AI-Nimals for Nvidia Jetson TX challenge. Information for Jury.

Content of package:

Catalogues:

- **config** – all configuration files and options used i training process, preprocessing and tracking scripts
- **pyimagesearch** – additional library I was using as helper in training and tracking process
- **videos** – samle videos I have tested the algorithm on because it's hard to catch a bird in the middle of a night when I was developing that project

Files:

- **ai_nimals_finetune_vgg16.py** – training script for fine tuning VGG16 model. Fine tuning was based on imagenet loaded weights to keras VGG model.
- **ai_nimals_finetuned_vgg16.model** – trained model that can obtain 93% accuracy in birds calssification
- **ai_nimals_scrapper_google.py** – scrapper that downloads birds images from Google for training purpose
- **ai_nimals_scrapper_prepare.py** – script that prepares scrapped files for autimatic birds detection and preprocessing before manual labeling is done
- **ai_nimals_tracker.py** – main file that is running on Jetson TX2 platform. This application does bird localization, classification and statistic update.
- **MobileNetSSD_deploy.caffemodel** – NN used for bird localization
- **MobileNetSSD_deploy.prototxt.txt** – NN used for bird localization
- statistic.csv – statistic csv file
- **ai_nimals_finetuned_vgg16.json** – training process printout
- **ai_nimals_finetuned_vgg16.png** - training process chart for controlling overfitting and training.

Details:

Like I said at the beginning of YT video: **https://youtu.be/XZM0I2xBqFk**, I normally work on full time job. And after work I do a lot of sports to stay fit and have power to stay 100% focused on all things that I am doing.

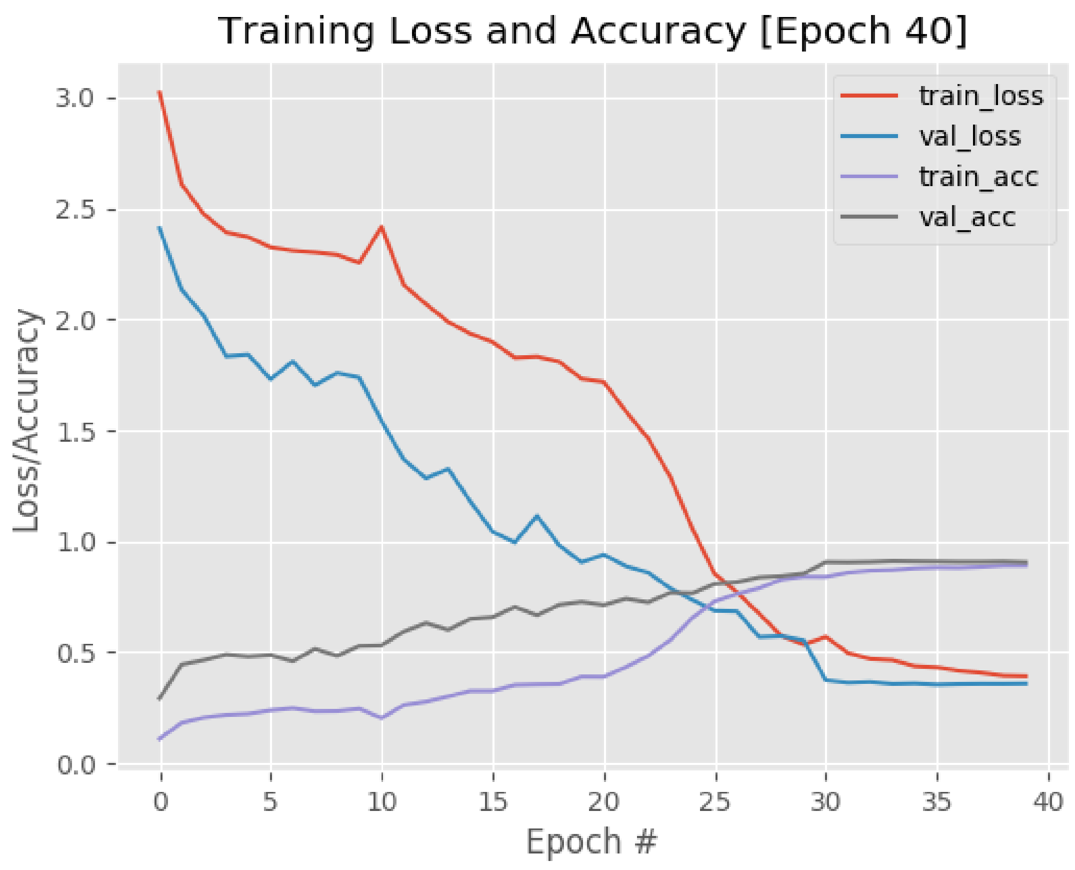
This project was fully made by myself after working hours - at night when my children were slipping. Prototype was developed on my desktop and than ported to Jetson TX2 platform. Jetson TX2 performance on testing video files can be seen on youtube movie at that moment: **https://youtu.be/XZM0I2xBqFk?t=180.**

The hardest part was to train proper classifier that will give me quite nice accuracy. First of all I wrote script that can download bird specious from google images **(ai_nimals_scrapper_google.py).** Than I run another script that cropped bird from all downloaded images and saved it in proper destinantion (**ai_nimals_scrapper_prepare.py).**

When downloading was done, me and my friend (ornithologist) manually checked all 40000 files fo be sure that training process will get correct classes.

Than I checked few DNN architectures like GoogLeNet, ResNet56, AlexNet, DeeperGoogLeNet and VGG16. All seemed to work fine but VGG16 gave the highest accuracy at the beginning so I started experimenting with fine tuning on imagenet weights.

In the end I trained VGG16 using fine tuning and control-break method for learning rate corrections (**ai_nimals_finetune_vgg16.py**). I keep over fitting quite nice what can be seen on this image:



All training was done on my GTX1060! One epoch took about 450 seconds. And to get this score I was making experiments for two weeks. This card is damned slow… But any way I get 93% in accuracy!

Mechanics.
All electronics was closed in Case with power battery supply to have mobility to test it in terrain. Today this device can run 10 hours on battery. It’s 100% mobile and it can also be closed in IP67 case and it will be waterproof.

Performance:
MobileNetSSD can run at 2-3 FPS. Due to having very large classification model VGG16 can classify 0.5 FPS, but it gets very high accuracy!

Scalability:
This project can be used to track any king of animal in the field. It’s just a matter of training data for 2 deep neural networks.

What next?
I want to continue developing this project and check other king of accuracy boosting (ensembles or gradient boosting). I want to add multiple birds classification and recognition because currently I can track and find only one that has the highest probability found by MobileNetSSD.