CLASSIFICAÇÃO DO COMPORTAMENTO DO MOTORISTA A PARTIR DE IMAGENS UTILIZANDO APRENDIZADO DE MÁQUINA PROFUNDO

Nome: Morsinaldo Medeiros
Docente: Adriao Duarte Doria Neto
Disciplina: Inteligência Artificial

DCA

UFRN
UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

O Problema

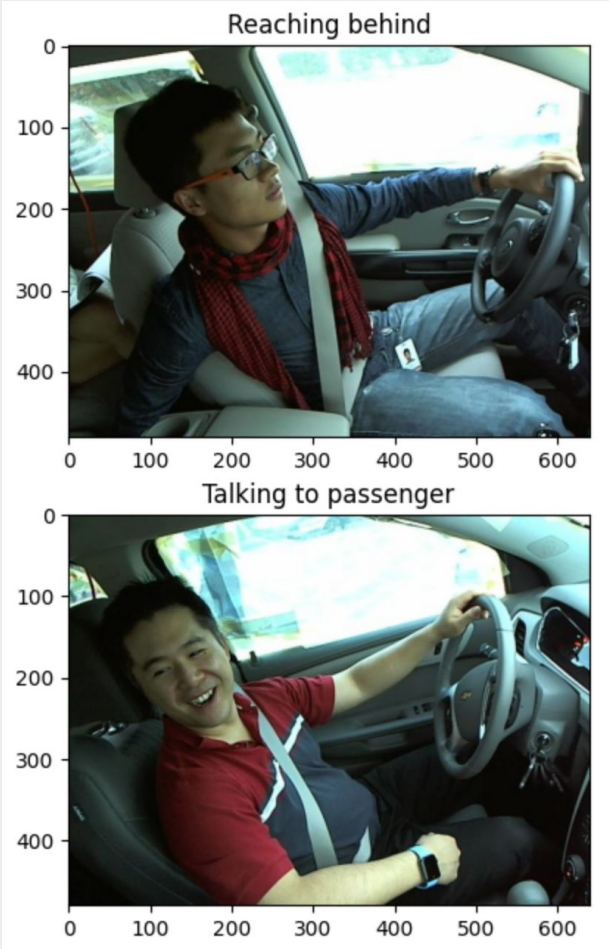
Os motoristas querem fazer outras atividades enquanto dirigem.

Alguns veículos estão embarcando câmeras para monitorar o motorista.

Embarcar um modelo de IA capaz de classificar o estado do motorista.

O Dataset

Classe	Label
c0	Safe Driving
c1	Texting - Right
c2	Talking on the phone - Right
c3	Texting - Left
c4	Talking on the phone - Left
c5	Operating the radio
c6	Drinking
c7	Reaching behind
c8	Hair and makeup
c9	Talking to passenger



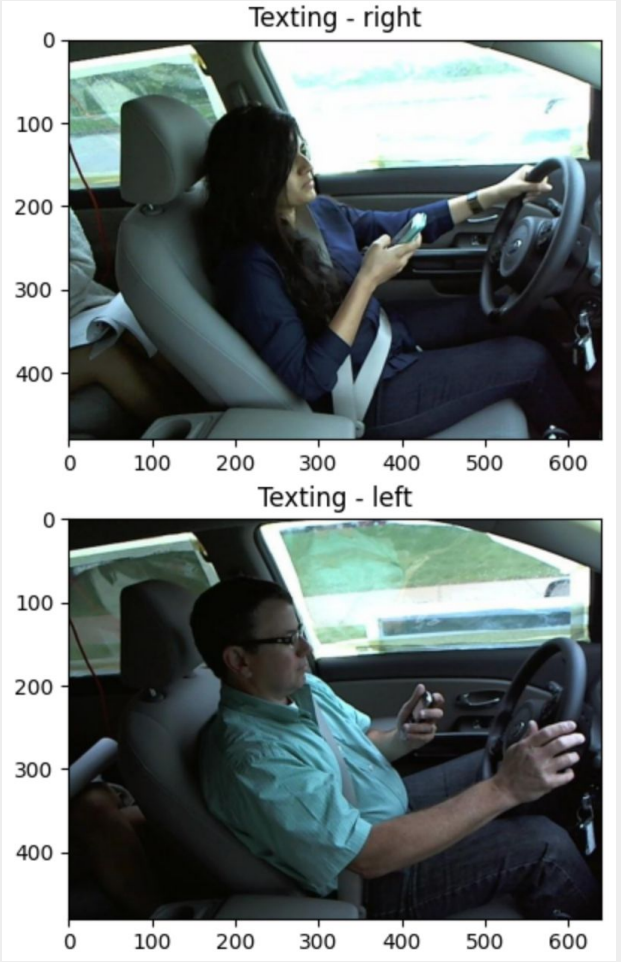
kaggle

22424 imagens

https://www.kaggle.com/c/state-farm-distracted-driver-detection/data

O Dataset

Classe	Label
c0	Safe Driving
c1	Texting - Right
c2	Talking on the phone - Right
c3	Texting - Left
c4	Talking on the phone - Left
c5	Operating the radio
c6	Drinking
c7	Reaching behind
c8	Hair and makeup
c9	Talking to passenger



kaggle

22424 imagens

https://www.kaggle.com/c/state-farm-distracted-driver-detection/data

O Dataset

Classe	Label
c0	Safe Driving
c1	Texting - Right
c2	Talking on the phone - Right
c3	Texting - Left
c4	Talking on the phone - Left
c5	Operating the radio
c6	Drinking
c7	Reaching behind
c8	Hair and makeup
c9	Talking to passenger

kaggle

22424 imagens

https://www.kaggle.com/c/state-farm-distracted-driver-detection/data

O Dataset

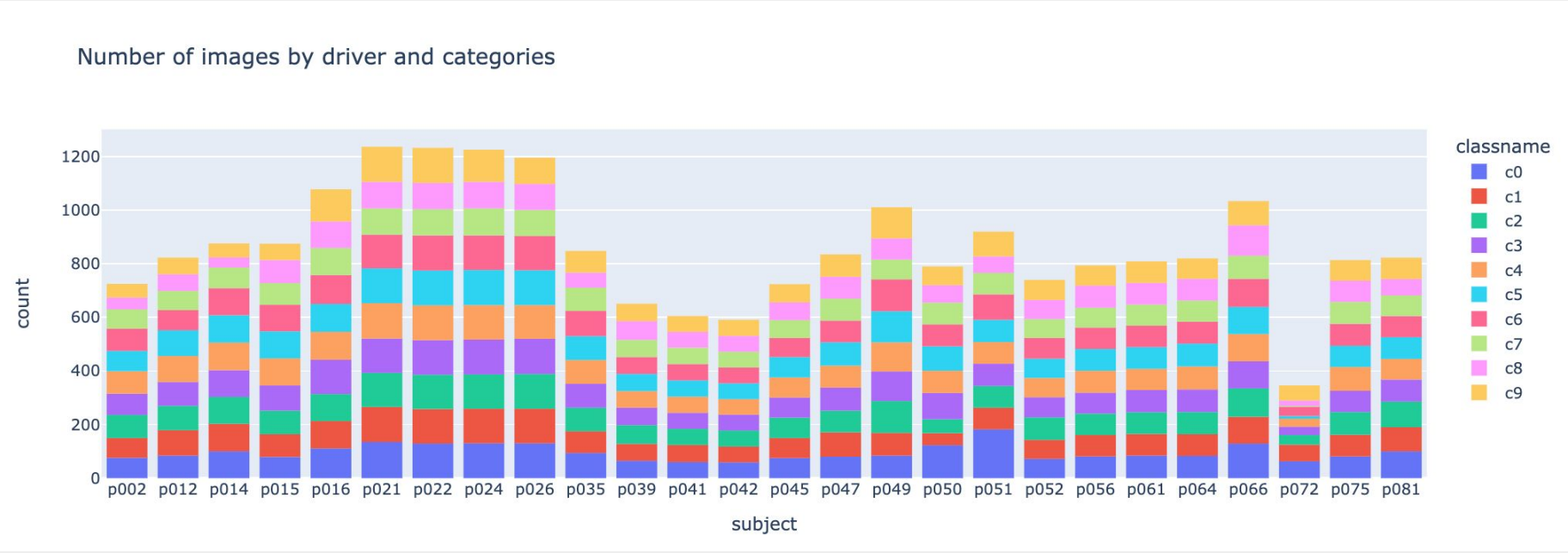
Classe	Label
c0	Safe Driving
c1	Texting - Right
c2	Talking on the phone - Right
c3	Texting - Left
c4	Talking on the phone - Left
c5	Operating the radio
c6	Drinking
c7	Reaching behind
c8	Hair and makeup
c9	Talking to passenger

kaggle

22424 imagens

https://www.kaggle.com/c/state-farm-distracted-driver-detection/data

O dataset



kaggle

22424 imagens

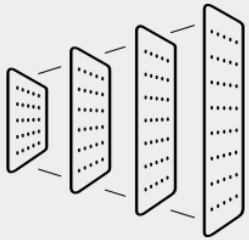
26 motoristas
10 classes

0.8 → 17.940 img. - treinam.

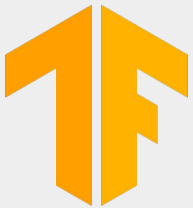
0.2 → 4484 img. - teste

O Treinamento

Modelo 1 - Baseline



```
DIM = 256
NB_CHANNELS = 3
NB_CLASSES = 10
```

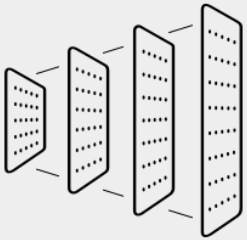


```
Total params: 58,097,922
Trainable params: 58,094,854
Non-trainable params: 3,068
```

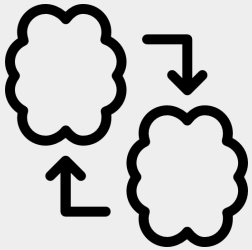
```
base_model = Sequential()
base_model.add(Conv2D(32, (3, 3), padding="same",
                      input_shape = (DIM , DIM , NB_CHANNELS)))
base_model.add(Activation("relu"))
base_model.add(BatchNormalization(axis=1))
base_model.add(MaxPooling2D(pool_size=(3, 3)))
base_model.add(Conv2D(64, (3, 3), padding="same"))
base_model.add(Activation("relu"))
base_model.add(BatchNormalization(axis=1))
base_model.add(Conv2D(64, (3, 3), padding="same"))
base_model.add(Activation("relu"))
base_model.add(BatchNormalization(axis=1))
base_model.add(MaxPooling2D(pool_size=(2, 2)))
base_model.add(Conv2D(128, (3, 3), padding="same"))
base_model.add(Activation("relu"))
base_model.add(BatchNormalization(axis=1))
base_model.add(Conv2D(128, (3, 3), padding="same"))
base_model.add(Activation("relu"))
base_model.add(BatchNormalization(axis=1))
base_model.add(MaxPooling2D(pool_size=(2, 2)))
base_model.add(Flatten())
base_model.add(Dense(1024))
base_model.add(Activation("relu"))
base_model.add(BatchNormalization())
base_model.add(Dense(10))
base_model.add(Activation("softmax"))
base_model.build((0,256,256,3))
base_model.summary()
```

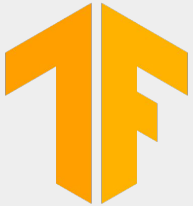
O Treinamento

Modelo 2 - ResNet-50



```
DIM = 256
NB_CHANNELS = 3
NB_CLASSES = 10
```



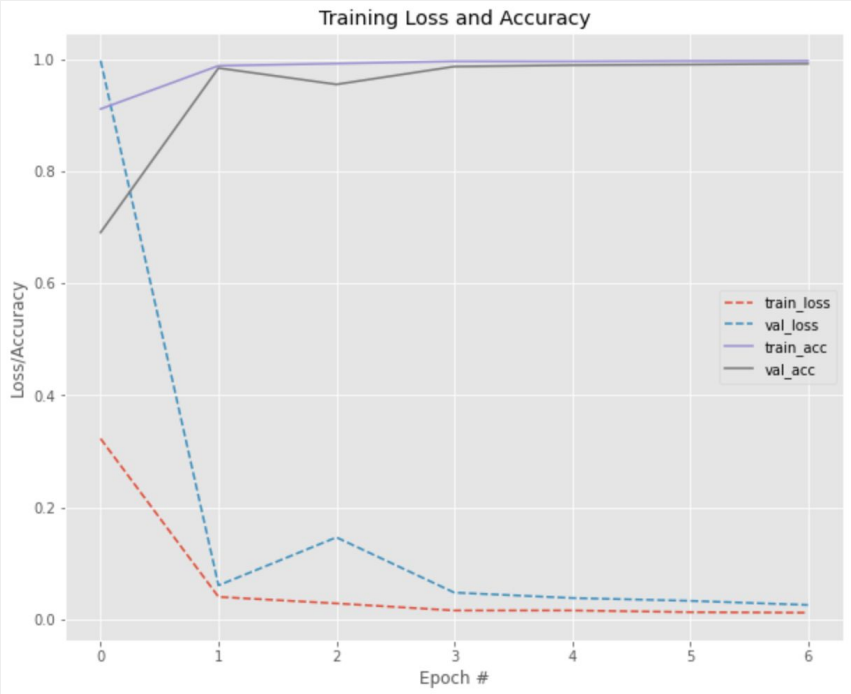


```
Total params: 24,770,698
Trainable params: 24,717,578
Non-trainable params: 53,120
```

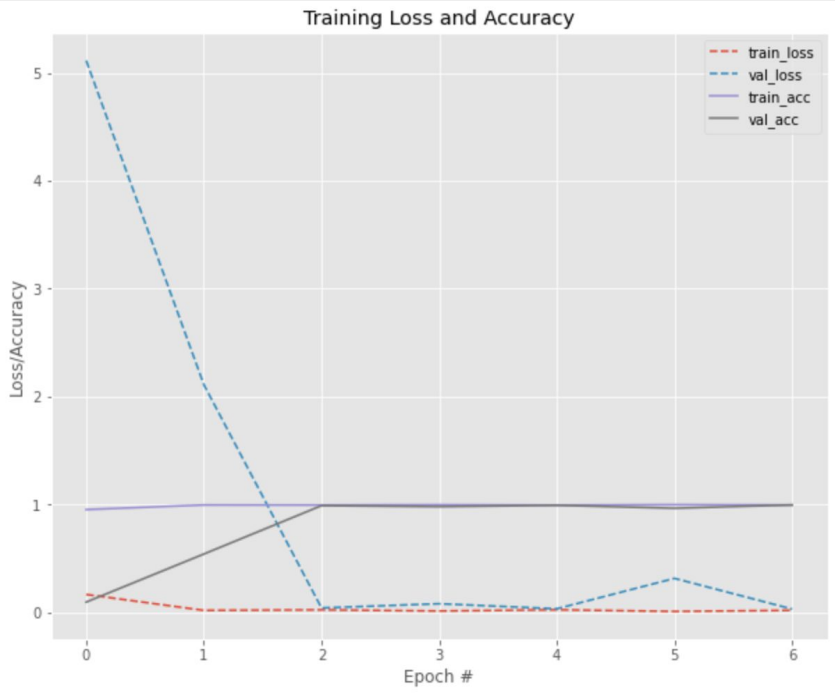
```
encoder_resnet50 = tf.keras.applications.ResNet50(include_top=False,
                                                   weights='imagenet',
                                                   input_shape=(DIM,DIM ,NB_CHANNELS))
new_model = Sequential(name = 'encoder_resnet_50')
new_model.add(encoder_resnet50)
new_model.add(GlobalAveragePooling2D())
new_model.add(Dense(512))
new_model.add(Dense(256))
new_model.add(Dense(10, activation='softmax'))
resnet50 = new_model
resnet50.summary()
```

```
Model: "encoder_resnet_50"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resnet50 (Functional)       (None, 8, 8, 2048)        23587712

 global_average_pooling2d (G  (None, 2048)             0
 lobalAveragePooling2D)

 dense_2 (Dense)             (None, 512)               1049088

 dense_3 (Dense)             (None, 256)               131328

 dense_4 (Dense)             (None, 10)                2570
```

O Treinamento



Baseline



ResNet-50

O Treinamento

Baseline

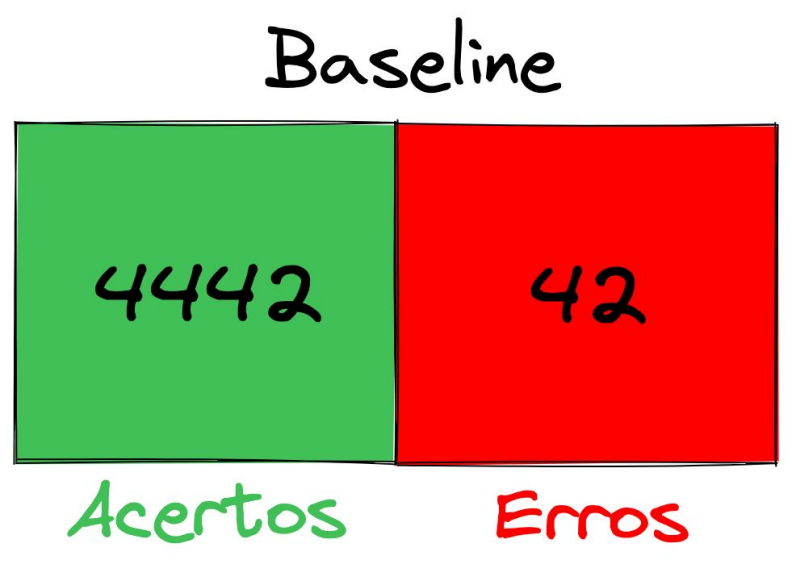
```
[INFO] 0.07547938452986881 kWh of electricity used since the begining
[INFO] Energy consumed for RAM: 0.0039195619514554 99 kWh
[INFO] Energy consumed for all GPU: 0.036508708096063 44 kWh
[INFO] Energy consumed for all CPU: 0.035051114482349 84 kWh
[INFO] CO2 emission 0.047626220824339746(in Kg)
```

ResNet-50

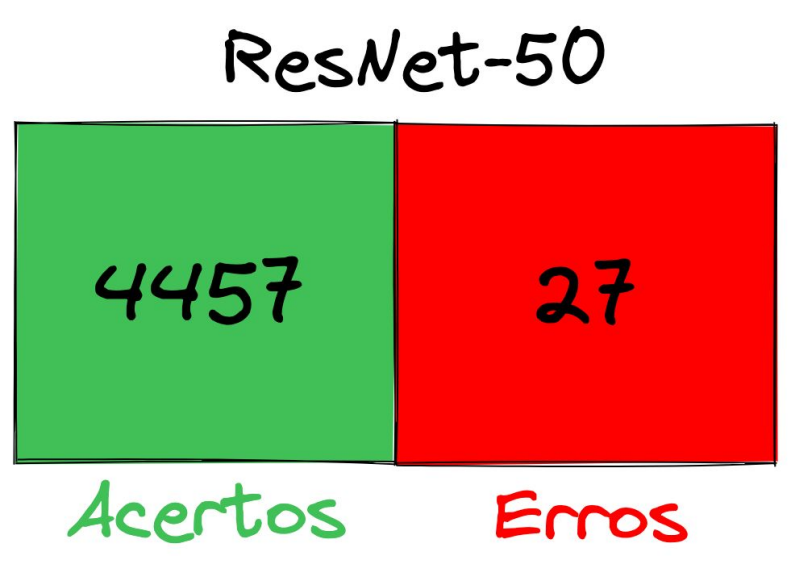
```
[INFO] 0.09873487794682977 kWh of electricity used since the begining
[INFO] Energy consumed for RAM: 0.0045570970551078 kWh
[INFO] Energy consumed for all GPU: 0.053422190006672136 kWh
[INFO] Energy consumed for all CPU: 0.040755590885049 78 kWh
[INFO] CO2 emission 0.012736179872663017(in Kg)
```

CODE CARBON

Os Resultados



Acc: 0.9906



Acc: 0.9940

CLASSIFICAÇÃO DO COMPORTAMENTO DO MOTORISTA A PARTIR DE IMAGENS UTILIZANDO APRENDIZADO DE MÁQUINA PROFUNDO

Nome: Morsinaldo Medeiros
Docente: Adriao Duarte Doria Neto
Disciplina: Inteligência Artificial

Obrigado pela atenção!

Email: morsinaldo.medeiros.075@ufrn.edu.br
Github: https://github.com/Morsinaldo