Explicabilidade em Modelos de Redes Neurais Convolucionais

Disciplina
INF 2064 - Visão Computacional - **Professor: Marcelo Gattass**
PUC-Rio - 2022.2

Aluno Extraordinário
Daniel da Silva Costa
danieldasilvacosta@gmail.com

Problema

- Apesar da grande utilização pela academia e pela indústria dos modelos de Aprendizagem Profunda e reconhecendo a sua capacidade em extrair características dos dados e classificá-los, a utilização desses modelos sofre de uma melhor facilidade em permitir aos usuários e praticantes entenderem mais claramente as regras internas desses modelos sobre quais informações estes se baseiam para realizar as classificações e as decisões.

Objetivo desta Pesquisa

- Explorar a explicabilidade em modelos de Aprendizagem Profunda baseados em convolução. (*)

(*) A escolha de redes baseadas em convoluções se deve ao fato do presente trabalho estar inserido em uma disciplina de Visão Computacional, área do conhecimento que tem utilizado largamente esse tipo de rede neural artificial.

Explicação (RAS et al., 2022)

*“De maneira geral, uma explicação é **qualquer informação** que **ajuda o usuário** a entender e a comunicar porque o modelo exibe algum **padrão de tomada de decisão** e como as decisões individuais ocorrem.”*

Explainability?

Interpretability?

Model Understanding?

Alguns Benefícios da Visualização

- **Verificar quais as áreas da imagem** que estão sendo priorizadas pelo modelo.
- Observar se o modelo está dando maior importância às **áreas do objeto-alvo** ou se está aprendendo áreas do entorno do objeto.
- Possibilitar um **melhor entendimento sobre quais exemplos** devem ser utilizados de cada classe.
- (RAS et al., 2022).
  - "ganhar *insight* sobre como a informação é extraída dos dados em **diferentes camadas da rede**."
  - Informação adicional que ajude a decidir melhor sobre, por exemplo, **a quantidade de dados rotulados, os valores de hiperparâmetros e a escolha de modelo**.

CAM - Class Activation Mapping

- Global Max Pooling (GMP) e **Global Average Pooling (GAP)**
- CAM - Class Activation Mapping (ZHOU et al., 2016)
  - Média ponderada das ativações da última camada de convolução para cada classe.
- Variações
  - Score-CAM e Grad-CAM

CAM - Class Activation Mapping

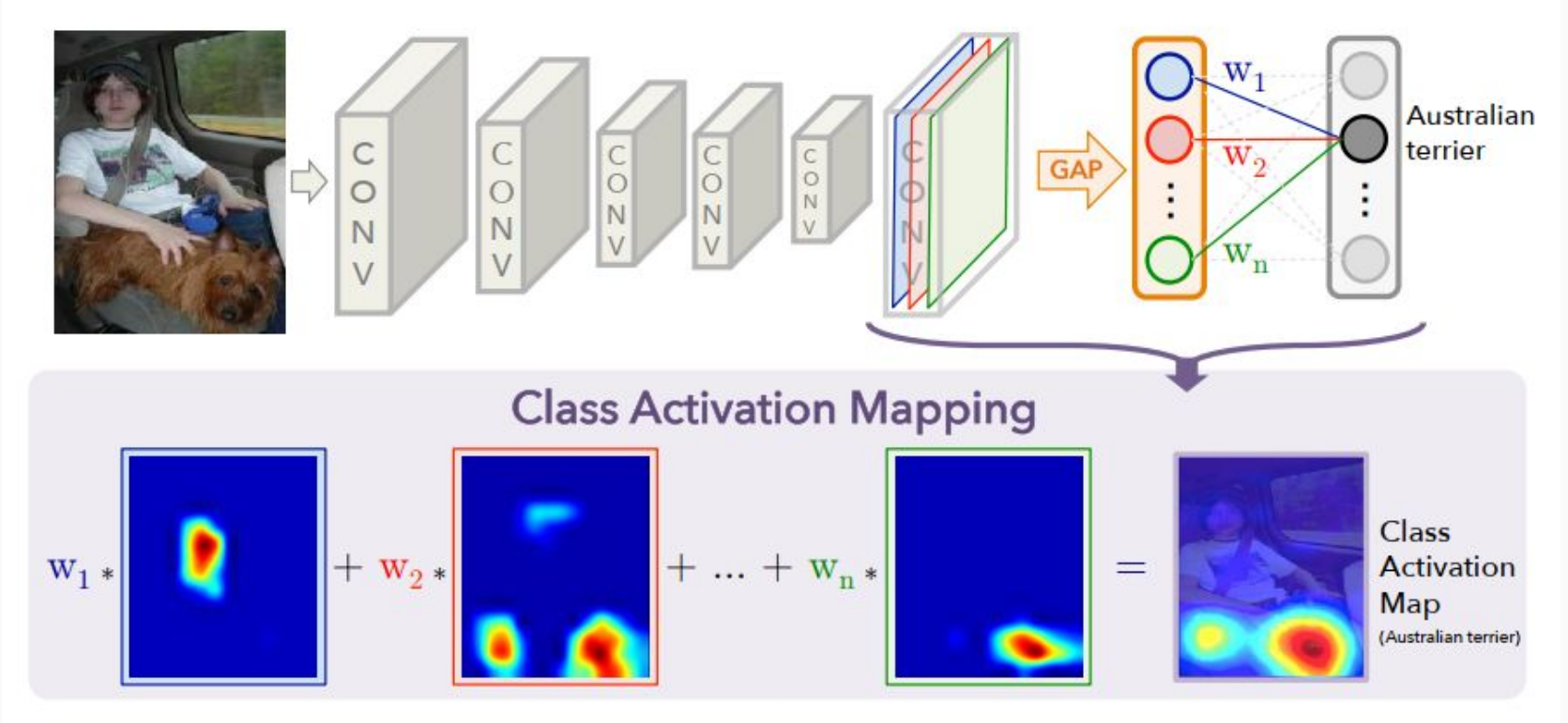


Imagem extraída de (ZHOU et al., 2016)

Arquitetura da Rede

```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 224, 224, 3)]     0
_________________________________________________________________
conv2d (Conv2D)              (None, 224, 224, 16)      208
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 112, 112, 16)      0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 112, 112, 32)      2080
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 56, 56, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 56, 56, 64)        8256
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 28, 28, 64)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 28, 28, 128)       32896
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 14, 14, 128)       0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 14, 14, 256)       131328
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 7, 7, 256)         0
_________________________________________________________________
flatten (Flatten)            (None, 12544)             0
_________________________________________________________________
dropout (Dropout)            (None, 12544)             0
_________________________________________________________________
dense (Dense)                (None, 10)                125450
=================================================================
Total params: 300,218
Trainable params: 300,218
Non-trainable params: 0
```

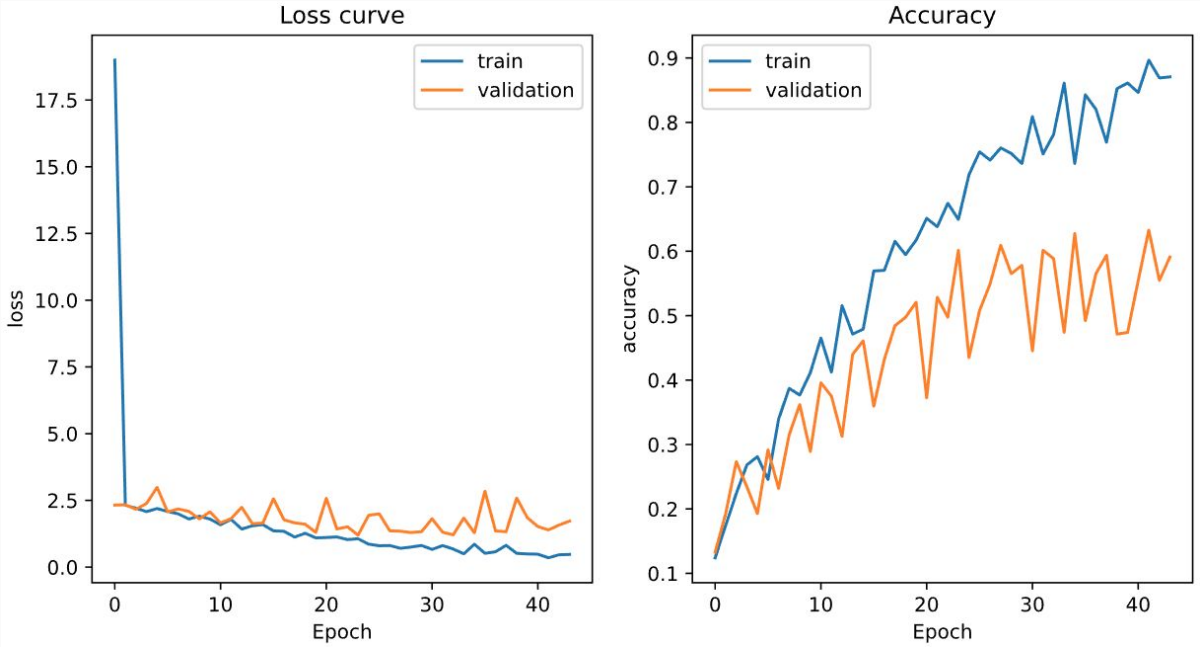
- *Activation: ReLU*
- *Filters: 16, 32, 64, 128, 256*
- *Dropout: 0.2*
- *strides: (1, 1)*
- *padding: same*
- *optimizer: RMSprop(0.001)*
- *last_conv_layer: conv2d_4*

Experimentos

- ***Kernel Sizes*: 2, 4, 8, 16, 32**
- *Batch Size*: 128
- Épocas
  - Máximo: 100
  - *EarlyStopping(patience* = 20)
  - *ModelCheckpoint(save_best_only = True)*
- *Dataset*
  - *Dataset* pequeno com imagens de macacos (10 classes).
  - Quantidade de exemplos de **treinamento: 1096**
  - Quantidade de exemplos de **validação: 272**
  - As imagens passaram por um **pré-processamento e por *Data Augmentation***.
    - Formato: JPEG.
    - Dimensões de saída: **224 x 224 px**
    - 3 canais (RGB)

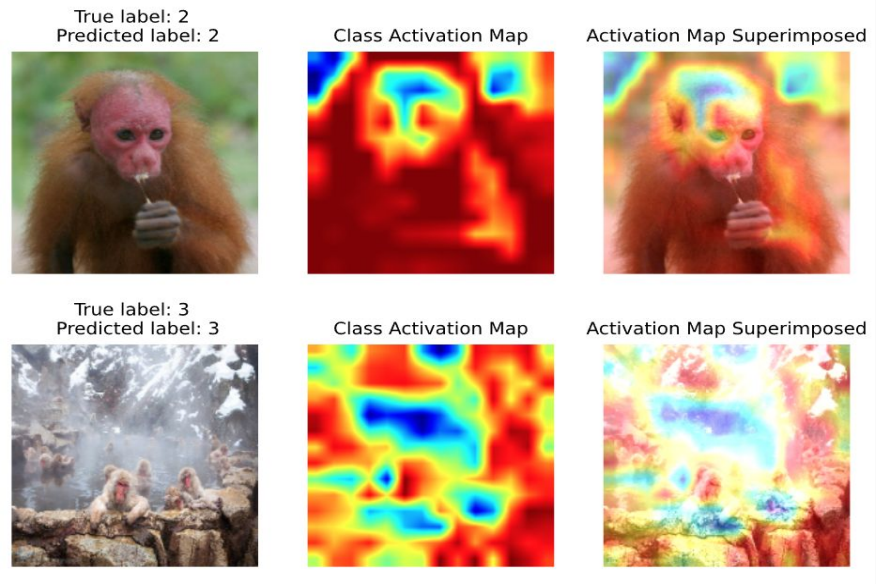
Resultados

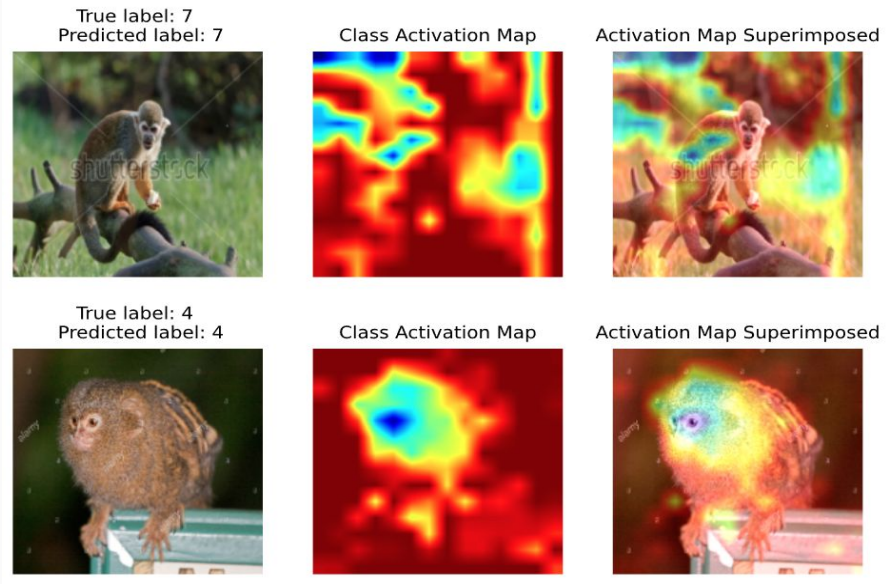
kernel_size = (2, 2)



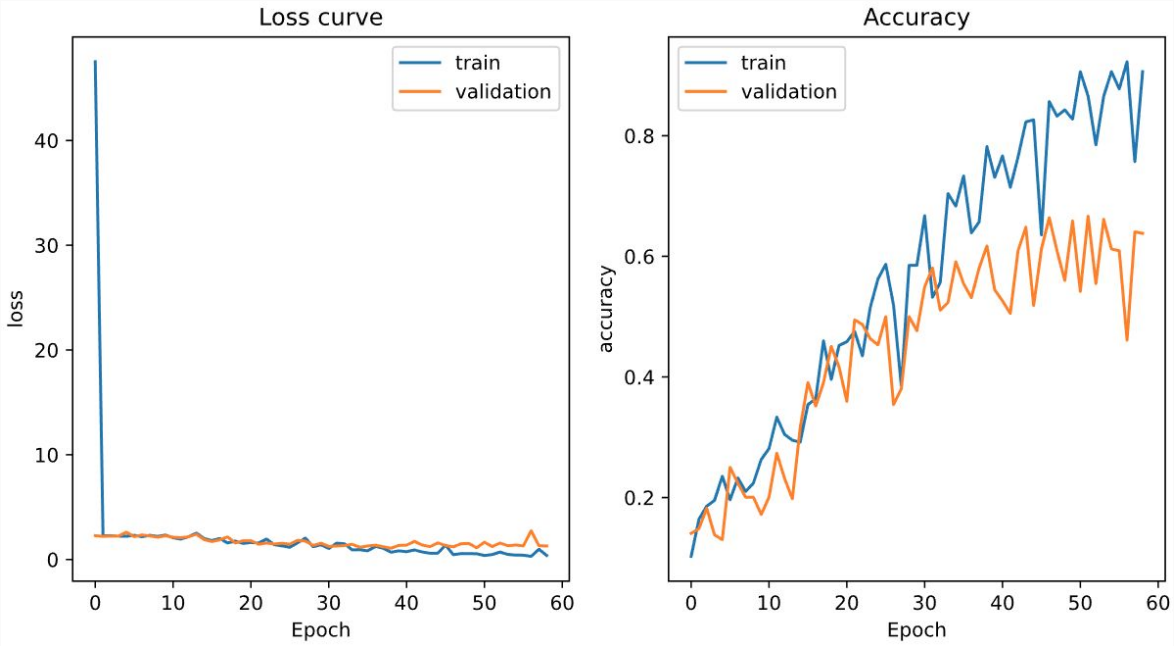
- ***Max Epoch 44/100***
- *Wall time: 19min 25s*
- ***Final validation accuracy: 58.33%***

kernel_size = (2, 2)



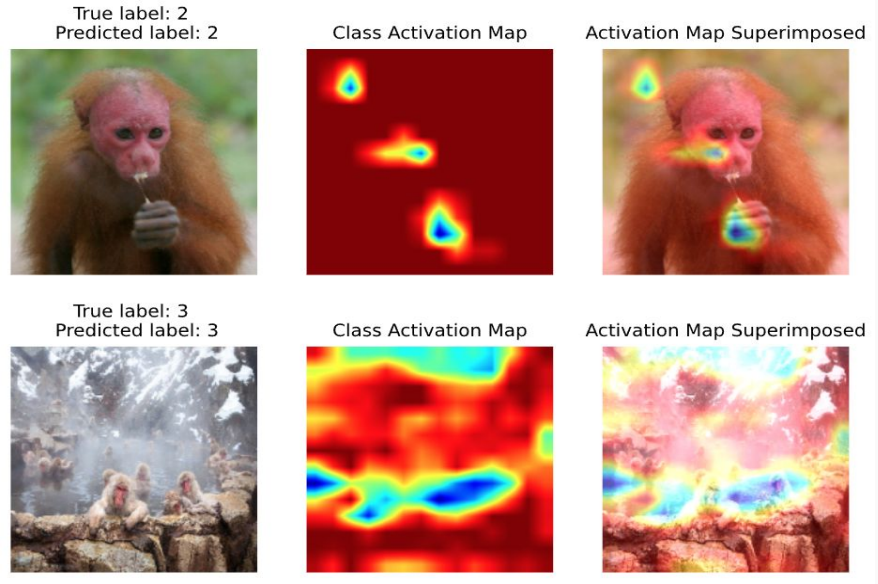


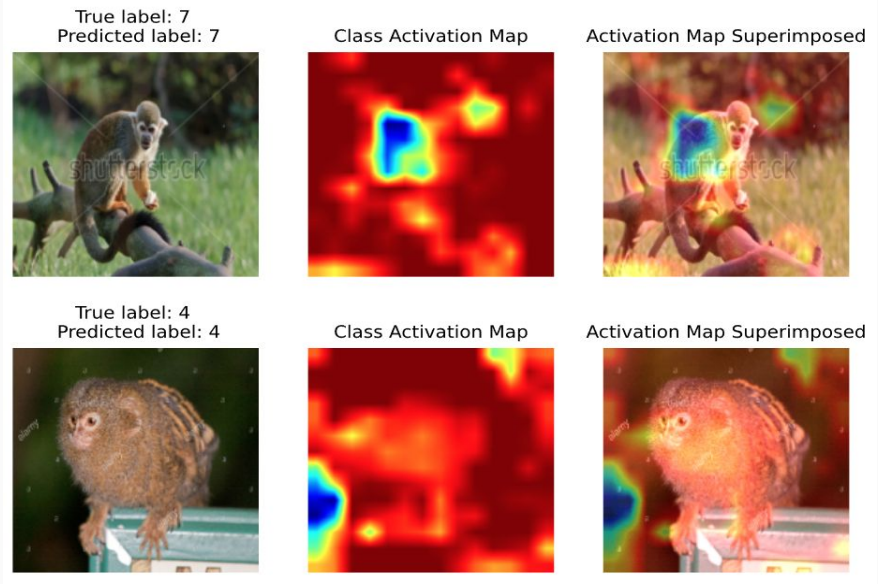
kernel_size = (4, 4)



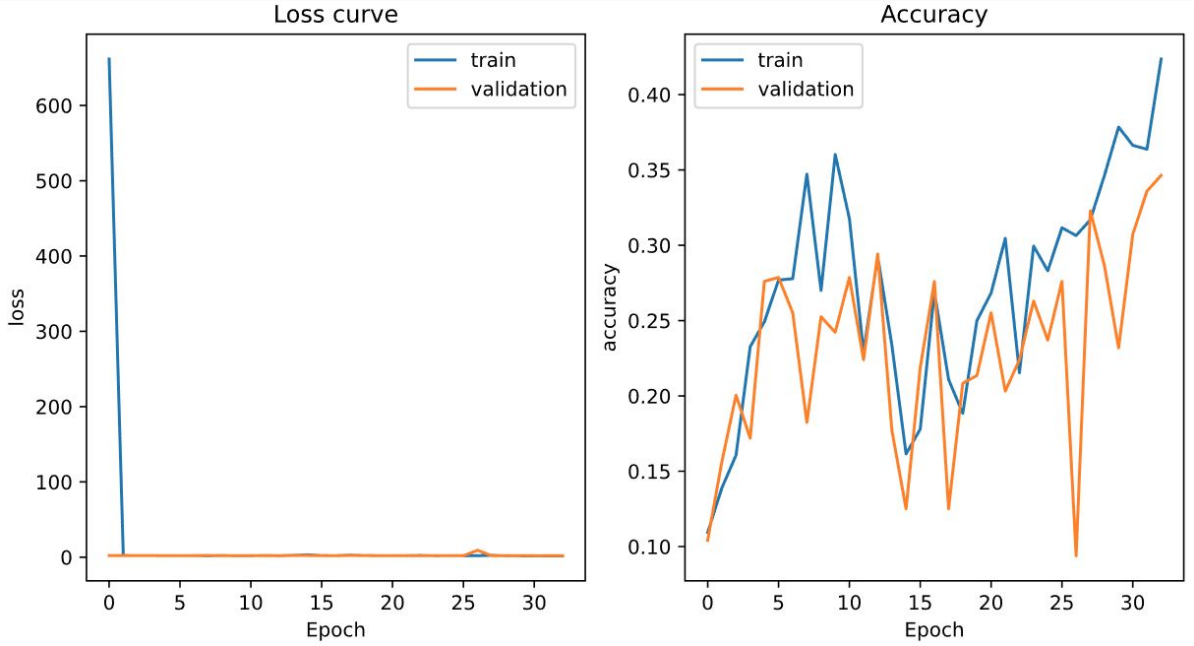
- ***Max Epoch 59/100***
- *Wall time: 26min 16s*
- ***Final validation accuracy: 61.46%***

kernel_size = (4, 4)



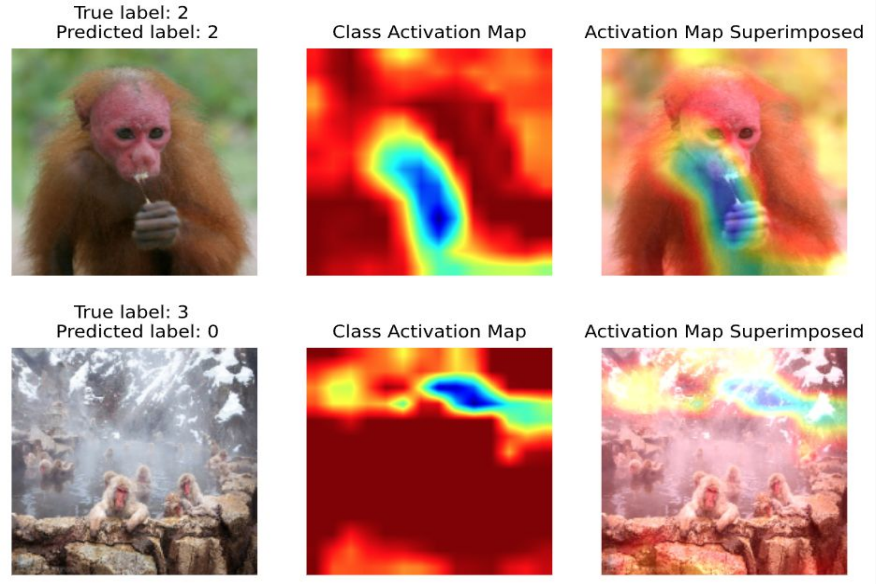


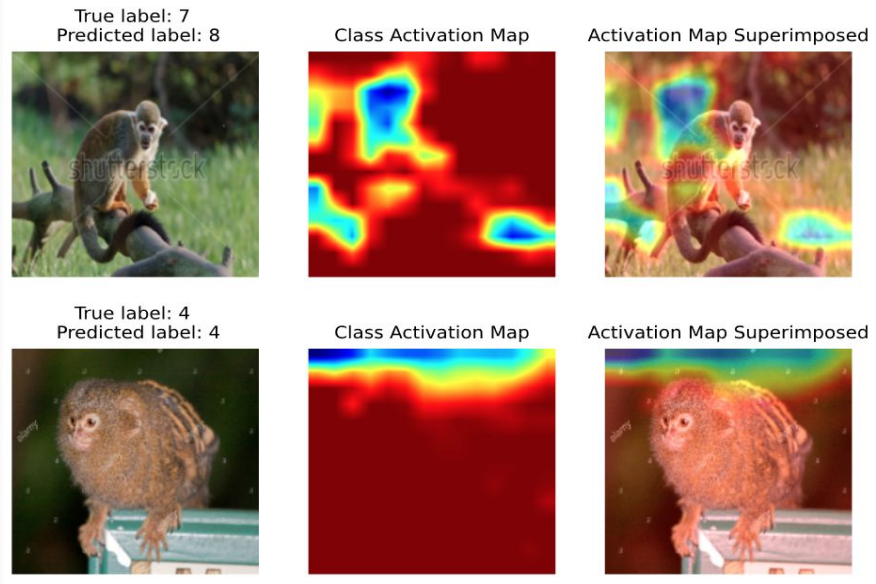
kernel_size = (8, 8)



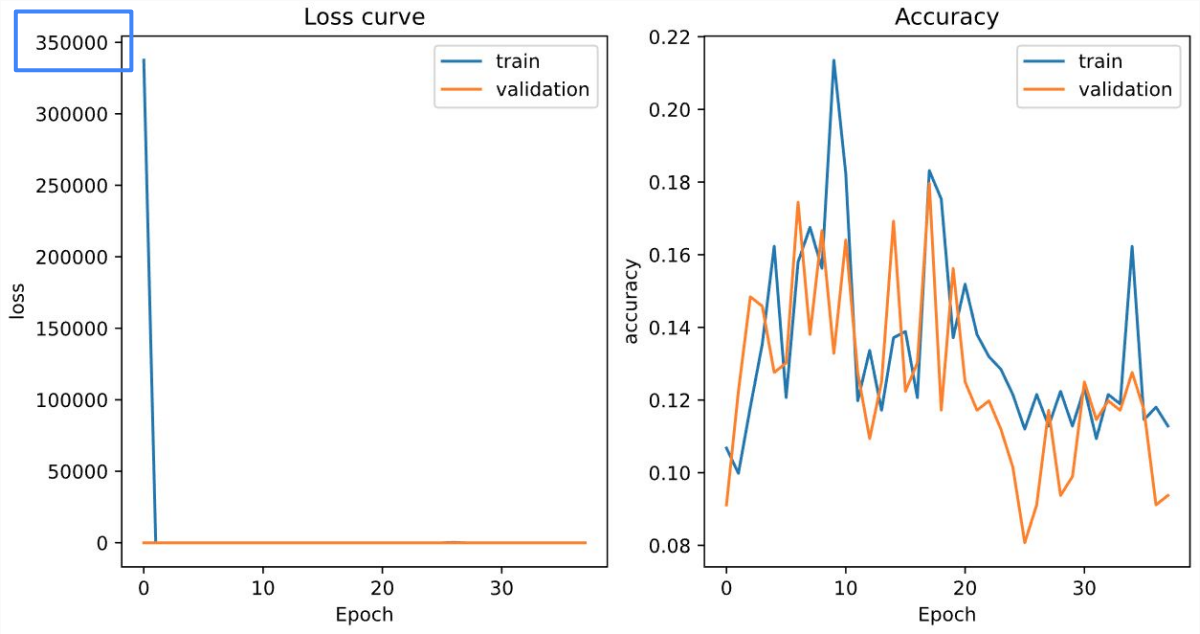
- *Max Epoch 33/100*
- *Wall time: 14min 44s*
- *Final validation accuracy: 29.69%*

kernel_size = (8, 8)



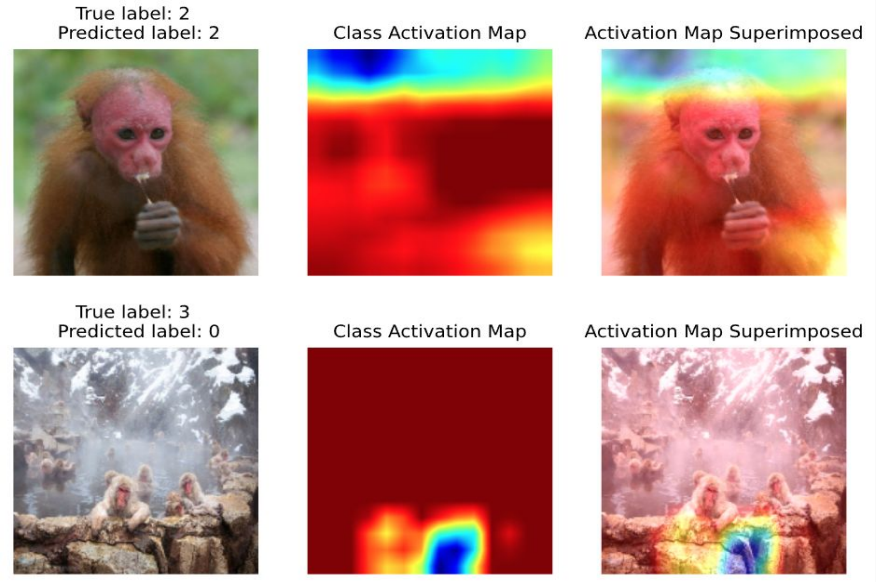


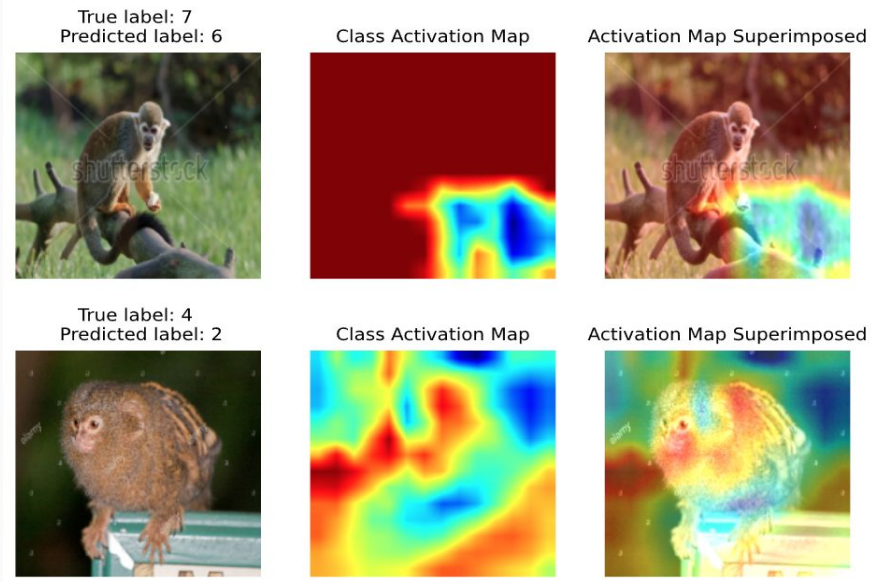
kernel_size = (16, 16)



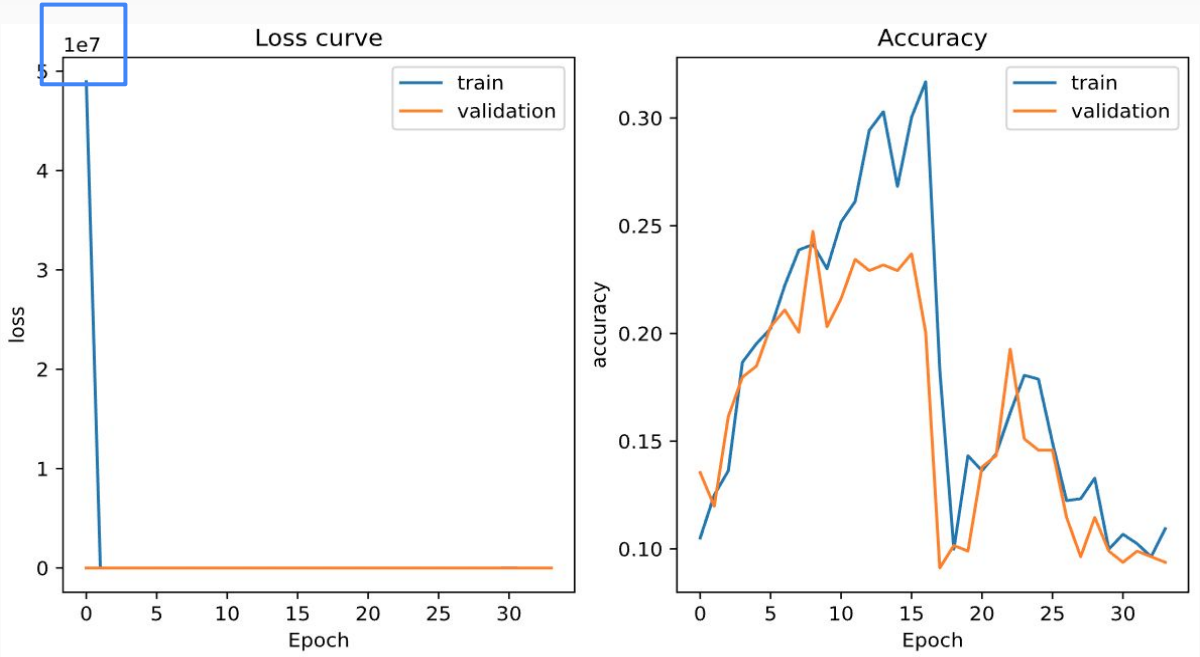
- *Max Epoch 38/100*
- *Wall time: 17min 5s*
- ***Final validation accuracy: 17.45%***

kernel_size = (16, 16)



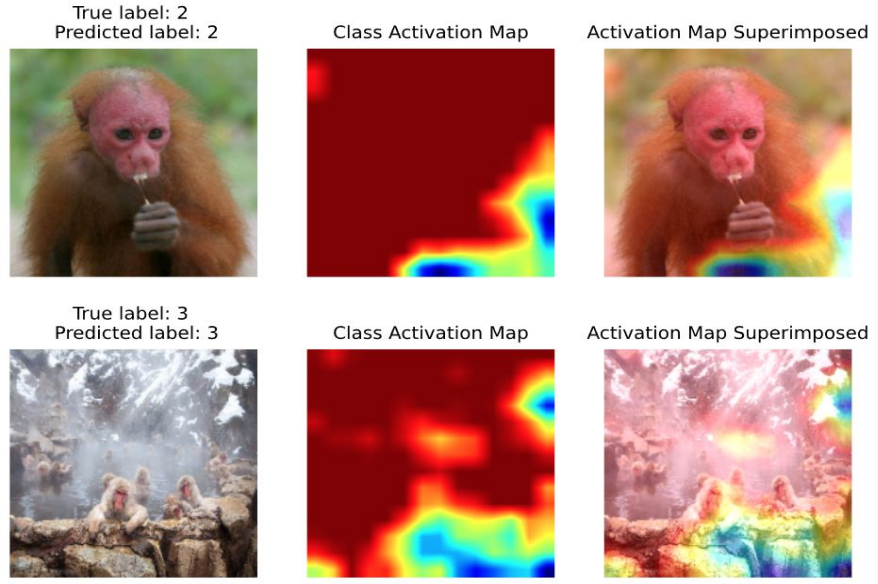


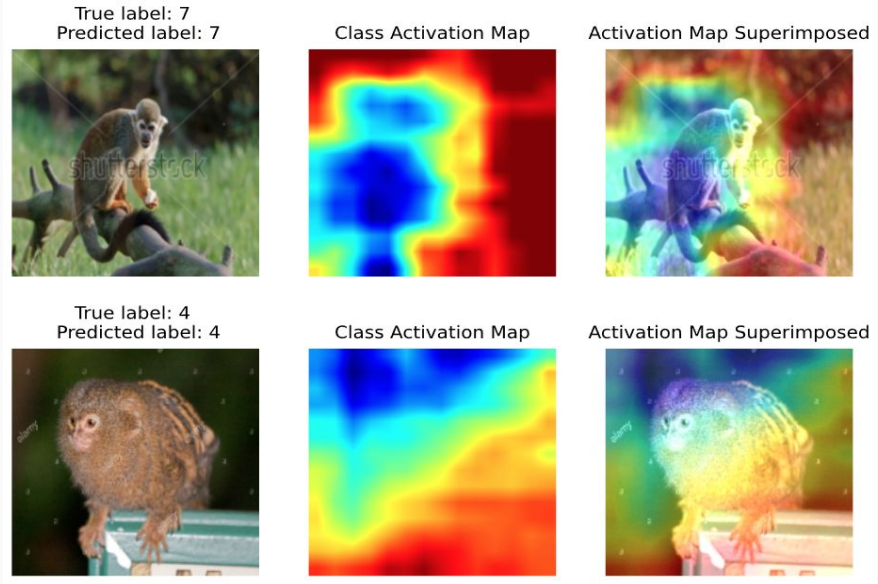
kernel_size = (32, 32)



- *Max Epoch 34/100*
- *Wall time: 16min 18s*
- *Final validation accuracy: 22.66%*

kernel_size = (32, 32)





Discussão

- Limitações de **memória** do computador usado nos treinamentos.
- Parece que com ***kernel size* menor o aprendizado ficou mais estável**.
- Necessidade de rodar **mais experimentos** para cada *kernel size*.
- Os modelos com resultados ruins de visualização que parecem se distanciar da percepção humana (ex.: *kernel sizes* 16 e 32), podem ter a sua aplicação, por exemplo: **talvez possam ajudar a identificar se o animal está ou não no seu *habitat* natural** no momento da fotografia.
- Interessante observar que a rede neural não necessariamente dá maior importância para *pixels* contíguos, ao contrário de técnicas “não-neurais”.

Trabalhos Futuros

- Rodar experimentos com **outros *datasets*** e com outras **resoluções de imagens**.
- Realizar experimentos com as **variações de CAM** propostas na Literatura.
- Desenvolver alguma **métrica que relacione as dimensões do *kernel size* com as dimensões da imagem** para o treinamento do “melhor modelo”.
    - Talvez haja uma relação entre o tamanho do objeto-alvo e o espaço livre restante na imagem. Explorar o uso de segmentação? Explorar o uso de *bounding boxes*?

Referências

- RAS, G., XIE, N., VAN GERVEN, M., DORAN, D., “**Explainable Deep Learning**: A Field Guide for the Uninitiated”, In: *Journal of Artificial Intelligence Research*, v. 73, pp. 329--397, 2022.
- ZEILER, M. D, KRISHNAN, D., TAYLOR, G. W., FERGUS, R., 2010, “Deconvolutional Networks”, In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. pp. 2528-2535, IEEE, 2010.
- ZEILER, M. D, FERGUS, R., 2014, “Visualizing and Understanding Convolutional Networks”, In: *European Conference on Computer Vision*, pp. 818-833, Springer, 2014.
- **ZHOU, B., KHOSLA, A., LAPEDRIZA, A., OLIVA, A., TORRALBA, A., 2016, “Learning Deep Features for Discriminative Localization”, In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 2921-2929, 2016.**

Obrigado

Explicabilidade em Modelos de Redes Neurais Convolucionais

Disciplina
INF 2064 - Visão Computacional - **Professor: Marcelo Gattass**
PUC-Rio - 2022.2

Aluno Extraordinário
Daniel da Silva Costa
danieldasilvacosta@gmail.com