Deepness: Deep Neural Remote Sensing QGIS Plugin

Marek Kraft

Poznań University of Technology

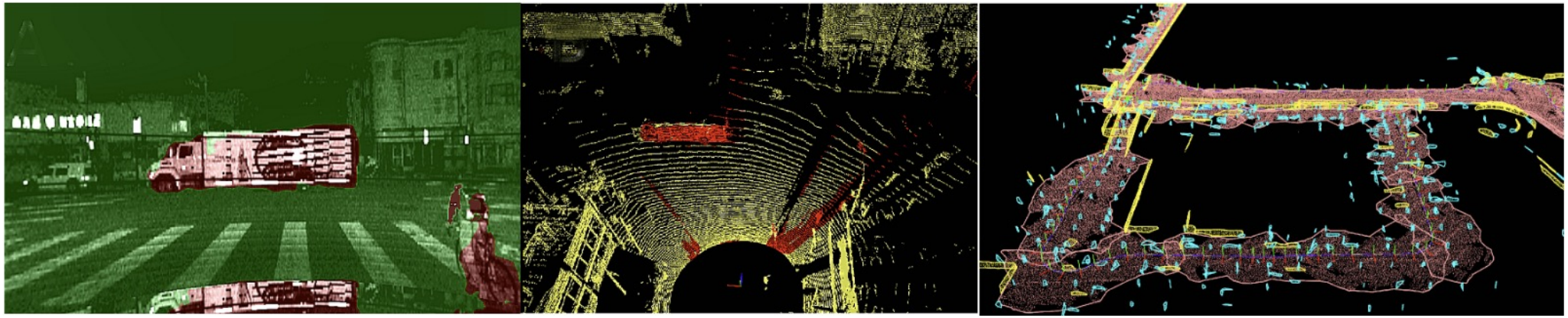
Space Data Management Workshop

Who we are

- I’m with the Poznań University of Technology, Institute of Robotics and Machine Intelligence, Computer Vision Lab
- Our research focus are inteligent, autonomous machines – perception, planning and action – with extensive use of AI

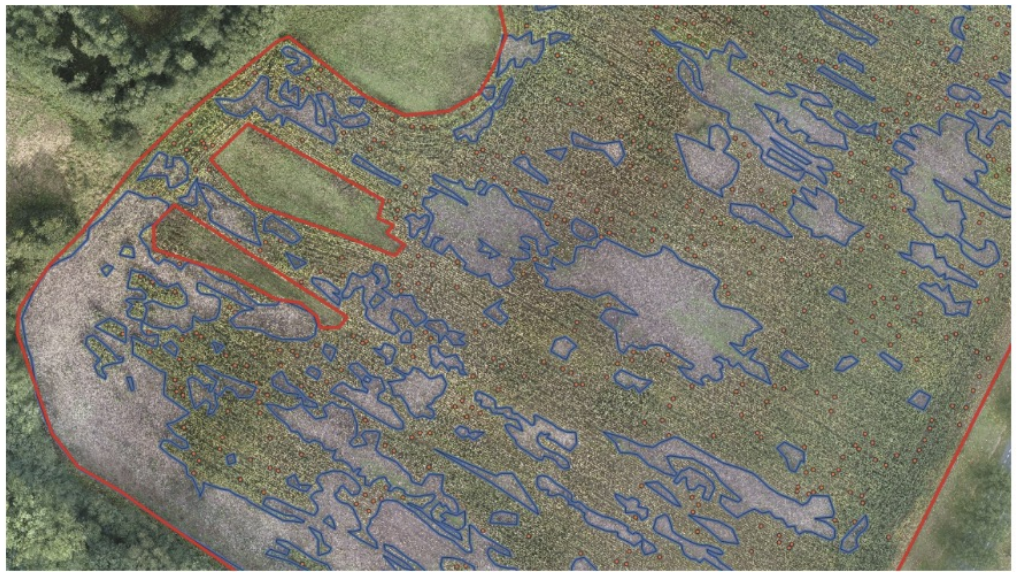
Who we are

- I’m with the Poznań University of Technology, Institute of Robotics and Machine Intelligence
- Our research focus are inteligent, autonomous machines – perception, planning and action with extensive use of AI



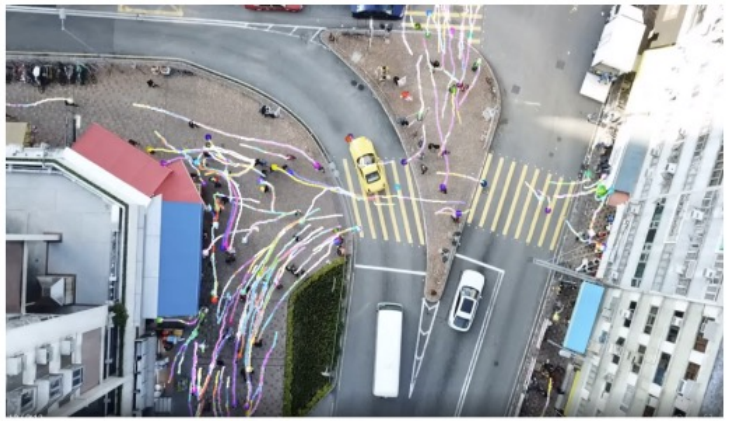
Who we are

- I’m with the Poznań University of Technology, Institute of Robotics and Machine Intelligence
- Our research focus are inteligent, autonomous machines – perception, planning and action with extensive use of AI



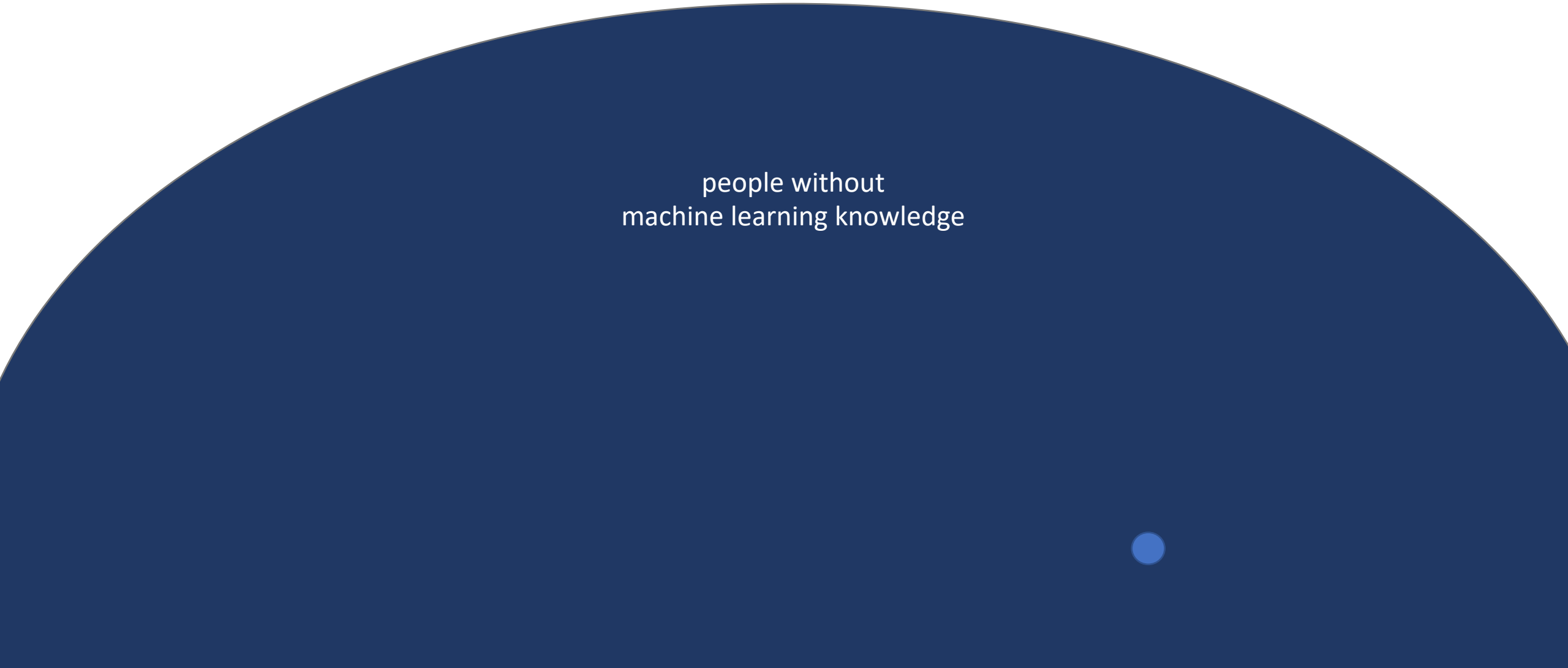
Who we are

- I’m with the Poznań University of Technology, Institute of Robotics and Machine Intelligence
- Our research focus are inteligent, autonomous machines – perception, planning and action with extensive use of AI

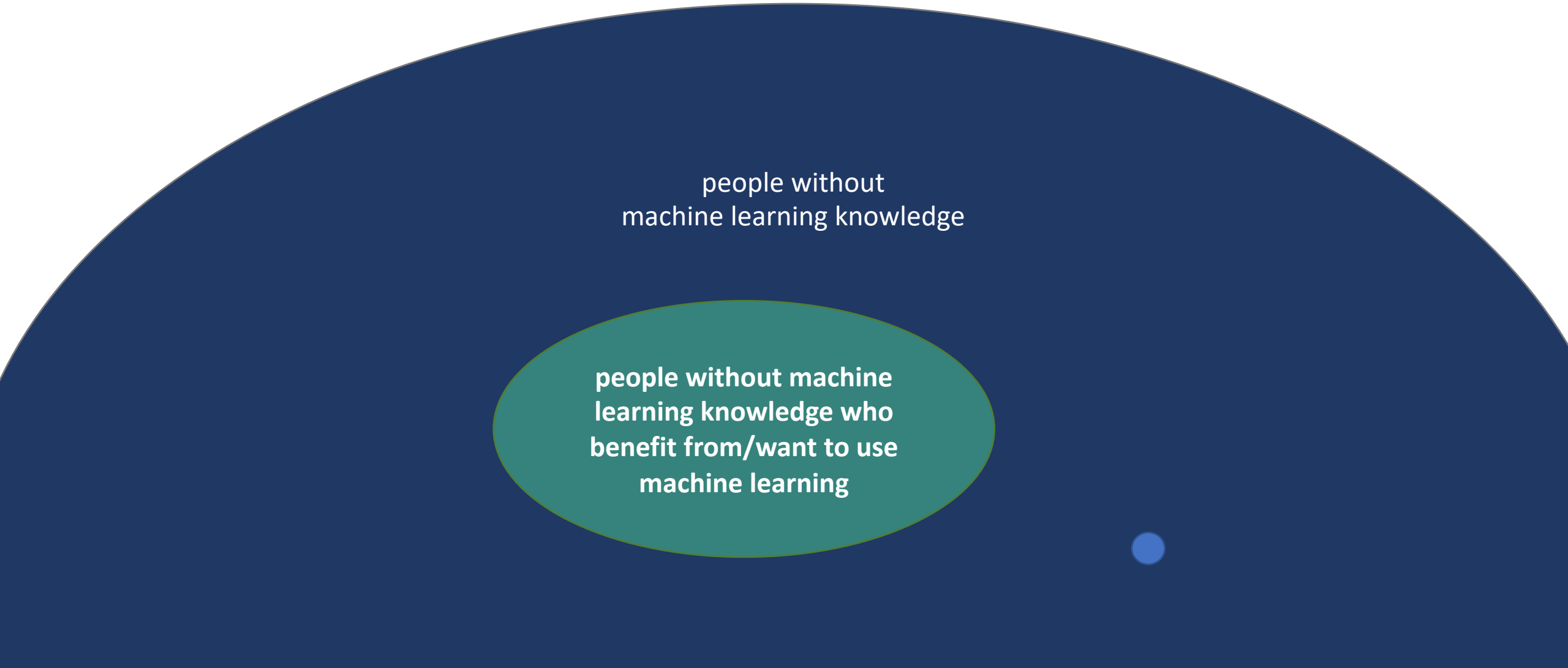



Motivation – why?

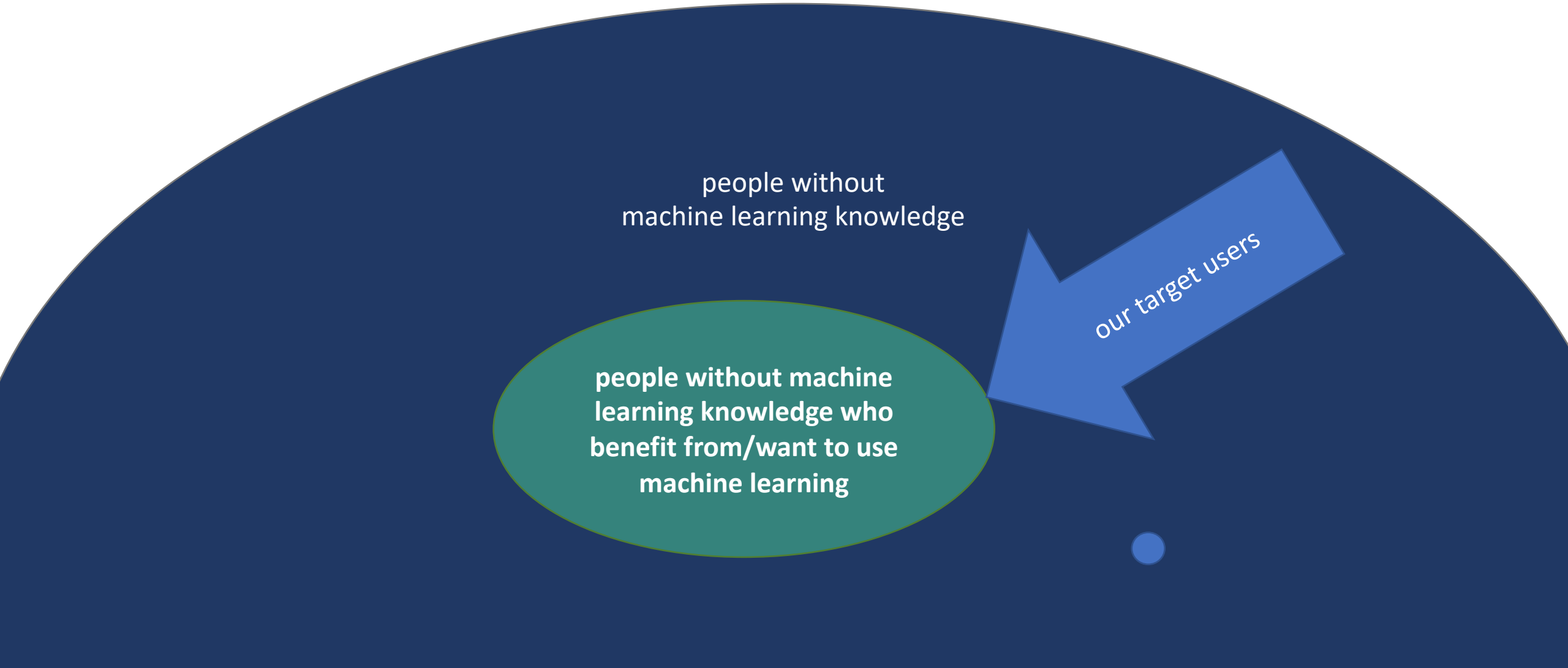
Motivation – why?



Motivation – why?



Motivation – why?



Motivation – why?

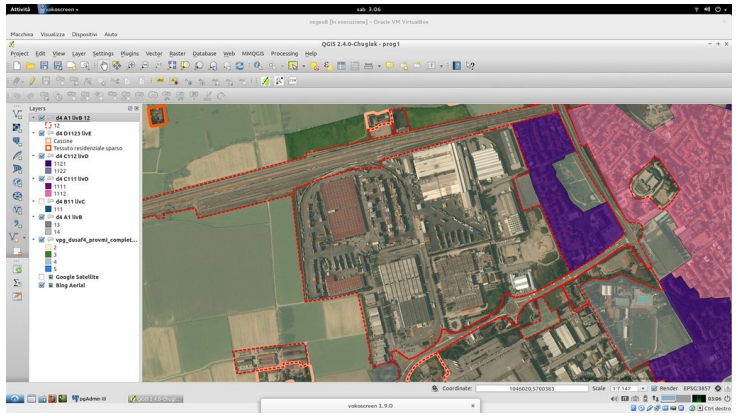
We want to democratize access to the potential created by deep learning applications by facilitating their use with remote sensing data without expert knowledge

How?

- We want appeal to a large base of users
  - ❑Prioritize familiar user experience
  - ❑Free, open-source software
  - ❑Easily extendable with new models and functionalities
  - ❑Provide quality documentation, tutorials etc.

How?

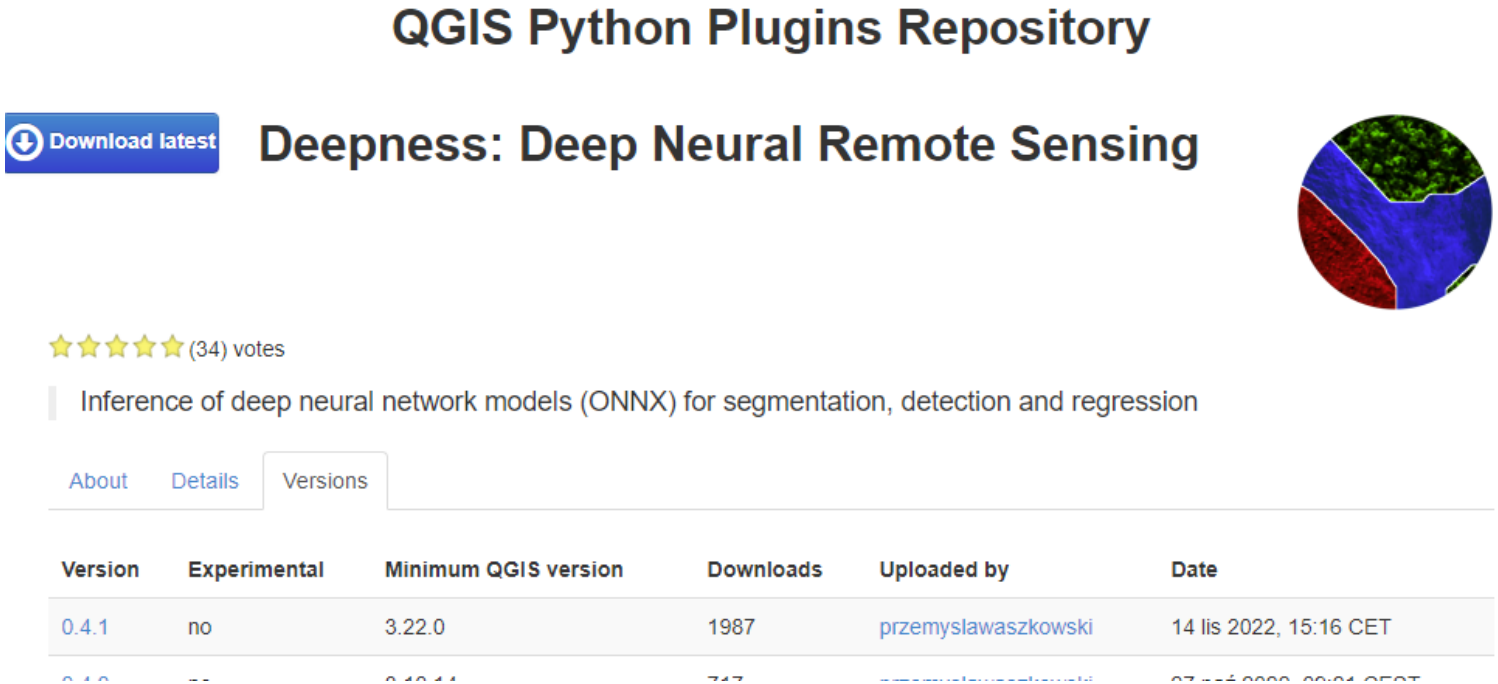
- We want appeal to a large base of users
  - ✓Prioritize familiar user experience
  - ✓Free, open-source software
  - ✓Easily extendable with new models and functionalities
  - ❑Provide quality documentation, tutorials etc.
- We can check the first three boxes with QGIS
  - Free, open-source GIS toolbox
  - Large user base
  - Easily extendable with plugins
  - Very wide range of applications



What?

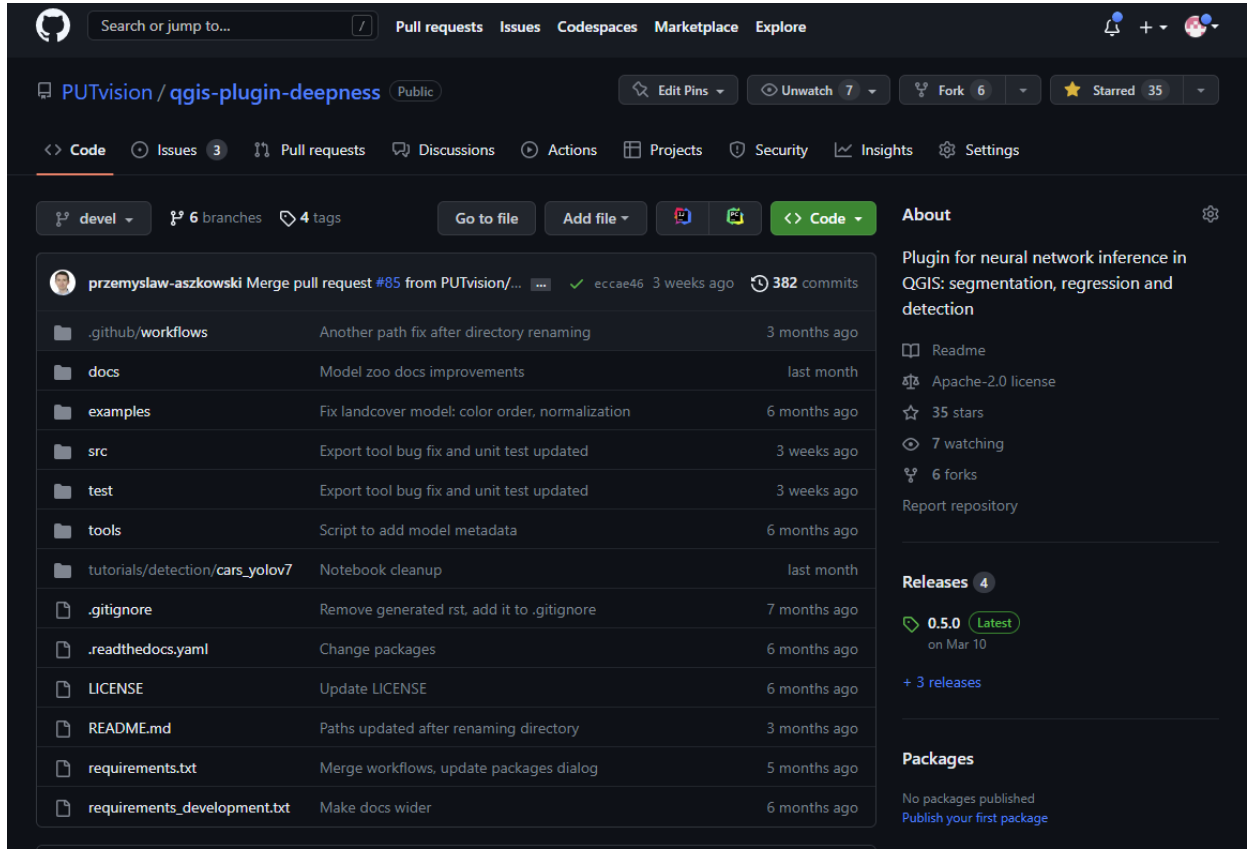
Deepness: Deep Neural Remote Sensing QGIS Plugin

- Available through QGIS Python Plugin Repository for easy installation (reaching 3000 downloads)
  - https://plugins.qgis.org/plugins/deepness/



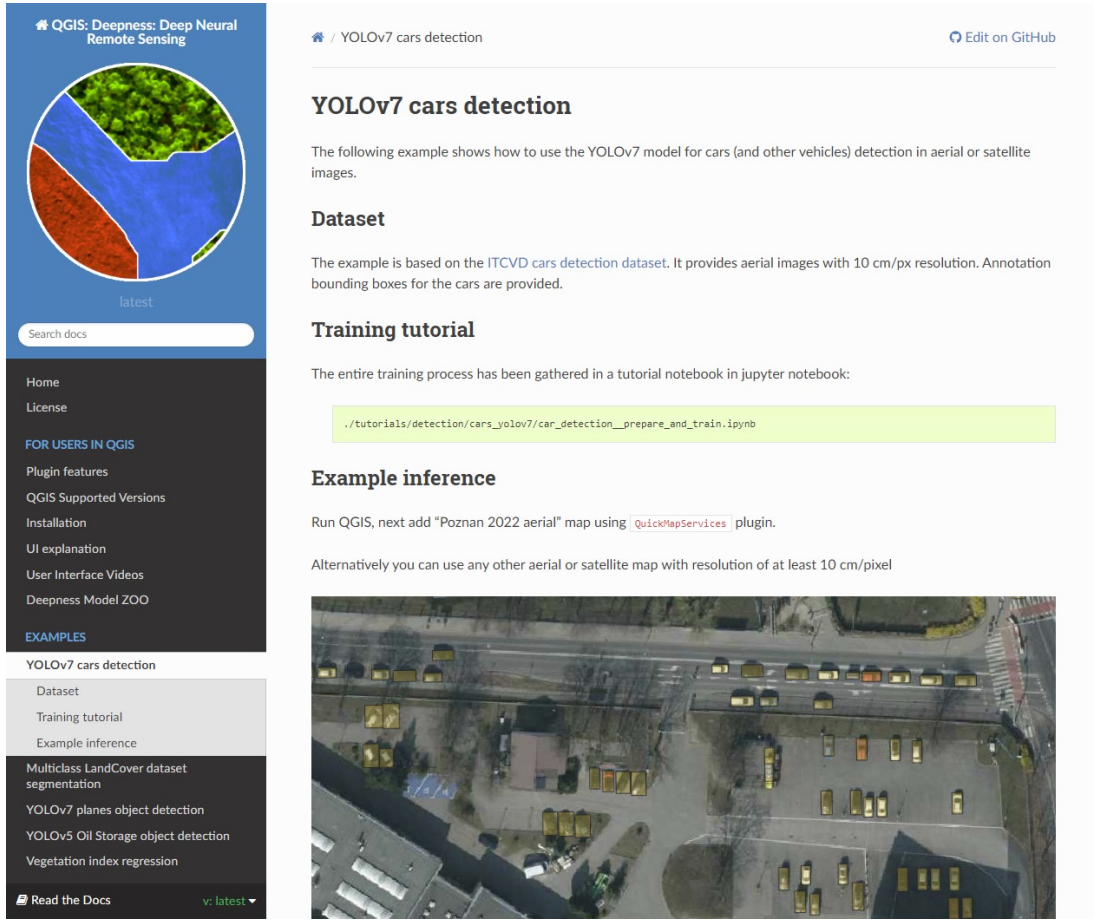
Deepness: Deep Neural Remote Sensing QGIS Plugin

- Open-source, with GitHub repository
  - https://plugins.qgis.org/plugins/deepness/
- Anyone can contribute and is welcome to do so
  - All levels of expertise welcome, there's always something we can improve



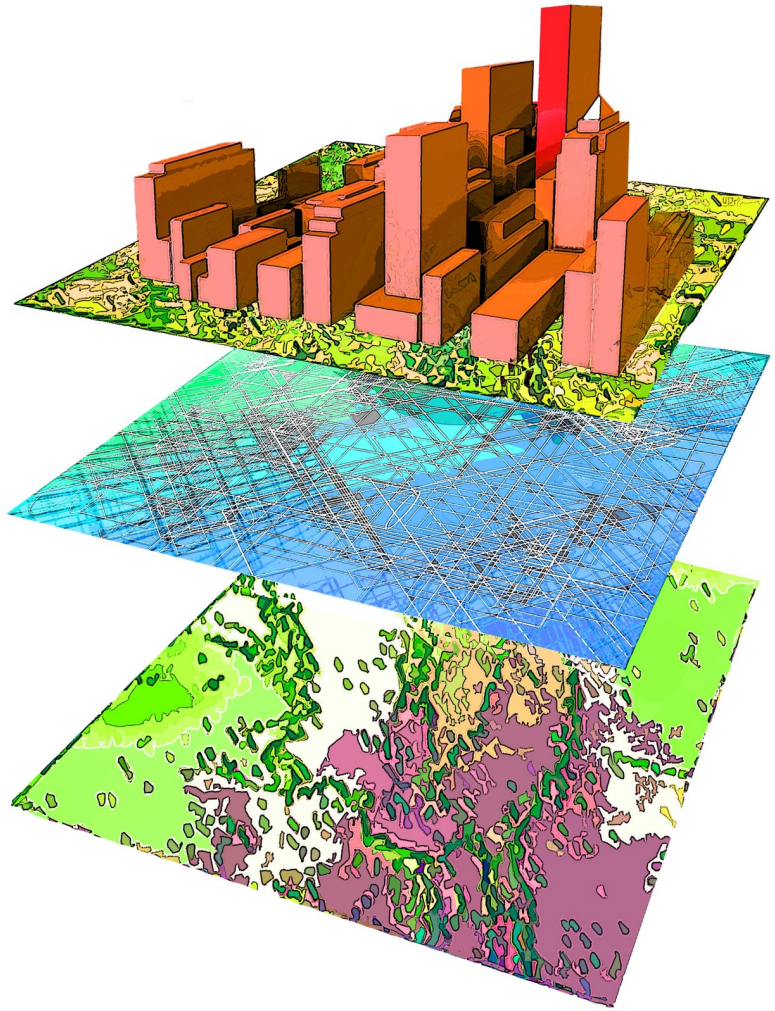
Deepness: Deep Neural Remote Sensing QGIS Plugin

- Comprehensive documentation with examples and tutorials
  - https://qgis-plugin-deepness.readthedocs.io/
- Very important but often neglected part of any software
- Documentation for any type of user
  - Using the plugin
  - Developing NN models
  - Plugin development



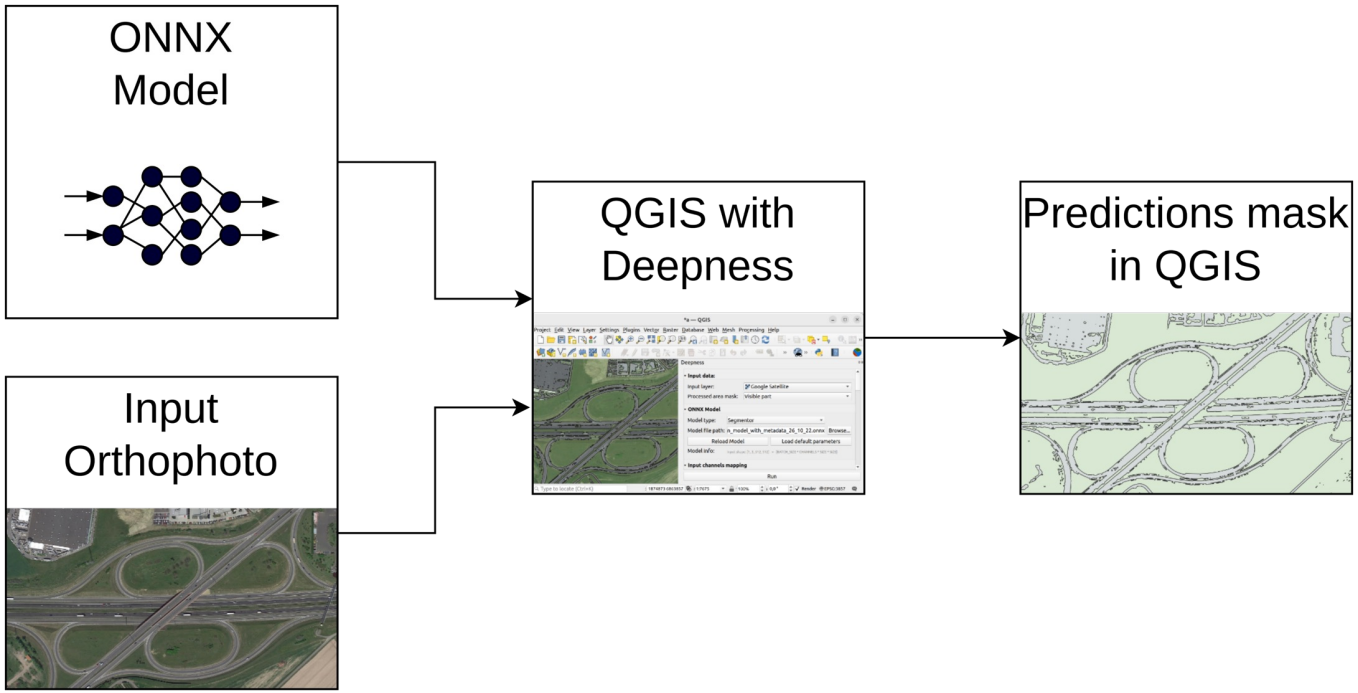
Deepness: Deep Neural Remote Sensing QGIS Plugin

- QGIS supports a lot of data sources
  - Satellite, aerial and UAV
  - A range of modalities and data types
  - Maps, surveying results
- Anything that can be represented as a layer and rasterized, can be an input to the neural network
  - Input images
  - Annotations
- Outputs are also QGIS layers



Deepness: Deep Neural Remote Sensing QGIS Plugin

- Model suport is provided for the portable QNNX format
- Uses a GPU if the system provides one (will install all necessary software automatically)

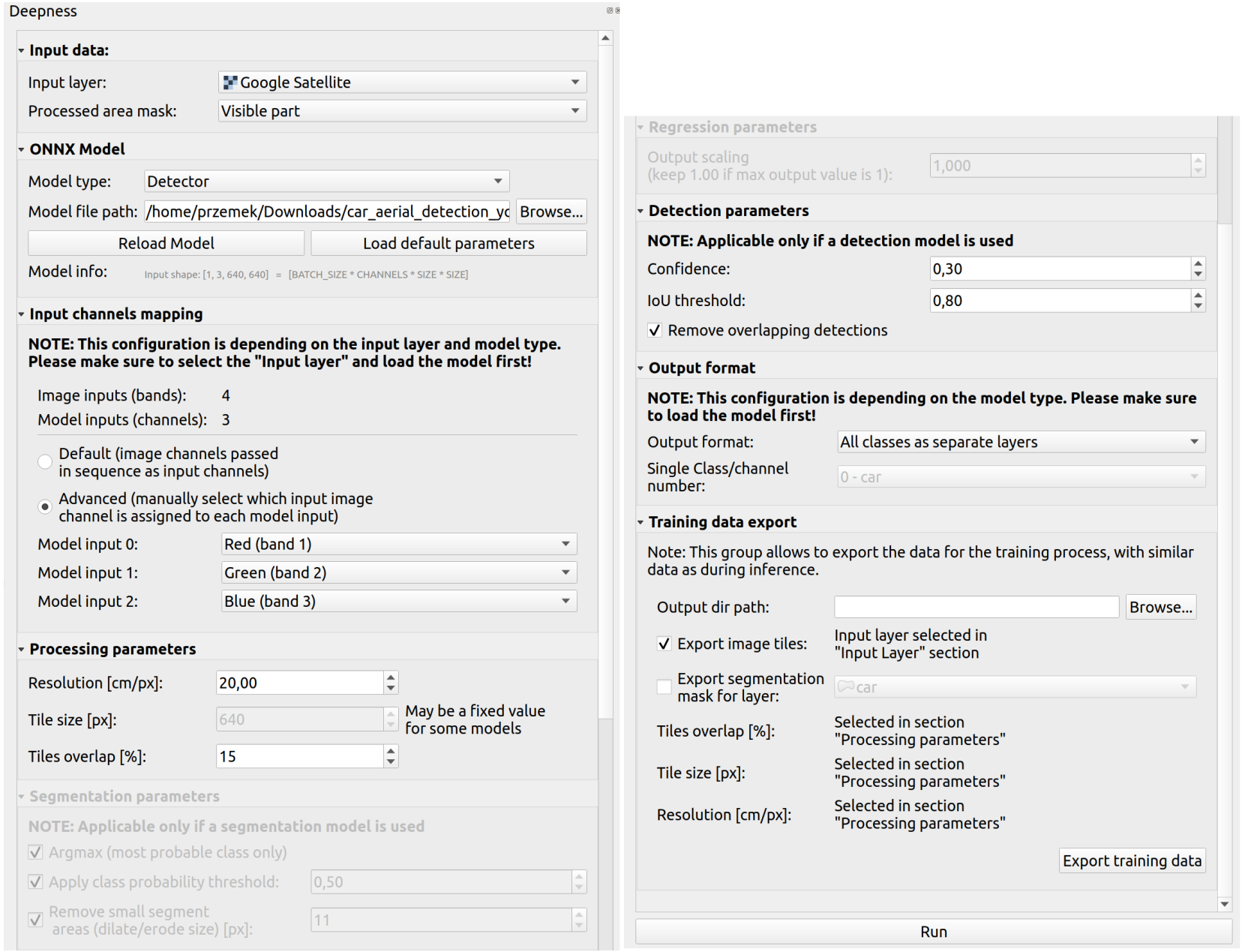


Deepness: Deep Neural Remote Sensing QGIS Plugin

- Deepness supports all major inference task types
  - Segmentation (binary, semantic, instance, …)
  - Detection (object bounding boxes, circles, …)
  - Regression (probability density, vegetation indexes, …)
  - You can set area of interest for all the above tasks
- Deepness enables easy export of data
  - Set the area of interest
  - Set channels of interest
  - Set tile size and stride

Deepness: Deep Neural Remote Sensing QGIS Plugin

- Typical Deepness workflow
  - Load data
  - Select input layers
  - Select a model for inference
  - Select model's parameters (or stick with the defaults)
  - Run processing
  - Review/save results


Deepness
Input data:
Input layer:
Google Satellite
Processed area mask:
Visible part
ONNX Model
Model type:
Detector
Model file path:
/home/przemek/Downloads/car_aerial_detection_yc
Browse...
Reload Model
Load default parameters
Model info:
Input shape: [1, 3, 640, 640] = [BATCH_SIZE * CHANNELS * SIZE * SIZE]
Input channels mapping
NOTE: This configuration is depending on the input layer and model type. Please make sure to select the "Input layer" and load the model first!
Image inputs (bands): 4
Model inputs (channels): 3
Default (image channels passed in sequence as input channels)
Advanced (manually select which input image channel is assigned to each model input)
Model input 0:
Red (band 1)
Model input 1:
Green (band 2)
Model input 2:
Blue (band 3)
Processing parameters
Resolution [cm/px]:
20,00
Tile size [px]:
640
May be a fixed value for some models
Tiles overlap [%]:
15
Segmentation parameters
NOTE: Applicable only if a segmentation model is used
Argmax (most probable class only)
Apply class probability threshold:
0,50
Remove small segment areas (dilate/erode size) [px]:
11
Regression parameters
Output scaling (keep 1.00 if max output value is 1):
1,000
Detection parameters
NOTE: Applicable only if a detection model is used
Confidence:
0,30
IoU threshold:
0,80
Remove overlapping detections
Output format
NOTE: This configuration is depending on the model type. Please make sure to load the model first!
Output format:
All classes as separate layers
Single Class/channel number:
0 - car
Training data export
Note: This group allows to export the data for the training process, with similar data as during inference.
Output dir path:
Browse...
Export image tiles:
Input layer selected in "Input Layer" section
Export segmentation mask for layer:
car
Tiles overlap [%]:
Selected in section "Processing parameters"
Tile size [px]:
Selected in section "Processing parameters"
Resolution [cm/px]:
Selected in section "Processing parameters"
Export training data
Run

Deepness: Deep Neural Remote Sensing QGIS Plugin

- Available models

Function	GSD [cm]	Description
Land cover segmentation	40	Trained on landcover.ai data, with classes for woodlands, buildings, water, roads
Corn field damage segmentation	3	Trained on an in-house dataset of data containing annotations for wildlife-induced corn plants damage
Road segmentation	21	Uses Google Earth images as input, generates binary mask
Airplane detection	70	YOLOv7-tiny version trained on the Airbus Airplane Detecition dataset
Oil storage tank detection	150	YOLOv5-m version trained on Airbus Oil Storage detection dataset
Car detection	10	YOLOv7-m for car detection on aerial imagery, trained on ITCVD dataset

Summary

- We have the software up and running and it’s a good base for further development of deep learning models in a wide range of different Earth Observations tasks
- Additional models are on the way, most of them suggested by potential end users:
  - Forest fire probability prediction
  - Biomass content prediction for grazing
  - Heat islands detection and tracking in urban areas
  - Land fertility prediction from hyperspectral data
- We intend to shift focus to model development, but suggestions for additional functionalities are welcome
- We hope to increase the community involvement

Thank you for your attention

visit us at: vision.put.poznan.pl

or write us an email: marek.kraft@put.poznan.pl