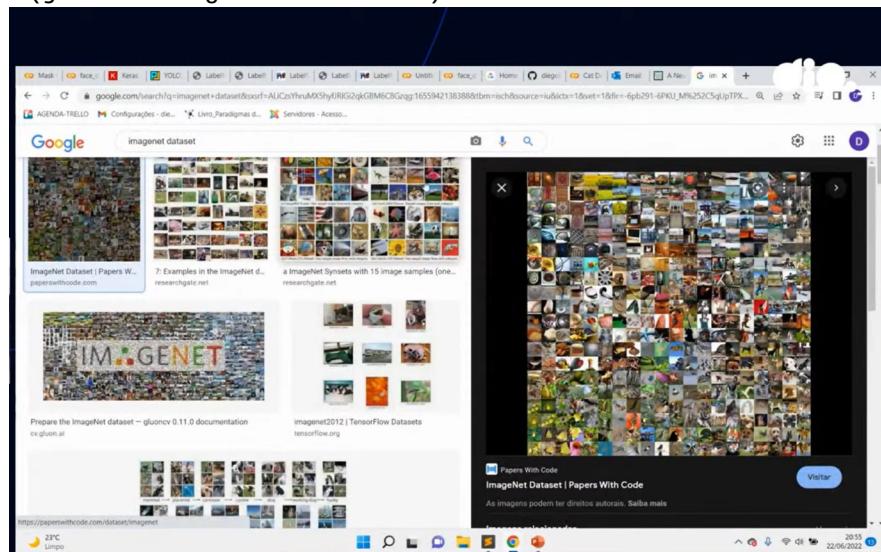
Depois de rotular a imagem, gera-se arquivo XML com as coordenadas X,Y das faces, passam-se a imagem e o XML que vai nortear o aprendizado do algoritmo. Keras Applications disponibiliza modelos prontos de redes neurais, que são rankeados por acurácia (% de acerto).



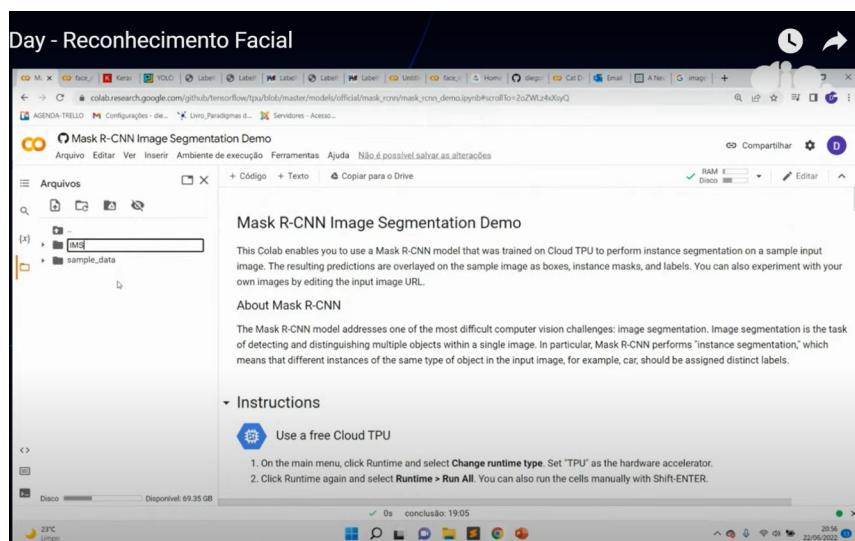
Pode copiar o código do modelo, importar a rede e utilizar toda sua estrutura.



Melhor rede neural no momento é a EfficientNetV2L com 97% de acurácia, baseada no dataset ImageNet (já tem imagens rotuladas).



Modelo de reconhecimento facial



Day - Reconhecimento Facial

```

from IPython import display
import numpy as np
TensorFlow_version 1.x
import tensorflow as tf
import sys
sys.path.insert(0, 'tpu/models/official')
sys.path.insert(0, 'tpu/models/official/mask_rcnn')
import coco_metric
from mask_rcnn.object_detection import visualization_utils

```

Load the COCO index mapping

This Colab uses a pretrained checkpoint of the Mask R-CNN model that is trained using the COCO dataset. Here is the mapping between the indices that the model predicts and the categories in text.

```

ID_MAPPING = {
    1: "person",
    ...
}

```

Classes treinadas

```

ID_MAPPING = {
    1: "person",
    2: "bicycle",
    3: "car",
    4: "motorcycle",
    5: "airplane",
    6: "bus",
    7: "train",
    8: "truck",
    9: "boat",
    10: "traffic light",
    11: "fire hydrant",
    12: "stop sign",
    13: "parking meter",
    15: "bench",
}

```

Aponta para imagem armazenada no drive do Google Colab.

```

!wget https://upload.wikimedia.org/wikipedia/commons/thumb/0/08/Kitano_Street_Kobe01ss5s4110.jpg/2560px-Kitano_Street_Kobe01ss5s4110.jpg
image_path = "./IMAGENS/chaves-4.jpg"

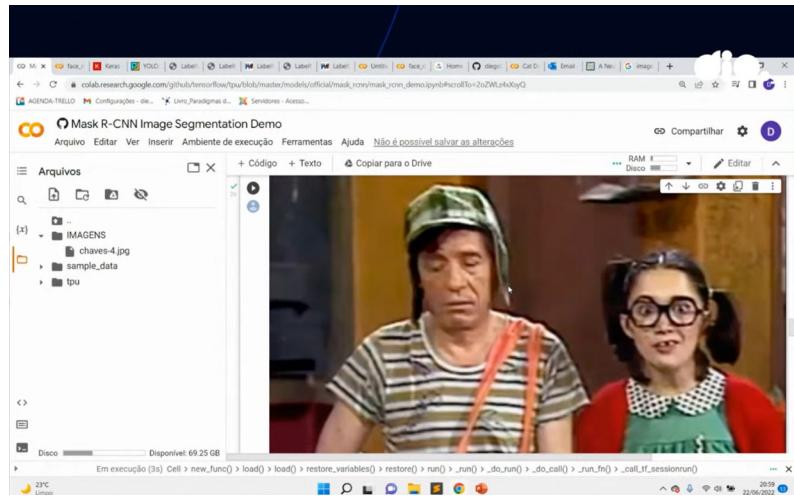
with open(image_path, 'rb') as f:
    np_image_string = np.array([f.read()])

image = Image.open(image_path)
width, height = image.size
np_image = np.array(image.getdata()).reshape(height, width, 3).astype(np.uint8)

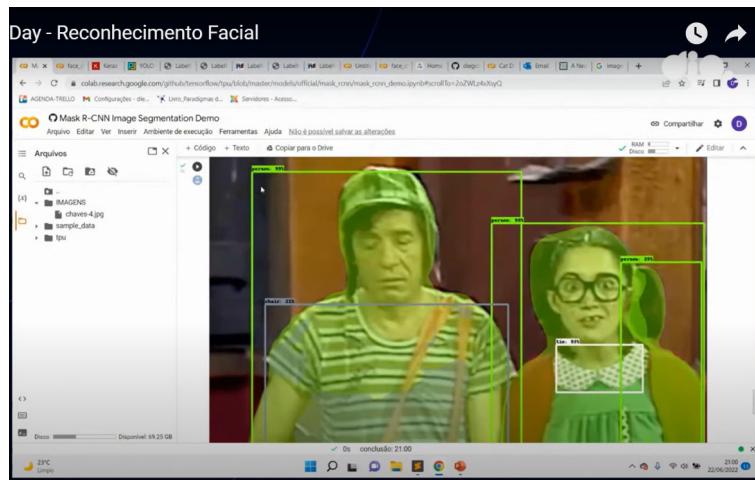
display.display(display.Image(image_path, width=1024))

```

Executa o código para obter a imagem de treinamento.



Resultado da detecção:

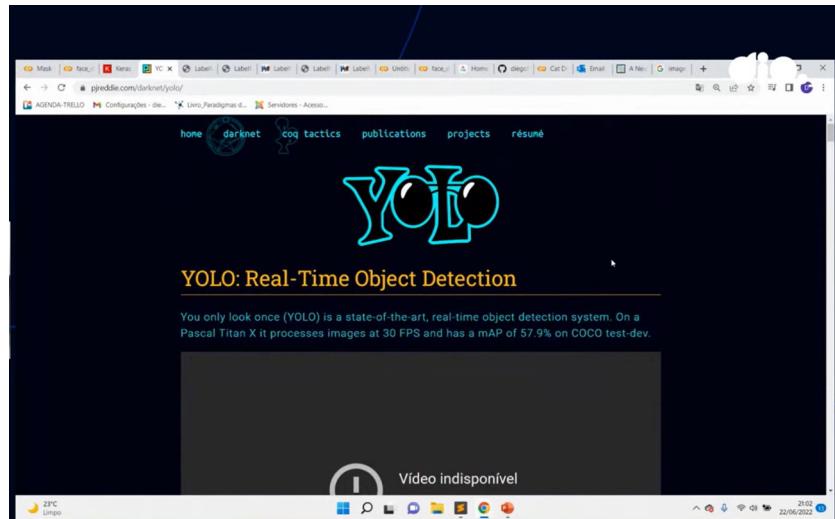


Apresenta erros: gola do vestido da Chiquinha foi rotulada como uma gravata e a camiseta do Chaves como uma cadeira. Podem ser tratados gerando um número de corte, geralmente desconsidera tudo abaixo de 50% de certeza.

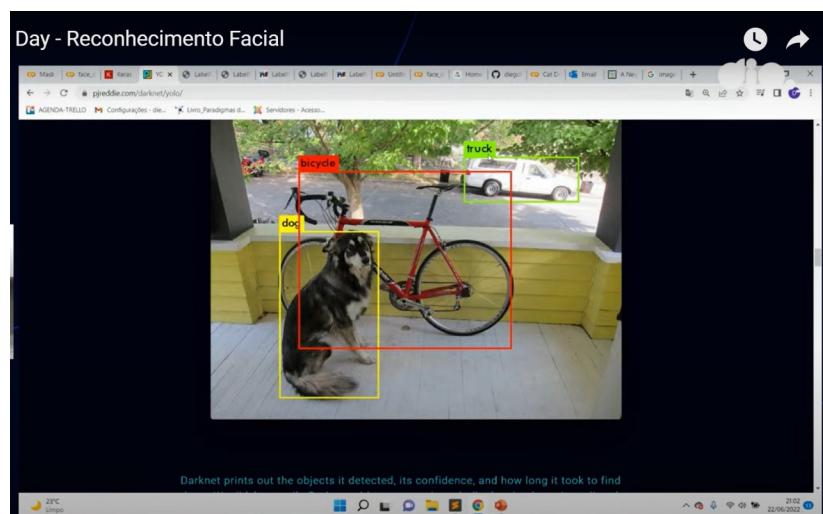
Aplicação de uma rede de deep learning, a rcnn, para detecção e segmentação.

```
Day - Reconhecimento Facial
[1]: use_tpu = False #@param {type:"boolean"} use_tpu: false
[2]: import os
[3]: import pprint
[4]: assert 'COLAB_TPU_ADDR' in os.environ, 'ERROR: not connected to a TPU! ADDRESS = %s' % os.environ['COLAB_TPU_ADDR']
[5]: print('TPU address is:', TPU_ADDRESS)
[6]: session = tf.Session(tpu_address=TPU_ADDRESS, graph=tf.Graph())
[7]: print('TPU devices:')
[8]: pprint.pprint(session.list_devices())
[9]: else:
[10]: session = tf.Session(graph=tf.Graph())
[11]: 
[12]: # Load the pretrained model
[13]: Loading the COCO pretrained saved model from the public GCS bucket.
[14]: 
[15]: saved_model_dir = 'gs://cloud-tpu-checkpoints/mask-rcnn/1555659859': saved_model_dir: "gs://cloud-tpu-checkpoints/mask-rcnn/1555659859"
[16]: + tf.saved_model.loader.load(session, ["serve"], saved_model_dir)
[17]: 
[18]: WARNING:tensorflow:from tensorflow.python.saved_model.loader_impl import load (from tensorflow.python.saved_model.loader_impl) is deprecated and will be removed
[19]: INSTRUCTIONS for updating:
[20]: This function will only be available through the v1 compatibility library as tf.compat.v1.saved_model.loader.load or tf.compat.v1.saved_model.lo
```

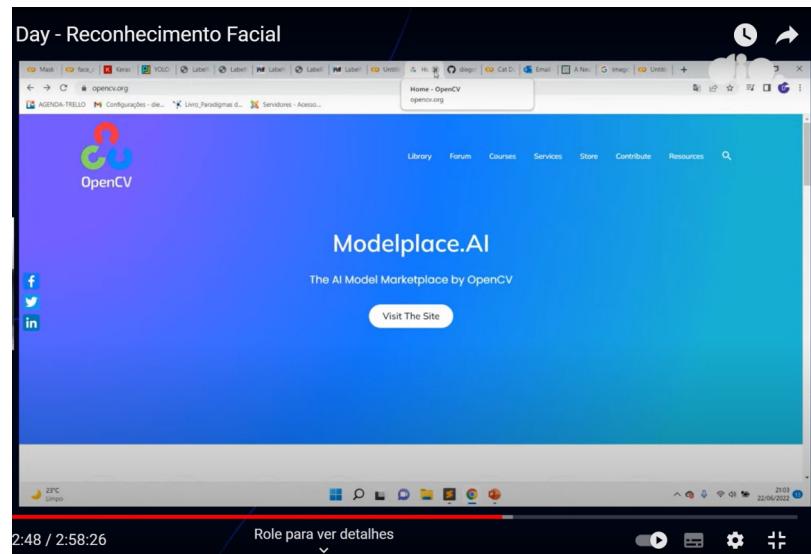
Outra opção de rede neural é a Yolo, que faz apenas detecção.



Resultado da Yolo



Biblioteca OpenCV para processamento de imagens com deep learning.



Depois do treinamento inicial da rede neural, onde se indica a face na imagem, pode ser feito outro treinamento com imagens rotuladas e indicar que o objeto é uma pessoa identificada pelo seu nome.

```

    numpy 1.18.1
    tensorflow 2.4.4
    matplotlib 2.2.2
    keras 2.2.4

    7. Transfer Learning

    [ ] %matplotlib inline
    %config InlineBackend.figure_format = 'retina'
    import matplotlib.pyplot as plt
    plt.rcParams['figure.figsize'] = (18,6)

    Load Dataset from KAGGLE DATA BASE
    [ ] import os
    import glob
    import zipfile
    import functools
  
```

Como realizar treinamento em uma base de dados nova para um modelo novo? Com Transfer Learning treina-se novamente uma rede já treinada, com base no treinamento passado, que é a transferência de aprendizado. Assim se aproveita todo o treinamento passado, filtros ajustados, tudo que pode ajudar no processo de treinamento futuro. Exemplo: tentativa de usar foto de celular em modelo treinado para verificar se ocorre o erro de detectar a imagem do rosto como uma face humana. Para o professor, houve reconhecimento com 99,97% para imagem da face e 71.82% para foto de sua face no celular. A detecção errada da imagem estática foi considerada por estar acima do limiar de 50%. Esse tipo de erro pode ser evitado com adoção de recursos de movimento no sistema de reconhecimento (pedir pra mover rosto em várias direções). Alguns celulares utilizam 2 câmeras ou laser para verificar se o rosto é plano ou tridimensional.

