ENEL 645 Project: Object detection in images with machine learning
Nov, 2021

Objective: In this project, you will be applying the object detection algorithm on a given dataset. The main goal is to understand how to detect the objects in an image.

Task: Each group will be assigned a specific dataset and algorithm for object detection. There will be two steps which will guide you to detect the objects. The detector will be trained on the training sets and will be tested on testing image from the testing dataset.

Deliverables:

- A final report on the following
    - Title page (group members) and chosen dataset and algorithm number
    - Part I: Data processing and understanding
        - Explain the data, algorithm, training
    - Part II: Results of chosen algorithm/model with evaluation
- Links to code, results and trained models (and run instructions)
- Links to the datasets gathered for training and test

Part I: Data preprocessing and understanding:

1) Download the data from respective links. Use some software (QGIS, Google Earth, python) to visualize some of the data.
2) The output of the algorithm will be bounding box with class name and probability scores on the image.

Part II. Using the assigned algorithm, the evaluation will be based on the following metrics below.

1) Export your test image in a compatible format for evaluation. This could involve splitting it into smaller sub-images for evaluation and combining the result.
2) Evaluate algorithm(s) using the following metrics:
    a. Accuracy of detection via mean average precision (mAP) and error percentage.
    b. If your algorithm outputs bounding boxes, accuracy given 50% overlap of boxes on your test images compared to labels.
3) Show/attach output of bounded box test image vs labelled groundtruth image to prove your algorithm is working as intended.

Note: The image processing algorithms may require a reasonable computation power, feel free to search or use known techniques to reduce the input size, dimensionality reduction. In addition, you may have free access to high-end GPU's with Google Collab/cloud, please make backup plans and files as needed: https://colab.research.google.com/notebooks/intro.ipynb

Note: Labelling your own images: When using additional images or your own dataset. You can use LabelImg, LabelMe, other software or ad-hoc to label objects in the images. The tools are quite intuitive and easy to learn.

Additional Information: Image processing:
Figure 1 and Figure 2 show sample outputs of models.
An image is simply matrices, where each pixel represents an intensity in the color (for example Red,Green,Blue there are 256 different intensities) from value 0 to 255 (or 0 to 1 if normalized).

Top left corner is (x,y)=(0,0). Bottom right corner is (x,y)=(width,height) of image. So this image here is (800,800) pixels. This image is an RGB image represented as a matrix of size [x,y] x 3 for each R,G,B channels = [800,800,3]

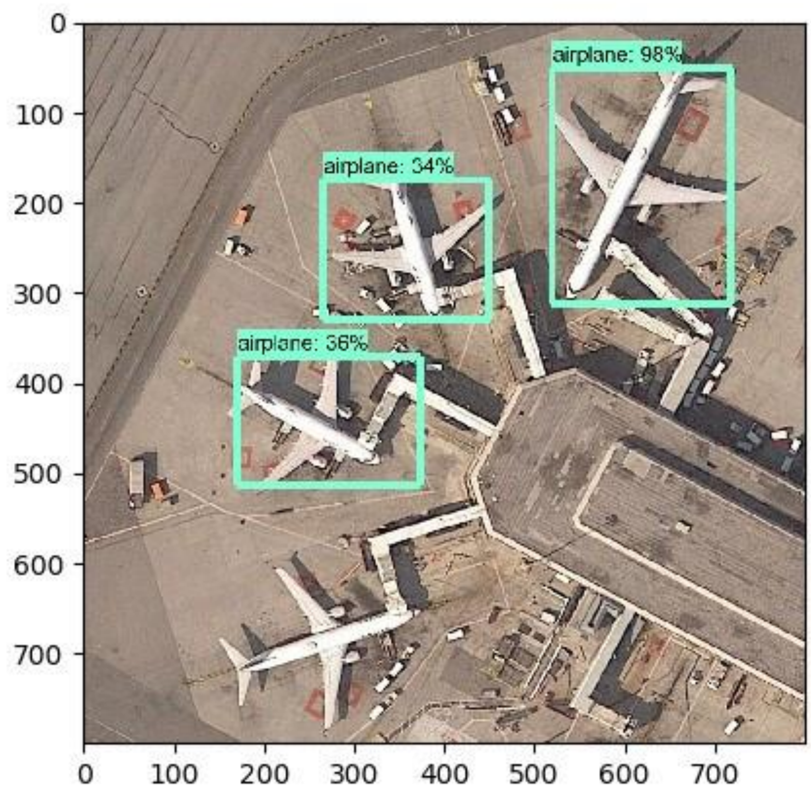

Fig 1. Sample image. This [800,800,3] image is passed to the algorithm, which outputs class probabilities and boxes. For this image, we only display probability of airplane and ignore vehicle/ship classes.

Background on object detection: Object detection is the task of localizing (where is the object?) and identifying (classifying what is the object) multiple objects in a single image.
Localizing: Find bounding boxes, or the coordinates where the object is. [xmin,xmax,ymin,ymax]
So one of the airplane locations is roughly [180,400,380,510]. This is already drawn as the green box in the figure above.
Classifying: For each box, also output the softmax probability of the classes. The corresponding probability is 36% to airplane class. (the other 64% goes to vehicle and ship class which we trained (3 object classifier)

Sometimes the resolution or location does not permit "ships/airplanes" as objects, so we consider buildings or land types as an object. For example, in a SAR image obtained right before the conclusion of this site: https://medium.com/gsi-technology/a-beginners-guide-to-segmentation-in-satellite-images-9c00d2028d52
A SAR image (left) rendered to 0-255, and (right) a sample building segmentation. White corresponds to the pixels of buildings.

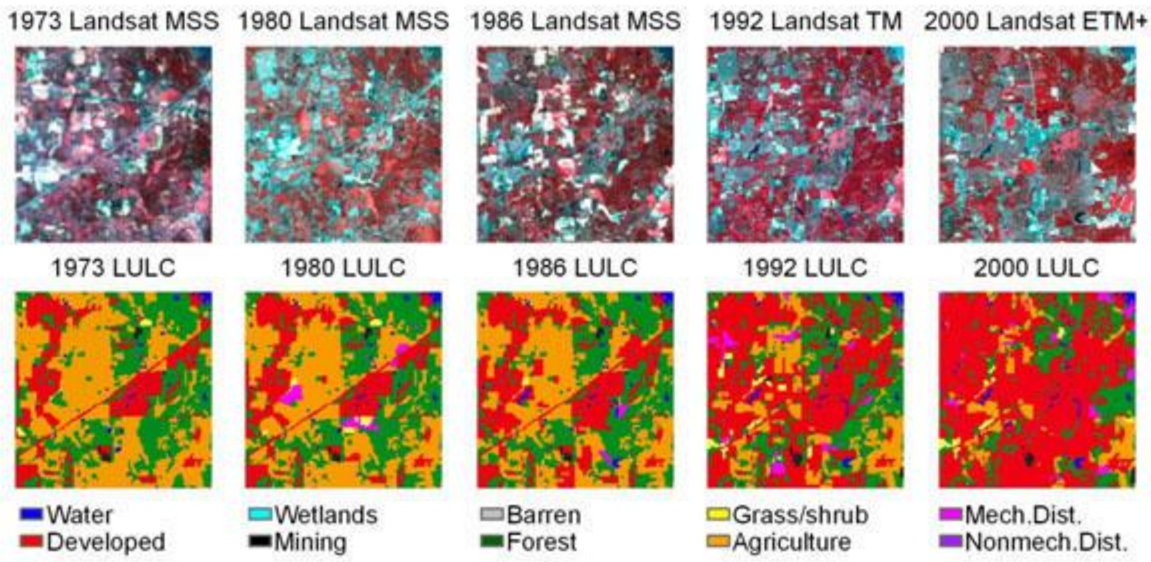

Fig 2: A common land use land cover map derived from LANDSAT over Mississippi. (Public domain) https://www.usgs.gov/media/images/5-maps-dates-land-use-and-land-cover-data