Training a Model with Custom Data

Using the YOLOv5 Workflow

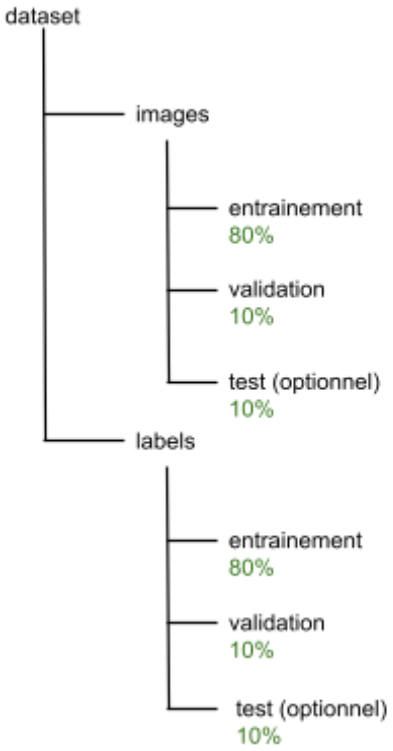
YOLOv5

- Built to detect objects in images
- Trained with a dataset of real images (ImageNet)
- Default model based on 80 object classes

The goal of the training is to adapt the model to our datasets and our classes, for it to detect the objects we want it to.

Dataset

- Annotate the objects of interest on the images to create labels : objects are defined by the position and dimensions of the boxes
- In the label file (.txt), each object corresponds to one row, in `class x_center y_center width height` format
- Organize labels and images in directories



The images and files of images and labels must have the same name.

Config file

The `dataset.yaml` file contains :

- The dataset directory path
- The relative paths to the training, validation and test directories
- A class names dictionary: in our case, the only class is `0: illustration`

```
path: ../dataset # dataset root dir
train: images/train  # training images (relative to 'path')
val: images/val  # validation images (relative to 'path')
test:  # test images (optional)

# Classes
names:
  0: illustration
```

Notes

- The training dataset is composed of the images the researchers annotate
- Our model needs to be able to detect each illustration individually rather than extracting illustrated pages as a unique illustration
- The model is trained with both images of manuscripts and printed books: does it impact the quality of the extraction?