Computer vision fundamentals

Wild cats detection and classification

Leo Škvorc

Zagreb, June 9. 2024.

Content

[1 Introduction 4](#_Toc168862419)

[2 Data 4](#_Toc168862420)

[3 Training models 4](#_Toc168862421)

[3.1 My model 4](#_Toc168862422)

[3.2 EfficentNetB0 and ResNet50 4](#_Toc168862423)

[3.3 Training 4](#_Toc168862424)

[4 Results 4](#_Toc168862425)

[4.1 Accuracy loss f1 socres 5](#_Toc168862426)

[4.2 Confusion matrix 5](#_Toc168862427)

[5 Conclusion 5](#_Toc168862428)

Tables

**No table of figures entries found.**

Images

**No table of figures entries found.**

# Introduction

This project focuses on the domain of computer vision, specifically targeting the challenge of detecting and classifying cats in wild settings. The purpose of this project is to develop a robust model that can accurately identify and classify cats in various outdoor environments.

# Data

Training and validation data is sourced from [10 Big Cats of the Wild – Image Classification](https://www.kaggle.com/datasets/gpiosenka/cats-in-the-wild-image-classification/data). The data is brought from train, validation and test folder to one master folder and then split 80-20 to train and valid data. Additionally test data is created by [Canva](https://www.canva.com) using prompts “(Cat name) in the wilderness”. Each test class has 40 images of different styles.

To prepare the data for training image generators have been used to rescale all the images to 0 and 1. Additionally train images have randomly been rotated by 20 degrees, shifted by 20%, horizontally flipped.

Image generators have been created with respected datasets, targeted image size of 256x256, batch size of 16 and categorical class model.

Training data contains 1946 images, validation data contains 493 images and test data contains 400 images. The data is split into 10 classes: AFRICAN LEOPARD, CARACAL, CHEETAH, CLOUDED LEOPARD, JAGUAR, LIONS, OCELOT, PUMA, SNOW LEOPARD and TIGER.

# Training models

## My model

Model has been created using multiple convolutional and pooling layers. The first layer additionally has input shape of the image what is 256x246x3, meaning that the height and width of the image are 256, and it has RGB colours. All convolution layers have relu activation functions. Before the end data is flattened and densed. The final densing function has the output of the number of classes and activation function softmax.

## EfficentNetB0 and ResNet50

State of the art models are adjusted to fit desired classes using pooling and dense function.

## Training

All models are compiled using “Adam” optimiser, “categorical\_crossentropy” loss and accuracy metrics.

Models are trained over 40 epohs and created model is saved to the file system.

# Results

## Accuracy loss f1 socres

[Describe each API endpoint, including methods, parameters, and example requests and responses.]

## Confusion matrix

[Explain how to authenticate with the API, if necessary.]

# Conclusion