

Monitoramento de  
espécies exóticas  
invasoras utilizando  
Redes Neurais  
Convolucionais

# 1. Introdução

- **Espécie exótica** é toda e qualquer espécie (vegetal ou animal) que se encontra fora de sua área de distribuição natural;
- **Espécie exótica invasora** representa uma ameaça ao ecossistema onde esta foi inserida e também é responsável por causar danos ambientais e prejuízos econômicos;
- O javali (*Sus scrofa*) é considerada uma espécie exótica invasora;
- De acordo com o IBAMA, abater um animal da fauna brasileira é crime ambiental;
- Por outro lado, a caça ao javali é autorizada como uma forma de controle, sendo vedada a comercialização de sua carne ou criação em cativeiro;



Respectivamente: catete, javali e queixada. A priori, são animais muito parecidos, tanto no aspecto visual quanto no comportamento.

## 2. Fundamentação teórica

- De acordo com Russel & Norvig, existem 3 tipos de aprendizado: **aprendizado não supervisionado**, **aprendizado supervisionado** e **aprendizado por reforço**;
- Uma **Rede Neural Artificial** é um modelo computacional abstrato do cérebro humano e, de forma similar, são compostas por uma unidade básica chamada neurônio;
- O neurônio artificial é composto por 3 partes principais: sinais de entrada, função de ativação e sinais de saída;
- Uma **Rede Neural de Múltiplas Camadas** ou **Rede Neural MLP** (*Multi Layer Perceptron*) possui pelo menos 3 camadas;

- Um neurônio sozinho é um classificador binário. É necessário combinar a saída de vários para problemas com mais de duas classes;
- Uma única camada resolve problemas linearmente separáveis;
- As **Rede Neurais Convolucionais** tiram um bom proveito da dimensão espacial das imagens que possuem 3 dimensões: largura, altura e profundidade (canais);
- São amplamente utilizadas na área de visão computacional;
- Se assemelham a Redes Neurais MLP comuns, mas com uma diferença: nem todos os neurônios da camada anterior se conectam aos neurônios da próxima camada, formando uma conectividade local;

- Suas camadas foram grupos: camadas de *pooling* e convolução;
- Esse tipo de Rede Neural necessita de um conjunto de treinamento relativamente grande para que não ocorra o problema de *overfitting*;

# 3. Desenvolvimento

- Foram utilizados quatro modelos de Redes Neurais para executar os experimentos: **AlexNet**, **VGG-16**, **Inception** e **ResNet-50**;
- Os testes foram realizados em uma máquina com i7-7700 com 32GB de RAM e placa de vídeo GTX 1050 Ti de 4GB GDDR5, usando Python 3.6, Keras 2.1.6-tf, Tensorflow 1.10.0 e CUDA 9.0;
- As imagens foram rotuladas e separadas manualmente em 3 classes: javali, cateto e queixada;
- Antes de submeter as imagens para a rede, foi aplicado uma técnica de aumento artificial de dados para aumentar o tamanho do conjunto de treinamento;

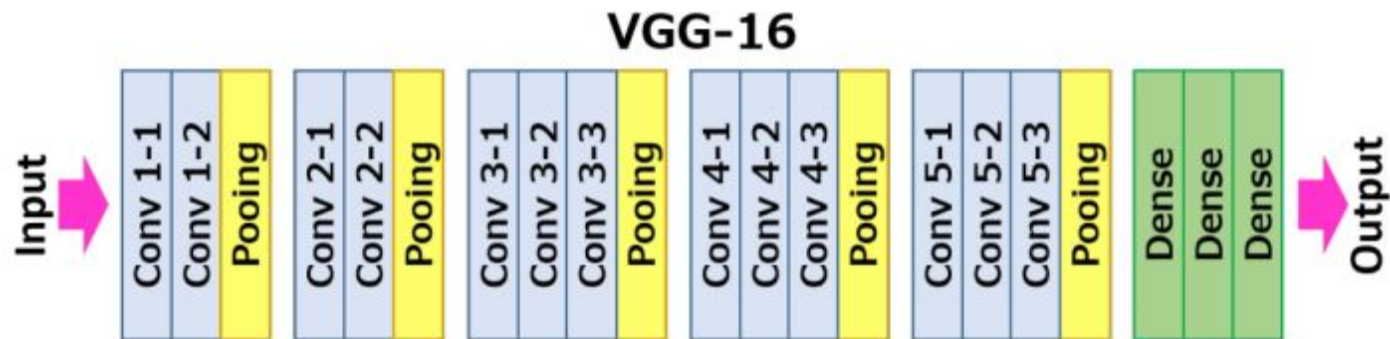


Figura 10 – Arquitetura da Rede Neural Conlucional VGG-16. Fonte: ([HASSAN, 2018](#))



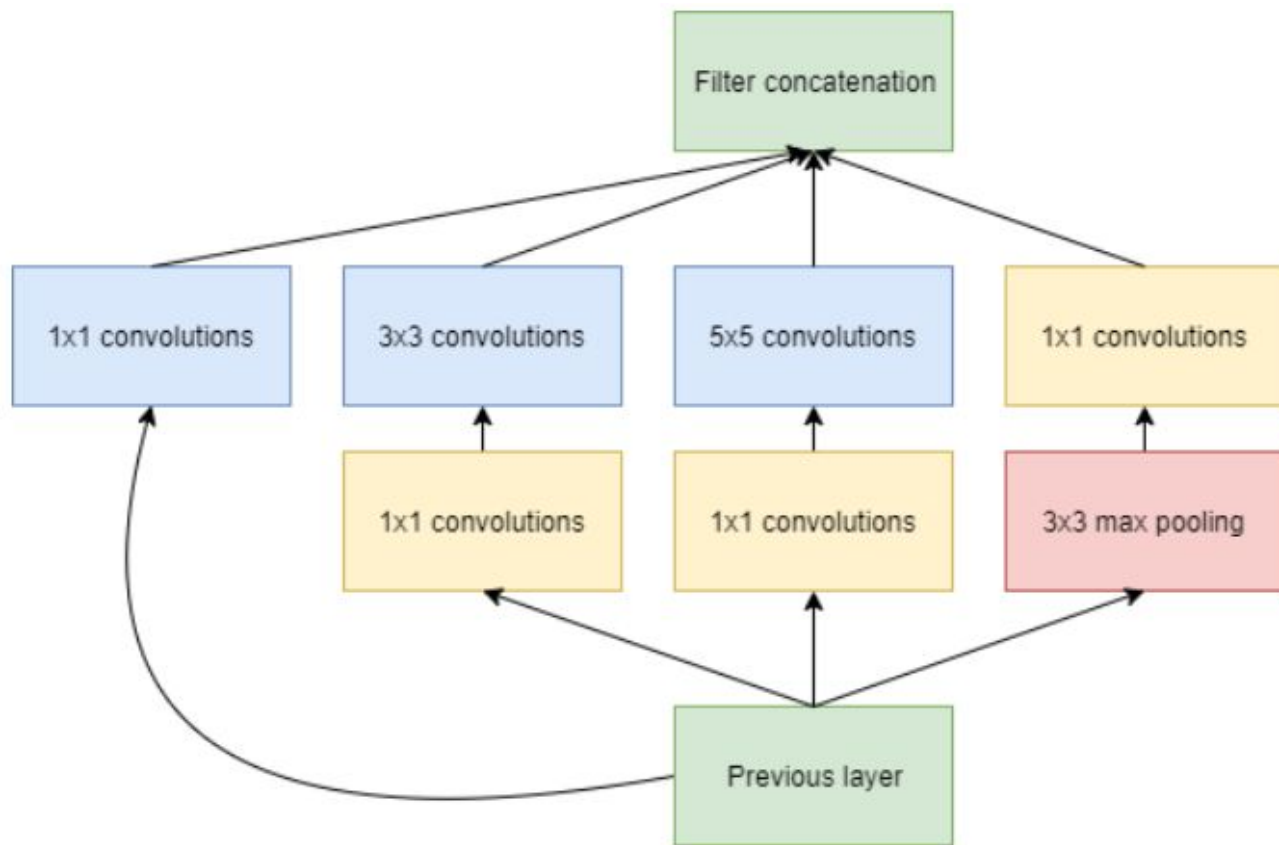


Figura 11 – Representação visual de um módulo Inception. Fonte: (SZEGEDY et al., 2015a)

### 34-layer residual

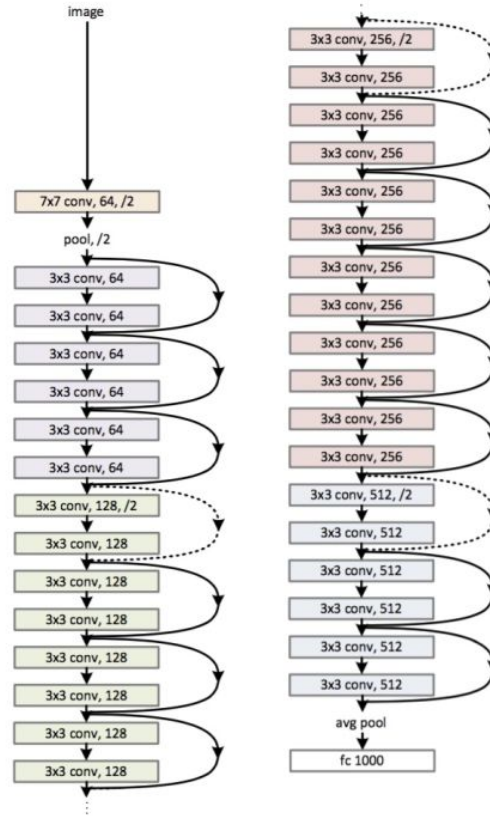


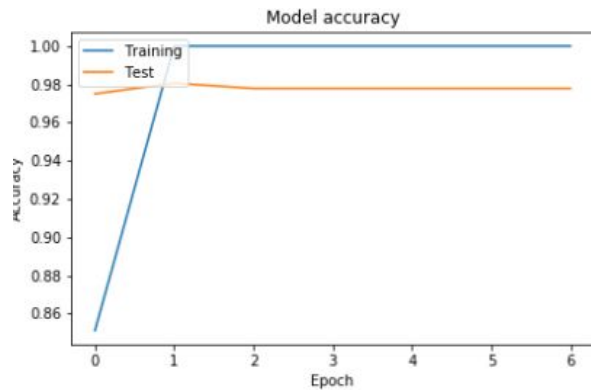
Figura 13 – Arquitetura da ResNet-34. Praticamente todas as arquiteturas Resnet seguem esse mesmo padrão: aplicação do ruído da camada anterior a cada duas convoluções.

- O conjunto foi separado em 60% para treinamento e 40% para testes;
- O conjunto possui 900 imagens no total;
- Após o aumento artificial de dados, o conjunto de treinamento passou a ter 9720 imagens (18 variações de cada);
- As imagens originais não foram utilizadas no conjunto de treinamento;

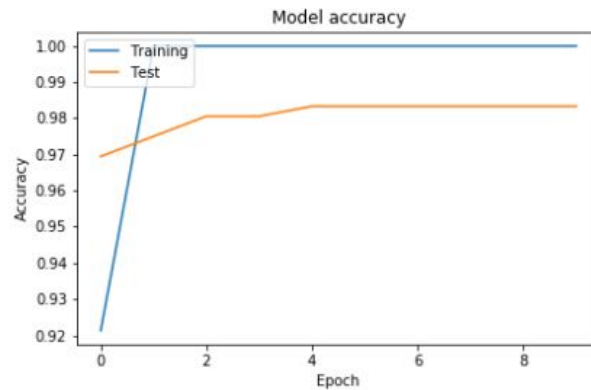
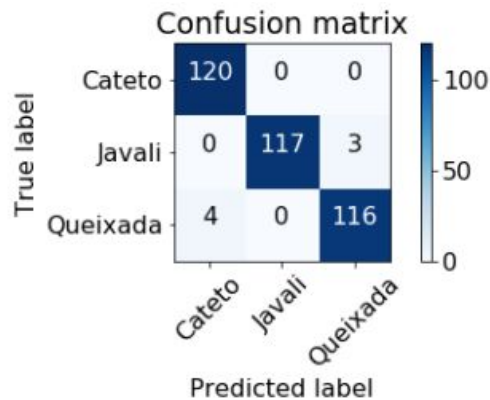
# 4. Resultados

- A AlexNet foi a única Rede Neural que foi treinada usando somente o conjunto de dados proposto e sem nenhum ponto de partida;
- Todas as demais foram utilizadas utilizando os pesos iniciais da ImageNet;
- Inicialmente a Inception-V3 foi incapaz de aprender sobre os dados, mas quando o modelo foi totalmente descongelado, a rede alcançou um bom resultado, mas com 1 porco do mato classificado como javali;
- O comportamento da ResNet-50 foi similar o da Inception-V3, mas esta alcançou uma acurácia de 99,17% no final;

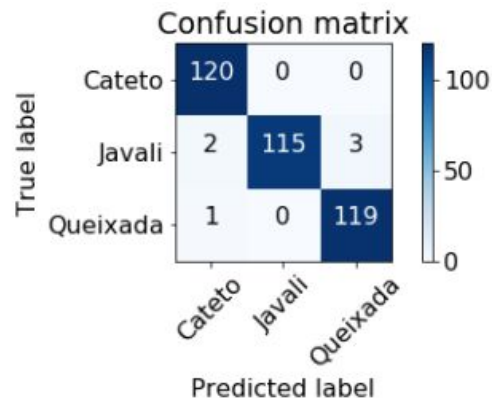
- A VGG-16 mostrou-se capaz de aprender algo desde o começo. A medida que mais camadas eram descongeladas, a acurácia da rede aumentava;
- A acurácia final do modelo foi de 98,33%;

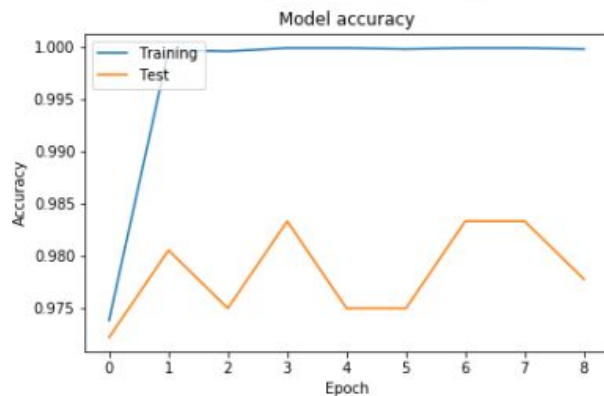
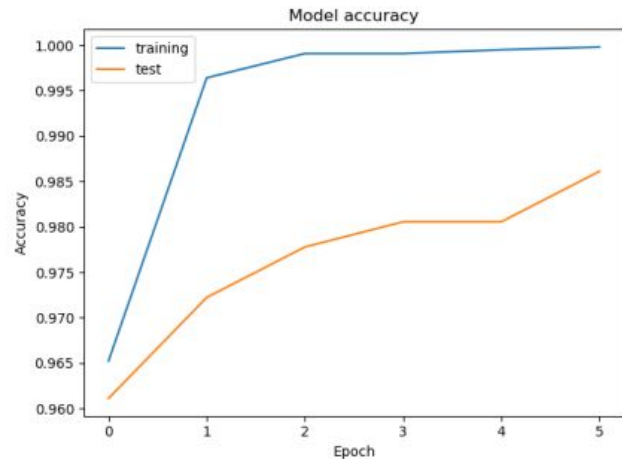


Teste 19 (VGG-16)

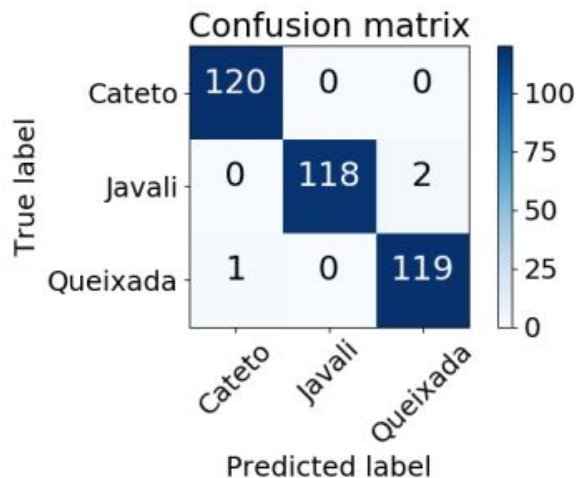


Teste 20 (VGG-16)

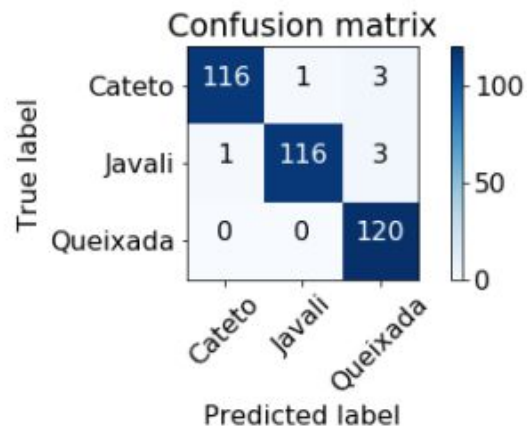




Teste 14 (ResNet-50)



Teste 8 (Inception-v3)



# 5. Conclusões

- Diferenciar classes diferentes que são muito próximas é parte do desafio da ImageNet;
- Levando em consideração os resultados obtidos após os treinamentos, a Inception-V3, ResNet-50 e VGG-16 se mostraram suficientes para resolver o problema abordado;
- Levando em conta a acurácia, a ResNet-50 foi escolhida. A VGG-16 é uma opção válida, mas com algumas ressalvas;