# Assignment 2: deep learning on images

## Content

* In this assignment, you will work with image data (sattelite images) and deep learning to predict swimming pool locations.
  + The data set consists of >14k images of satellite imagery containing swimming pools and can be downloaded from [this link](https://drive.google.com/file/d/1w9TMEyfCO9FUNk4AIDjDN36jIh0n97g-/view?usp=sharing) (warning: 3GB download)
  + You will also need [metadata.json](http://seppe.net/aa/assignment2/metadata.json) which contains more information about the images, in particular the coordinates of the points that describe the polygon for the swimming pool in each image. The original lat/lon coordinates are also kept for those that want to take a look on e.g. Google Maps

Goal -- pick from the list below:

* + Doable: construct a deep learning model to locate the swimming pool in a given image that contains one, using a bounding box (upright rectangle)
    - Since the given information describes a polygon, you will need to preprocess a bit and find the tightest rectangle
    - You can consider whether you want to predict the upright rectangle or the best fitting rotated one
    - Note that all images contain swimming pools, and they are located in the center
    - Also, all images are the same size
  + More difficult: fully predict the bounding polygon
    - There are different ways to approach is. Predicting a list of points is probably difficult, so a per-pixel (segmentation) model might be more suitable. This can also easily be obtained from the given information: all pixels inside of the polygon belong to the swimming pool
  + Think about how you can prevent overfitting to e.g. also handle images for which the swimming pool is not in the center
  + Also think about how and whether you can deal with images that do not contain a swimming pool
  + Finally, can you model also handle images of a different size. If not, which preprocessing would you apply first?

Use can use any deep learning library you want, using pre-trained models, image augmentation etc. is fine. As stated in class, you should approach this as a "how far can I get in a small amount of time" style project like you would be facing in real life. You can also take a look at exploring (or developing) labeling tools to refine / adjust the positions of the polygon vectors - they're not always perfectly positioned!

The second part of your lab report should contain:

* + Overview of your full pipe line, including architecture, trade-offs, ways used to prevent overfitting, other steps you took
  + Results based on your chosen evaluation metric (take a look at e.g. the "intersection over union" metric)
  + Illustration of your model's predictions on a test image (or one of your own)
  + You don't need to hand in any intermediate predictions or solutions

The data was gathered using [Overpy](https://python-overpy.readthedocs.io/en/latest/) and the [Overpass API](https://overpass-turbo.eu/) together with the [Mapbox static maps API](https://www.mapbox.com/static-maps) for the images themselves. The Overpass query to obtain the locations and polygons of swimming pools in the South of France was:

result = api.query("""

nwr["leisure"="swimming\_pool"]["access"="private"](42.633959,-1.798096,44.789632,3.365479);

out center;

""")