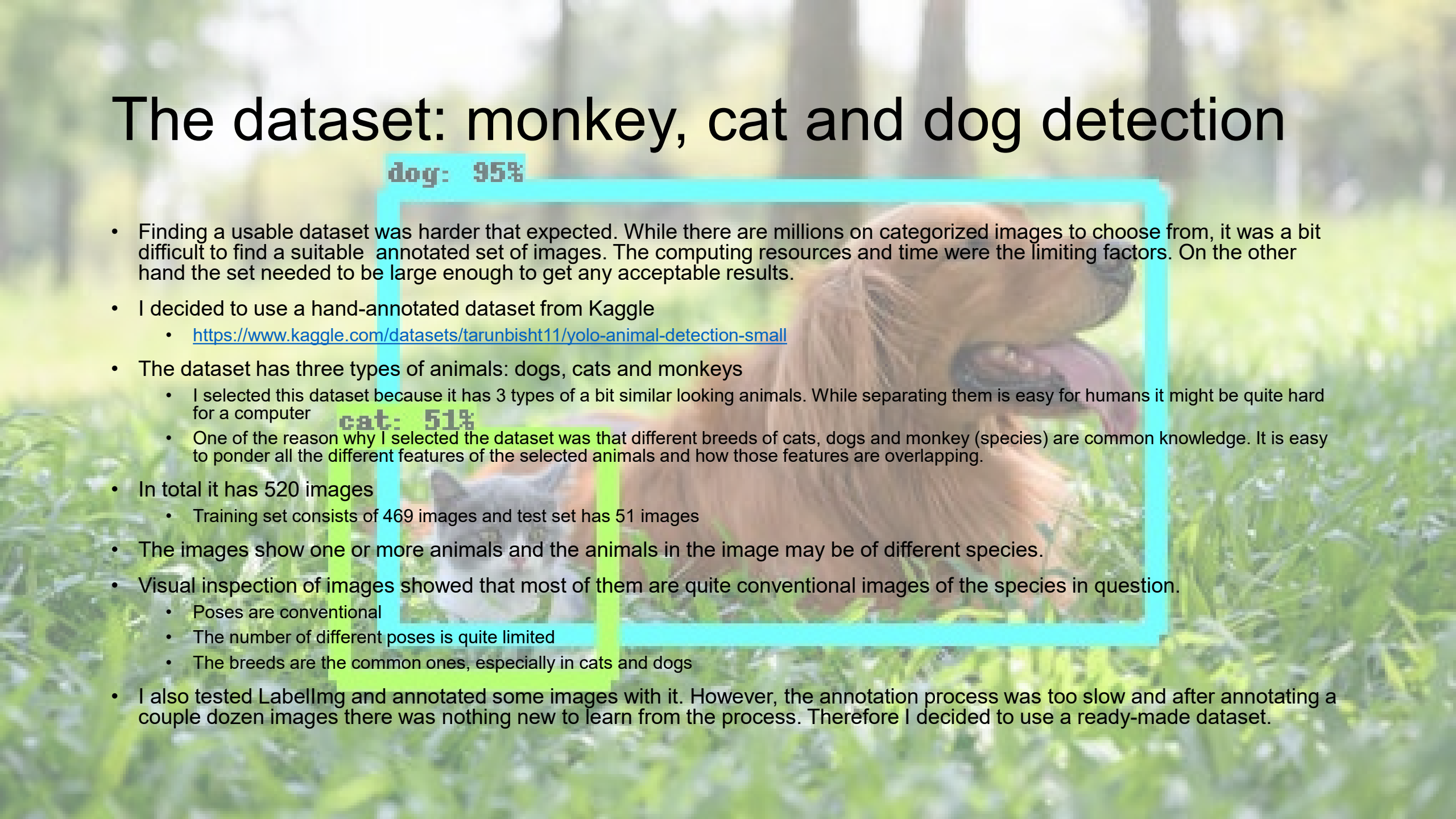
Computer vision and sensor fusion

Assignment 3: object detection

Jari Lehto

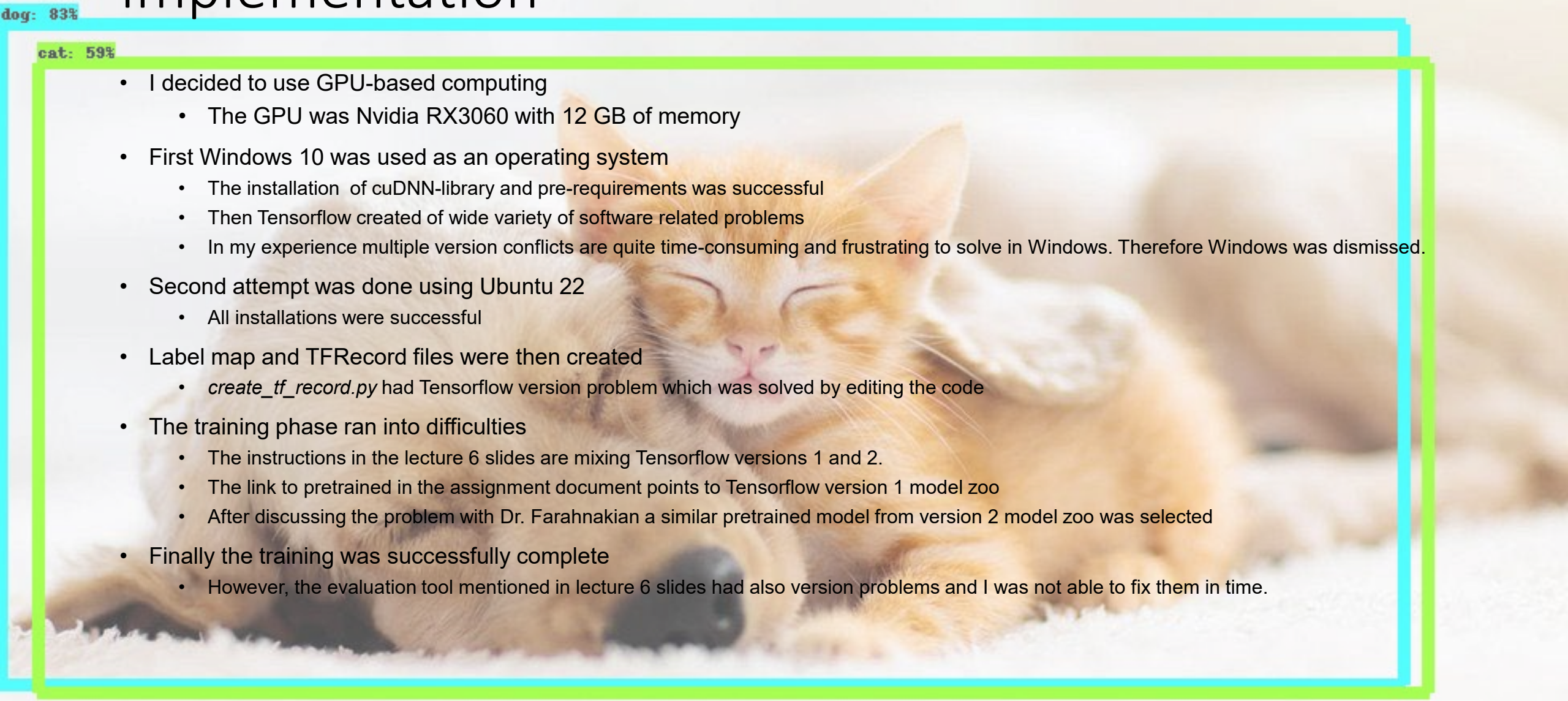
2.3.2023

The dataset: monkey, cat and dog detection



- Finding a usable dataset was harder that expected. While there are millions on categorized images to choose from, it was a bit difficult to find a suitable annotated set of images. The computing resources and time were the limiting factors. On the other hand the set needed to be large enough to get any acceptable results.
- I decided to use a hand-annotated dataset from Kaggle
  - https://www.kaggle.com/datasets/tarunbisht11/yolo-animal-detection-small
- The dataset has three types of animals: dogs, cats and monkeys
  - I selected this dataset because it has 3 types of a bit similar looking animals. While separating them is easy for humans it might be quite hard for a computer
  - One of the reason why I selected the dataset was that different breeds of cats, dogs and monkey (species) are common knowledge. It is easy to ponder all the different features of the selected animals and how those features are overlapping.
- In total it has 520 images
  - Training set consists of 469 images and test set has 51 images
- The images show one or more animals and the animals in the image may be of different species.
- Visual inspection of images showed that most of them are quite conventional images of the species in question.
  - Poses are conventional
  - The number of different poses is quite limited
  - The breeds are the common ones, especially in cats and dogs
- I also tested LabelImg and annotated some images with it. However, the annotation process was too slow and after annotating a couple dozen images there was nothing new to learn from the process. Therefore I decided to use a ready-made dataset.

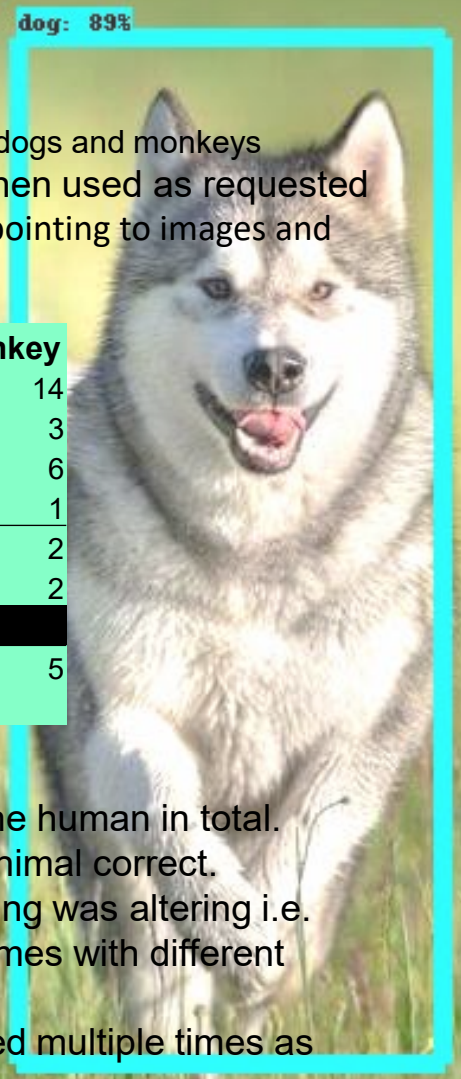
Implementation



- I decided to use GPU-based computing
  - The GPU was Nvidia RX3060 with 12 GB of memory
- First Windows 10 was used as an operating system
  - The installation of cuDNN-library and pre-requirements was successful
  - Then Tensorflow created of wide variety of software related problems
  - In my experience multiple version conflicts are quite time-consuming and frustrating to solve in Windows. Therefore Windows was dismissed.
- Second attempt was done using Ubuntu 22
  - All installations were successful
- Label map and TFRecord files were then created
  - *create_tf_record.py* had Tensorflow version problem which was solved by editing the code
- The training phase ran into difficulties
  - The instructions in the lecture 6 slides are mixing Tensorflow versions 1 and 2.
  - The link to pretrained in the assignment document points to Tensorflow version 1 model zoo
  - After discussing the problem with Dr. Farahnakian a similar pretrained model from version 2 model zoo was selected
- Finally the training was successfully complete
  - However, the evaluation tool mentioned in lecture 6 slides had also version problems and I was not able to fix them in time.

Evaluation

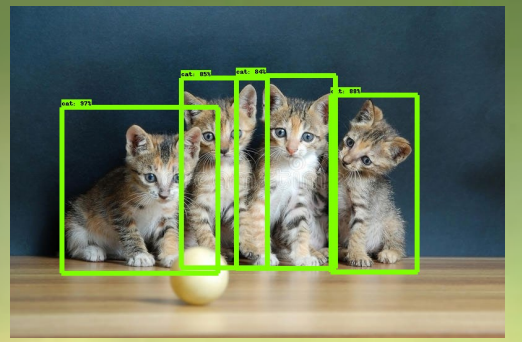
- 45 new images were downloaded from internet
  - The images have one or many animals of cats, dogs and monkeys
- object_detection_tutorial.ipynb notebook was then used as requested
  - Most of the code is unmodified. Only paths pointing to images and model needed to be changed.

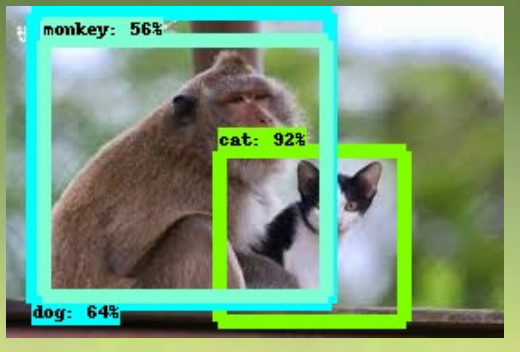


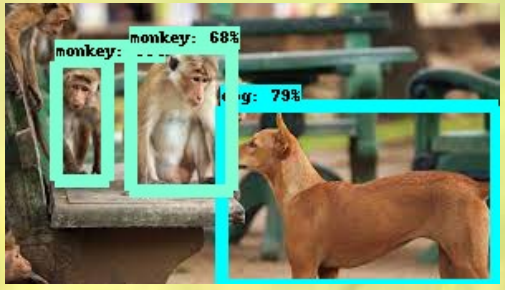
Inherence results	Dog	Cat	Monkey
Correctly detected	15	17	14
Correctly detected multiple times	3	2	3
Correct and incorrect multiple times		4	6
Incorrectly detected multiple times			1
Detected incorrecly as a dog		1	2
Detected incorrecly as a cat	3		2
Detected incorrecly as a monkey			
Not detected	1	1	5
Detected human as a dog	1		

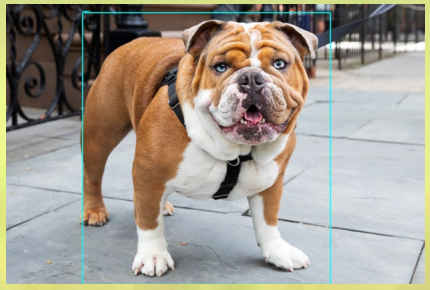
Table 1: number of individual animals detected

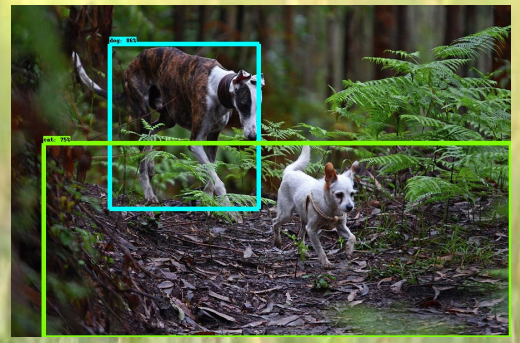
- The evaluation images have 80 animals and one human in total.
- Most times the model was able to detect the animal correct.
  - Sometimes the species was correct framing was altering i.e. the same animal was detected multiple times with different framing
  - A few times the same animal was detected multiple times as a different species
- There were no images of humans in the training set and there is no category “person” either so the model is not able to detect a human.

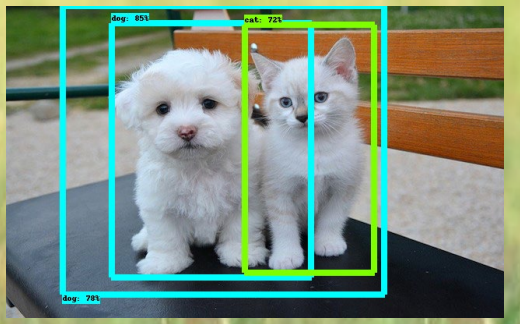


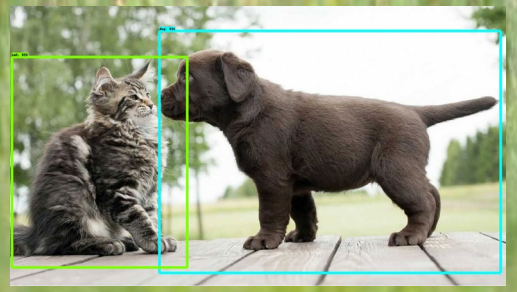








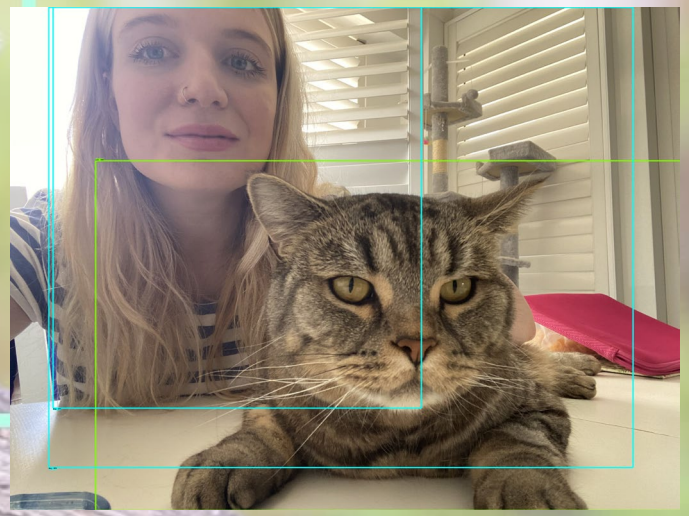




Conclusions

- Object detection is a very powerful tool but needs to be used with care.
  - The model is as good as its training data and selected categories
  - It is also important to remember the limitations. What is the model trained for?
  - When designing a model and selecting its training data one must consider the purpose of the model. What kind of images will the model see? Obviously there will be objects we want to detect but is there the also similar ones?
  - For example if we want the model to detect dogs will there be handlers in the same images? If so maybe it is also necessary to add people to training data (and a category person) so model can separate dogs from people.
- The amount and quality of training data are crucial
  - The more variation there is within the category, the more data is needed
    - For example different breeds
  - The variation must be covered as well as possible.
  - Similarities between categories have to be considered. Are there enough separating features?
- The model performed surprisingly well on the evaluation images
  - In almost all images which were similar to images in the training set the animals were correctly detected
  - As expected the performance on images that were dissimilar to training images was modest
  - While all cats and dogs are same species, the monkeys have multiple species and therefore even more variation
  - By looking at the result images it is kind of easy to see what the model detected and understand the errors.

Even a human might not recognize this one correctly. Yes, it is a monkey, but it does look a bit like a cat or a dog.



There were no images of humans in the training set and the model does not have a category “person”. Does she look like a dog like the model says? Well, she has eyes, nose, mouth and long hair which can be interpreted as lop-ears.