

PowerAI Vision Workshop Exercise 2 – Object detection

In this exercise you will practice Object Detection. With Object Detection, a ML algorithm learns to locate and identify defined objects in a scene. It is a sophisticated and very useful technique for several AI applications. The goal of the exercise is to familiarize with the user interface, the pipeline and OD related tasks.

1. Create a **dataset** called Buildings.
 - a. On the welcome screen, press 'Get Started' (alternatively, select Data Sets from the menu bar). The Data Sets screen appears.
 - b. Press the plus sign to 'Create a new dataset' on the leftmost tile. (.zip file upload is valid only for datasets exported from PAIV). Name the data set 'Buildings'.
 - c. Click on the tile 'Buildings'. A 'Data set / Buildings' screen will open.
 - d. Click on 'Import files' under the leftmost tile 'Drop files here'. Select the 'buildings.zip' file for upload.
 - e. The 'Buildings' data set is created. Duplicate the Data set with a new name 'Building_g_b' (for good and best).
 - f. At the bottom of the pane, select 'Items per page: 100' to view as many images as possible. This will likely improve productivity.
 - g. Set aside some pictures for testing your model later. The pictures should be deleted from the dataset and not used for training.
2. Labeling **preparation**. Because labeling objects is tedious, we would use an iterative process that improves the effectiveness of our work: with minimal labeling, train a simple, rough model, that will be used to help label more images; we will do only the correction manually.
 - a. Open the 'Building_g_b' dataset.
 - b. Select all the images that do not contain chimneys on the buildings. Delete them.
 - c. Create two categories: 'good' and 'best'. We will use these to organize images.
 - d. Select 10-15 images that are the most relevant and have clear visibility for chimneys on the building. Assign the category 'best' to these images.
 - e. Select another bunch (~20) images that are fairly well representing the chimneys. Assign the category 'good'.
 - f. From categories, unselect 'good' and 'best'. Only uncategorized images are shown. Select all and delete them.
 - g. The dataset now contains only the good and the best images. We will use it in phase 2 of the iterative training.

- h. Go back to the Data Sets screen.
 - i. Duplicate the data set with a new name 'Buildings_b' (for best).
 - j. Select the category 'good' images and delete all.
3. Define and **label** objects.
- a. *Note: we will use bounding box in the first round, just because it is quicker. Should you have time, try Polygon selection and use the Detectron model for training.*
 - b. *Note: before you start, click on an image then click on 'Label objects' and familiarize yourself with the interface. Look at options (represented by a small gear) and Keyboard shortcuts. The shortcuts availability depend on the platform / keyboard you use.*
 - c. On each image, perform the following steps.
 - i. Click on the image and select 'label objects'
 - ii. If there is no object category, select 'Add new' and create an object. You will use 'chimney' objects.
 - iii. Select the object tag and draw a bounding box around the object.
 - iv. All objects of which the majority is visible should be labeled. Do not label objects that are not clearly representing what you want to recognize. Do not leave much empty space around the object. Do not let the bounding box get out of the image boundary.
 - v. Alternatively, you can select multiple images from the dataset and select pictures to work with from the labeling interface. This improves efficiency of your work.
 - d. Once you finished labeling, augment the data with horizontal flipping and tweaking color. Create a new dataset e.g. 'Building_b_aug'.
4. Select the new dataset and start **training** for Object detection (R-CNN)
- a. *Note: For object detection, the training can be stopped without losing the model. Once training loss flattens, there is no need to wait any longer, because the model will not improve.*
5. Once the model is ready (it will not be a very good model, we just hope it is better than manual work), deploy it.
6. Go back to Data sets and work with 'Buildings_b_g'.
- a. Select Auto Label, and from the deployed models list specify the recently trained and deployed model.
 - b. Let Autolabel run for some time.
 - c. On each image, perform the following steps:
 - i. Click on the image and select 'label objects'

- ii. If there is no object category, select 'Add new' and create an object. You will use 'chimney' objects.
 - iii. Check and correct the object bounding boxes and labels assigned by auto labeling.
 - iv. Complete labeling for missing boxes and objects.
 - v. Select the object tag and draw a bounding box around the object.
 - vi. All objects of which the majority is visible should be labeled. Do not label objects that are not clearly representing that you want to recognize. Do not leave much empty space around the object. Do not let the bounding box get out of the image boundary.
- d. Once you finished labeling, augment the data with horizontal flipping, cropping, rotation and color. Create a new dataset, e.g. 'Building_b_g_aug'.
- 7. Select this dataset and start **training** for Object detection (R-CNN). Before that, if you need to free up resources (training will require a GPU), you can delete the previously deployed model (it was bad anyway).
 - a. *Note: For object detection, the training can be stopped without losing the model. Once training loss flattens, there is no need to wait any longer, because the model will not improve.*
- 8. Once the model is ready (it will still not be an excellent model), deploy it.
- 9. Go back to Data sets and work with 'Buildings'.
 - a. Select Auto Label, and from the deployed models list specify the recently trained and deployed model.
 - b. Let Autolabel run.
 - c. On each image, perform the following steps:
 - i. Click on the image and select 'label objects'
 - ii. If there is no object category, select 'Add new' and create an object. You will use 'chimney' objects.
 - iii. Check and correct the object bounding boxes and labels assigned by auto labeling.
 - iv. Complete labeling for missing boxes and objects.
 - v. Select the object tag and draw a bounding box around the object. All objects of which the majority is visible should be labeled. Do not label objects that are not clearly representing that you want to recognize. Do not leave much empty space around the object. Do not let the bounding box get out of the image boundary.
 - d. Once you finished labeling, augment the data with horizontal flipping and color. Create a new dataset, e.g. 'Building _aug'.
- 10. Delete the deployed model if you need to free up a GPU for training.

11. Train the model using the 'Building_aug' dataset. Wait for the training loss to stabilize below your acceptance level.
12. Deploy the model. It should be a fair model.
13. Test the model for object detection using images that were not involved in the training process.
- - - IF TIME ALLOWS : - - -
14. Restart the process with Polygon selection and Segmentation (Detectron) training. The results and the testing could be slightly different:
15. Evaluate the model and compare the results. Testing shows tangible difference.
16. Export / delete / reimport / retest the model.
17. Export the (even partially) labeled dataset. Look at the exported .zip file, and the respective XML descriptions. Optionally, delete and reimport the dataset.

This concludes the exercise.