Calculating the solar potential of rooftops in cities

Daniel Bumke, Jérémie Joudioux, Corstiaen Versteegh

Data Science Retreat

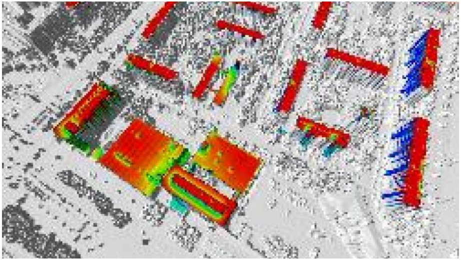
https://github.com/moreshiny/solar_berlin

Mentor: Adam Green
with help and comments of Tristan Behrens and Markus Hinsche

Some cities produce solar energy potential maps

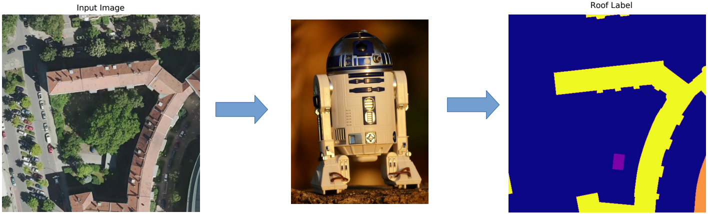
- Solar potential maps guide solar investment decisions and priorities
- Solar potential maps reflect total potential solar energy output of an area

Producing solar maps is expensive



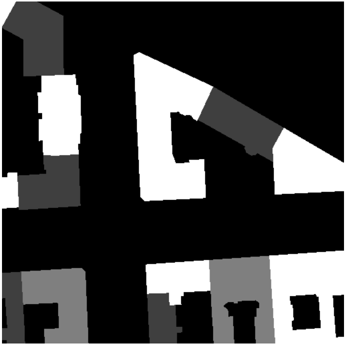
- Combination of urban planning information and 3D data from LIDAR measurements
- Estimates based on roof size, angle, shading, and orientation
- Problem: Data are expensive to obtain and not widely available

Can deep learning help create maps more easily?



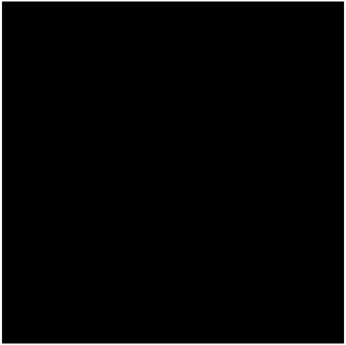
- Identify buildings
- Categorise roofs by suitability for solar
- Output estimates for downstream estimation of solar potential of a building, area, or entire city

Good news: City of Berlin provides labeled data!

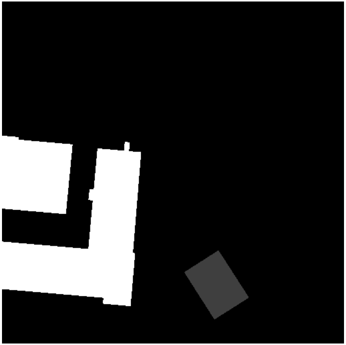


- Aerial images from 2013
- Solar potential maps based on 2013 LIDAR data
- Four categories of roofs: very suitable, quite suitable, somewhat suitable, not suitable

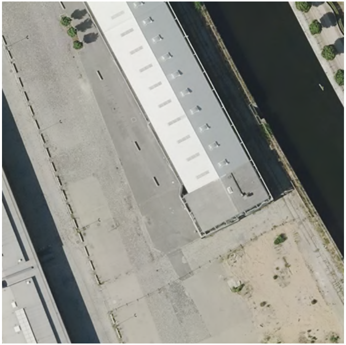
Bad news: Labels are really not that good

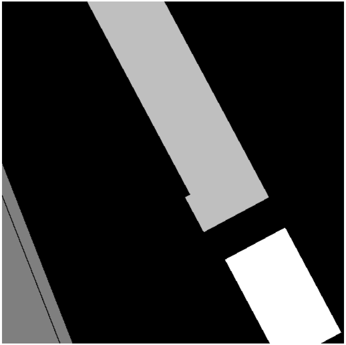


Bad news: Labels are really not that good

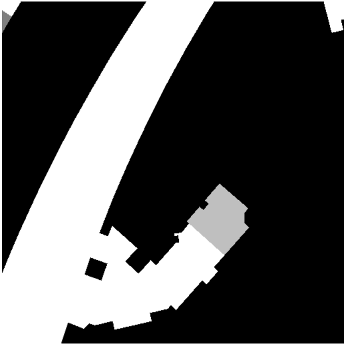


Bad news: Labels are really not that good





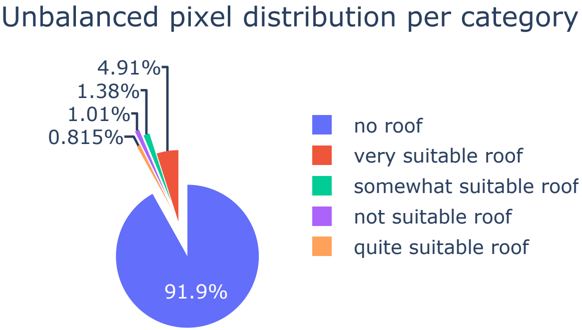
Bad news: Labels are really not that good



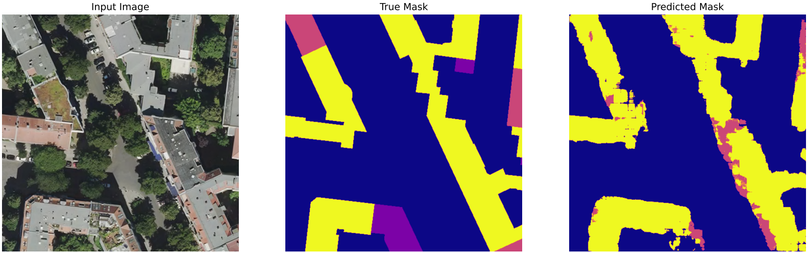
Berlin training data are very unbalanced

After data extraction,

- ca. 100 000 tiles of size $(512, 512)$ covering all of Berlin
- 45% masks $\rightarrow$ completely empty
- 40% masks $\rightarrow$ less than 20% roof coverage



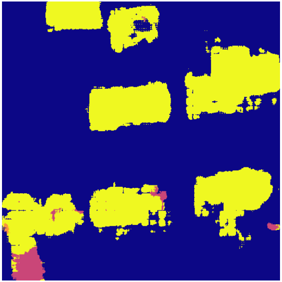
We trained two models on these data



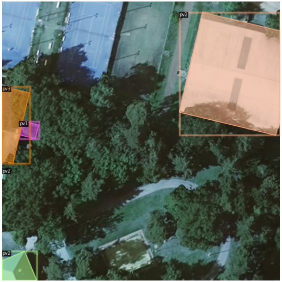
- Our models can estimate the percentage roof area of the highest suitability category in a part of a city
- All we need are aerial photographs of your city

Two models

- Pixel classifier: Unet architecture with a pre-trained backbone:
  - Roof classifier.
  - Multiclass pixel classifier.



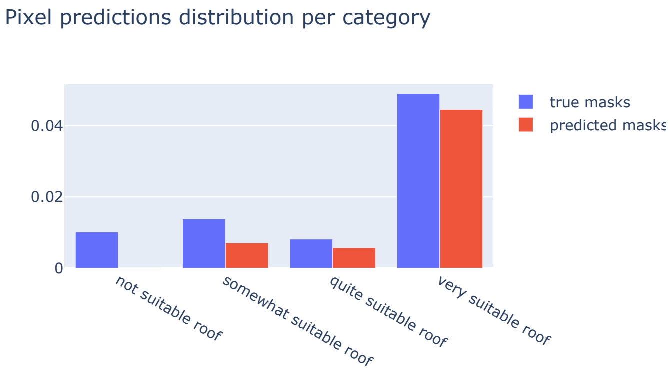
- Object detector and classifier: Mask R-CNN from Detectron2.



Roof classifier

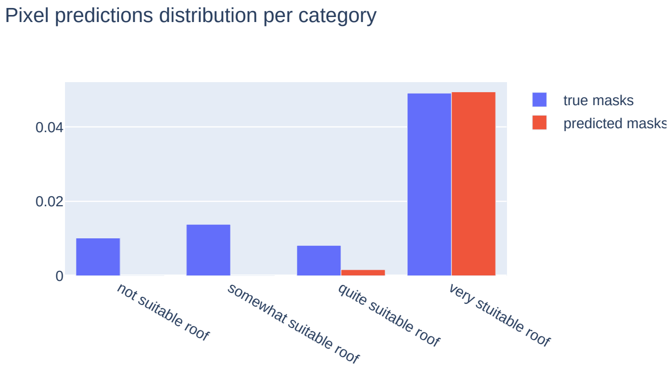
- On a cleaned dataset, the validation metrics are
  Accuracy: 0.97, Precision: 0.82, Recall: 0.66, IoU: 0.57
- Used to identify incorrect masks in the large dataset

Multiclass pixel classifier.



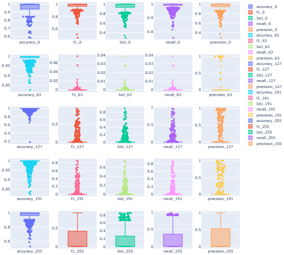
- The model predicts 4.46% of "very suitable roof" on a test set of Berlin (true value 4.91%)

Mask R - CNN.



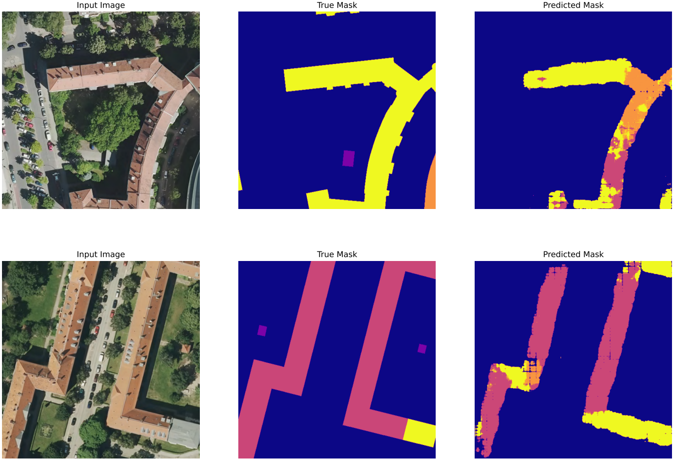
- The model predicts 4.94% of "very suitable roof" on a test set of Berlin (true value 4.91%)

The models perform poorly at identifying and categorising individual buildings

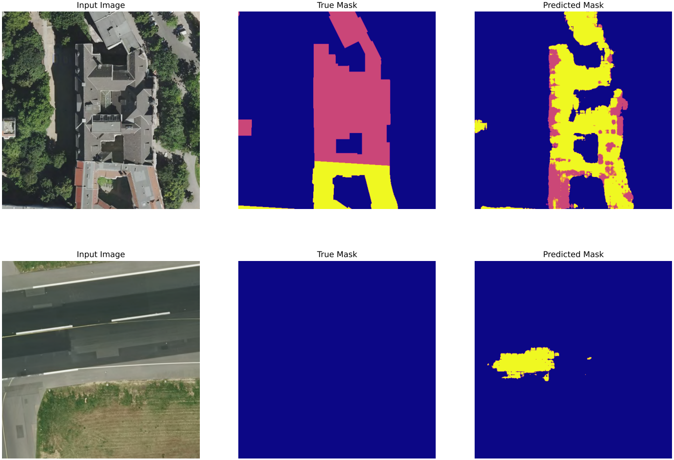


The metrics are . . . interesting.

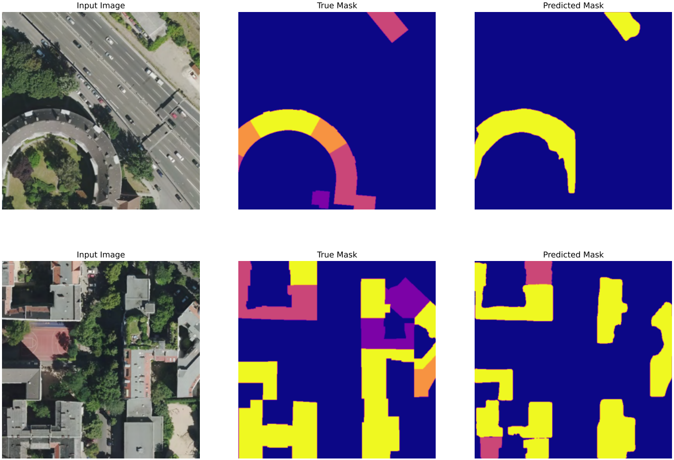
What predictions look like - Multiclass pixel classifier



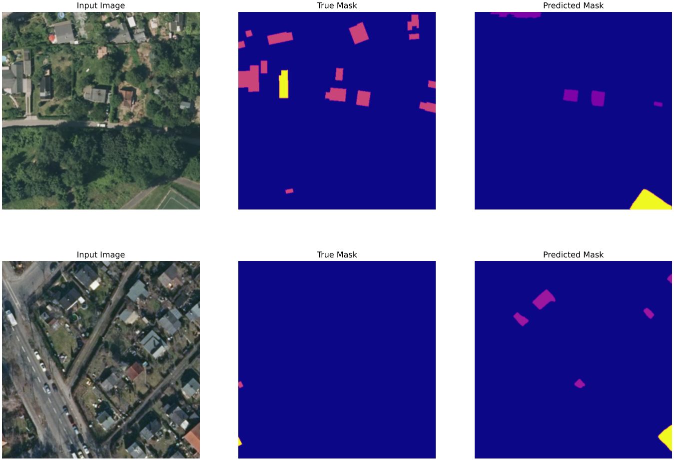
What predictions look like - Multiclass pixel classifier



What predictions look like - Mask R-CNN



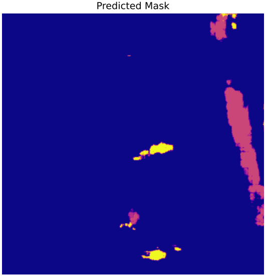
What predictions look like - Mask R-CNN



What other limitations are there?

- The model has learned from mislabeled data (partial cleaning)
- The model has learned only from aerial photographs of one resolution
- The model knows only what roofs in Berlin look like

What's happening in Leibnitz?



What's Next

- Data cleaning, data cleaning, data cleaning
- Dealing with class imbalance
- Better model: Work with a transformer; work with a model pre-trained on aerial photographs

Meanwhile in Jakarta

Input Image

Predicted Mask