ITUEvents

ITU-ML5G-PS-012: ML5G-PHY: Beam Selection (Universidade Federal do Pará, Brazil)

07 August 2020

ITU

AI/ML in 5G

Challenge

Applying machine learning in communication networks

ai5gchallenge@itu.int

Bronze sponsor:

Organized by:

ITU Artificial Intelligence/Machine Learning in 5G Challenge

An Overview of the ITU-ML5G-PS-012 "ML5G-PHY [beam selection]"

Aldebaro Klautau
Federal University of Pará (UFPA) / LASSE
http://ai5gchallenge.ufpa.br

Joint work with Profs. Diego Gomes (UNIFESSPA), Francisco Müller (UFPA), Nuria González-Prelcic (NCSU), Robert Heath (UT) and several students

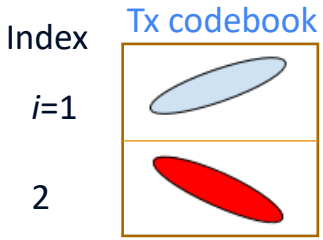
August 07, 2020

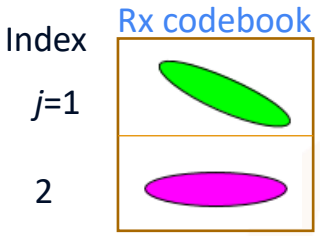
Hints

1. From our experiments: it is harder to work with “raw” data (Raymobtime), but a customized frontend can outperform the baseline frontend that we provided ➔ will release extra slides
2. We have better images: raw_data\images_data_s008_v2
   It outperformed the previous dataset at images_data_s008
3. Use Raymobtime_visualizer.py to understand your scenario and eventually “debug”: https://github.com/lasseufpa/ITU-Challenge-ML5G-PHY/blob/master/Visualizer/Raymobtime_visualizer.py specially to understand the bias that the scenario imposes (for example: unbalanced dataset)
4. We will provide extra data in September to help tuning the algorithms and checking the whole submission procedure

Example: Neural network for beam-selection

Example with $M_t = M_r = 2$ vectors per codebook

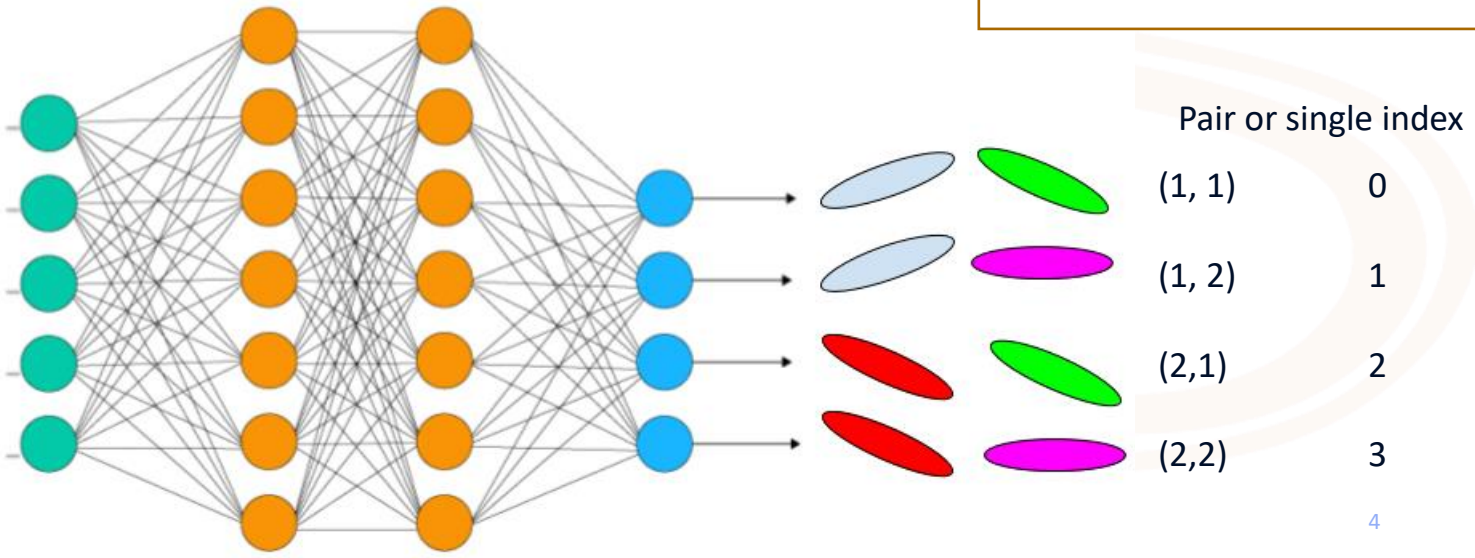




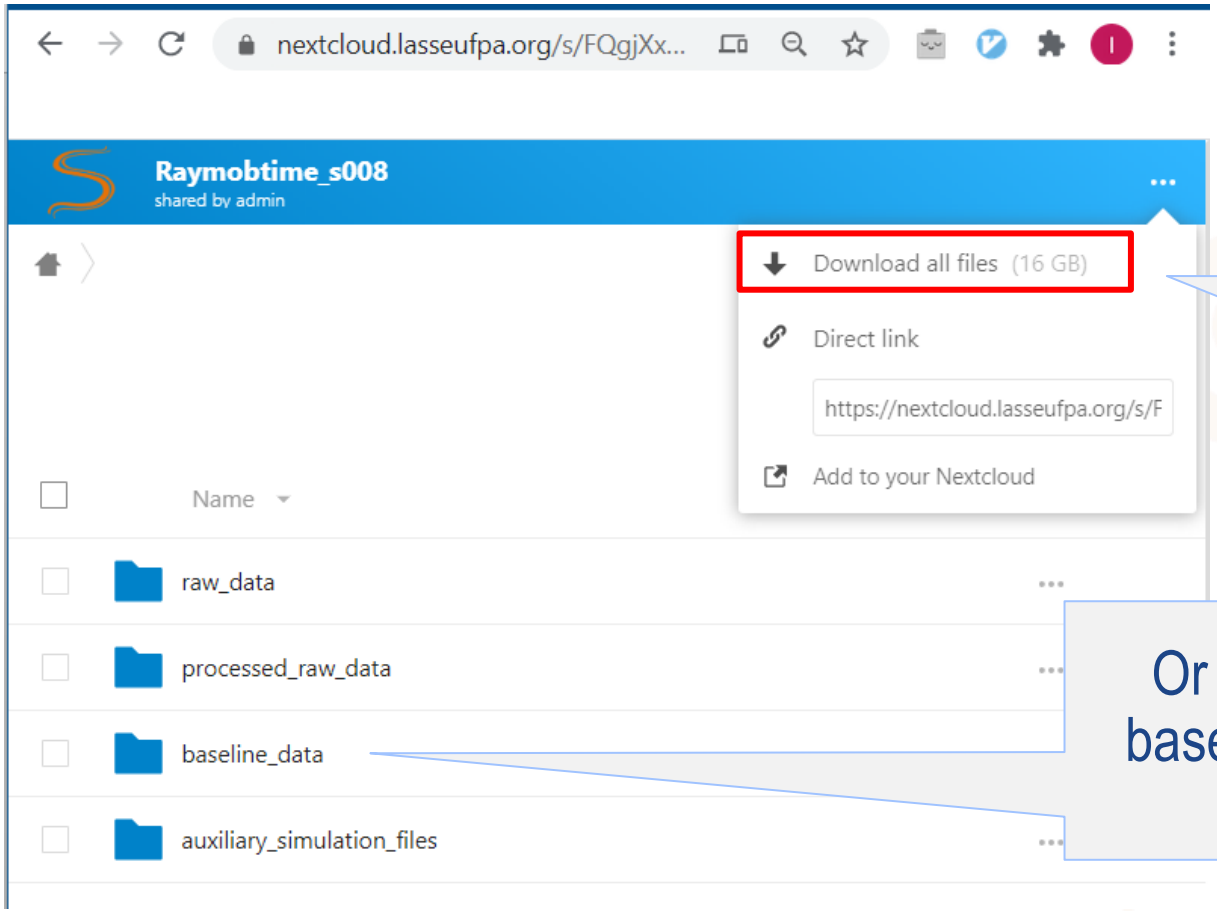
Goal: maximize magnitude of *combined channel*:

$$y(i,j) = |\boldsymbol{w}_j^H \boldsymbol{H} \boldsymbol{f}_i|$$

Features obtained from baseline **or** customized frontend



Two options: Work with baseline or raw data



Site: http://ai5gchallenge.ufpa.br/ or directly: https://www.lasse.ufpa.br/raymobtime/

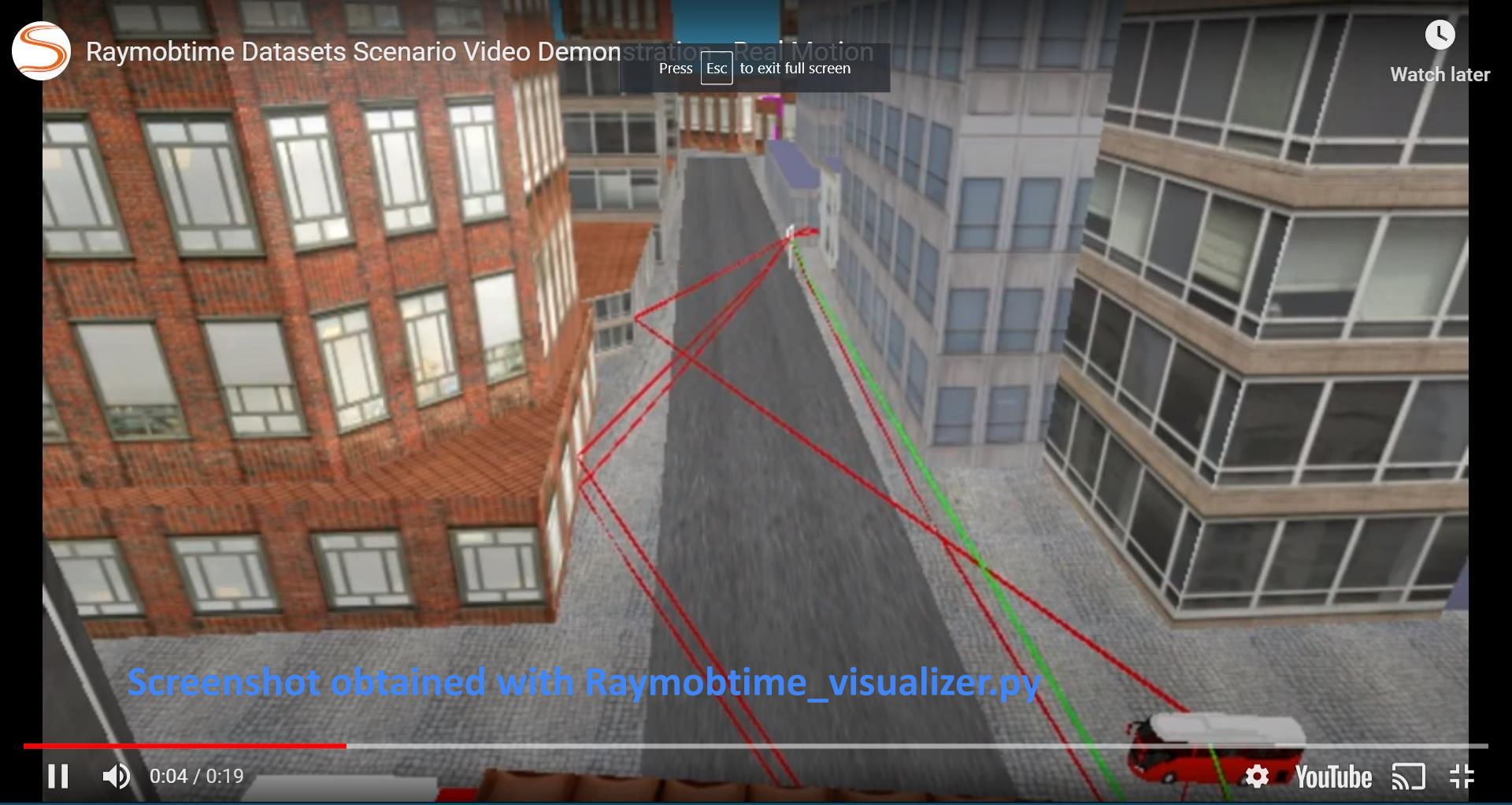
Download all data (including raw, ~16 GB)

Or download only baseline data (~512 MB)

Repository structure:

```
Dataset Home
├──auxiliary_simulation_files
│       s008_Rosslyn_10MobileRx_60GHz_2086episodes_1scenes_Ts1s_InSite3.2.zip
│       s008_SUMO_config_files.zip
│
├──baseline_data   Baseline
│   ├──beam_output
│   │       beams_output_train.npz          Combined channel for each pair of beams
│   │       beams_output_validation.npz
│   │
│   ├──coord_input
│   │       coord_train.npz                 Position features: (x,y) for receiver
│   │       coord_validation.npz
│   │
│   ├──image_input
│   │       img_input_train_20.npz          Resampled images
│   │       img_input_validation_20.npz
│   │
│   └──lidar_input
│           lidar_train.npz                 Quantized LIDAR PCDs
│           lidar_validation.npz
│
├──processed_raw_data
│       lidar_data_s008.zip
│
└──raw_data   Raw
    │   CoordVehiclesRxPerScene_s008.csv
    │   ray_tracing_data_s008_carrier60GHz.zip
    │   s008_Blensor_rosslyn_scans_lidar.zip
    │
    └──image_data_s008
            camera1.zip
            camera2.zip
            camera3.zip
```

← New version (v2)



Reviewing the hints

1. From our experiments: it is harder to work with “raw” data (Raymobtime), but a customized frontend can outperform the baseline frontend that we provided ➔ will release extra slides
2. We have better images: raw_data\images_data_s008_v2
   It outperformed the previous dataset at images_data_s008
3. Use Raymobtime_visualizer.py to understand your scenario and eventually “debug”: https://github.com/lasseufpa/ITU-Challenge-ML5G-PHY/blob/master/Visualizer/Raymobtime_visualizer.py specially to understand the bias that the scenario imposes (for example: unbalanced dataset)
4. We will provide extra data in September to help tuning the algorithms and checking the whole submission procedure

Thanks to all Raymobtime team

Aldebaro Klautau

Ailton Oliveira

Arthur Nascimento

Brenda Vilas Boas

Ilan Correa

Diego Gomes

Francisco Müller

Robert Heath Jr

Isabela Trindade

Marcus Dias

Virgínia Tavares

Pedro Batista

Walter Tadeu

Nuria González-Prelcic

Yuyang Wang

Jamelly Ferreira