Project specification: AM-Traffic I. Phase 2. Iteration 2

Purpose of the document

The purpose of this document is to give explicit description of tasks that has to be accomplished during Phase 2-Iteration 1 of AM-Traffic 1 project.

Tasks description

Task #1

Brief description:

Classification of weather conditions based on following categories:

a. Clear (0)
b. Raining, three scales:
    - Weak rain (1)
    - Mediocre rain (2)
    - Heavy rain (3)
c. Snowing, three scales:
    - Weak snow/sleet (4)
    - Mediocre snow/sleet (5)
    - Heavy snow/sleet (6)
d. Mist/Fog

Detailed description:

Prepare and develop end-to-end pipeline (from dataset aggregation to network deployment on end device) for a weather conditions classification light-weight neural network. As an input this model should take a video sequence from CCTV camera; As an output model should classify weather conditions (Clear, Rain, Snow, Mist/Fog). Network should be light enough to run in realtime on a Jetson Nano device.

It's clear that it's hard to find such CCTV dataset on the web, so extensive 'weather conditions augmentation' module can be considered as a solution.

Task #2

Brief description:

- Classification of road condition regarding weather based on following categories:
    a. Dry (1)
    b. Moist (2)
    c. Wet (3)
    d. Wet & Salty (4)
    e. Frost (5)
    f. Snow (6)
    g. Ice (7)
    h. Probably moist/salty (8)
    i. Slushy (9)

Detailed Description:

Prepare and develop end-to-end pipeline (from dataset aggregation to network deployment on end device) for a road condition classification light-weight neural network. As an input this model should take a video sequence from CCTV camera; As an output model should classify road condition (Dry, Moist, Wet, Wet & Salty, Frost, Snow, Ice, Probably moist/salty, Slushy). Network should be light enough to run in realtime on a Jetson Nano device.

Task #3

Brief description

Measurement of road lighting conditions on a numeric scale.

Detailed Description:

Prepare and develop end-to-end pipeline (from dataset aggregation to network deployment on end device) for a road lighting classification light-weight neural network. As an input this model should take a video sequence from CCTV camera; As an output model should classify lighting condition. Network should be light enough to run in realtime on a Jetson Nano device.

Task #4

Brief description

Parse data from Finland road traffic data website.

Detailed Description:

Prepare and develop a set of python scripts to aggregate data from the Road Traffic Finland website.

Scripts should carry out following functionality:

1. Save data from sensors to a SQL-database with a timestamp and GPS location.
2. Put pins on a map with a sensor type (camera, weather station, etc.).
3. Find cameras located nearby to other types of sensors (radius around 100-200 meters).
4. Create a dataset from camera image and sensor data pairs for Tasks 1, 2 and 3.
5. Build an AWS-based solution to run scripts on a schedule and record data in AWS-database.

Additional requirements:

1. Scripts should be well documented.

2. There's got to be an Anti DDOS mechanism, so the website wouldn't blacklist caller IP-address.
3. Aggregated data must be used in Tasks #1,2,3, so there's gotta to be a conversion script from database to Tasks #1,2,3 format.

Project quality requirements

For the completion of the project, the employee must deliver the following to Ilya Kutsenko (ilya.kutsenko@recon-ai.fi)

1. Project schedule and cost
2. Documentation of the solution including:
    a. Links to datasets utilized in project (if datasets were annotated manually, links to source data, annotation files, annotation tools with usage manual)
    b. The architecture, its input and output spaces,
    c. Training and running codes,
    d. Potential pre- or post-processing.
3. Source code, including
    a. Dataset parsing scripts
    b. Dataset augmentation scripts (if used separately from training script)
    c. Network architecture
    d. Training and validation script
    e. Post processing script (if needed)
    f. Network deployment and/or conversion scripts (for example Tensorflow to TensorRT, Pytorch to ONNX to TensorRT)
    g. Testing (Demonstration) script (for example video processing demo with labelling or json output)
4. Github repository(-ies) (private) with source code
5. Valohai-execution with end-to-end pipeline (except Jetson Nano deployment step) and test scripts
6. A video of the solution, i.e., a video processed by your network of the pre-augmented dataset including an illustration of the objects of interest.
7. The architecture and weights of the neural network used

The project will be assessed based on the following criteria:

1. Quantitative quality of the network: The network will be run on data similar to the given training/testing data. Based on this, the quality of the network is assessed with quality metrics described below.
2. The network must be lightweight enough to be run on Nvidia Jetson Nano.
3. Object detection algorithms should have mAP@[0.5:0.95] > 70.0 (average 'mean Average Precision' over different IoU thresholds, from 0.5 to 0.95; see 'COCO mAP') and batch size = 1 processing time < 70 ms.
4. Schedule, specifically:
    - Time required for the completion of the project

- Delivery within the set schedule

Project schedule

Phase 2 Iteration 2 must be completed before 31.3.2020