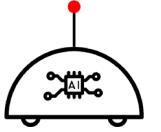


# FiftyOne

Reconocimiento de objetos

Máster en Robótica y Sistemas Inteligentes



## Installation



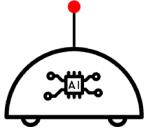
Prerequisites:

- Python 3.7 – 3.10

Installation

> pip install fiftyone





## How it works

```
In [1]: 1 import fiftyone as fo  
2 import fiftyone.zoo as foz
```

```
In [2]: 1 dataset = foz.load_zoo_dataset("open-images-v6", split="validation", max_samples=50, shuffle=True)
```

Downloading split 'validation' to 'C:\Users\virgi\fiftyone\open-images-v6\validation' if necessary

Necessary images already downloaded

Existing download of split 'validation' is sufficient

Loading existing dataset 'open-images-v6-validation-50'. To reload from disk, either delete the existing dataset or provide a custom `dataset\_name` to use

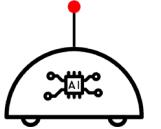
`fiftyone.zoo.datasets.load_zoo_dataset(name, split=None, splits=None, label_field=None, dataset_name=None, dataset_dir=None, download_if_necessary=True, drop_existing_dataset=False, overwrite=False, cleanup=True, **kwargs)`

Loads the dataset of the given name from the FiftyOne Dataset Zoo as a `fiftyone.core.dataset.Dataset`.

Parameters:

- name – the name of the zoo dataset to load. Call `list_zoo_datasets()` to see the available datasets
- split (None) – ("train", "validation", "test"). If neither split nor splits are provided, all available splits are loaded. Consult the documentation for the `ZooDataset` you specified to see the supported splits
- splits (None) – a list of splits to load, if applicable. Typical values are ("train", "validation", "test"). If neither split nor splits are provided, all available splits are loaded. Consult the documentation for the `ZooDataset` you specified to see the supported splits
- label\_field (None) – the label field (or prefix, if the dataset contains multiple label fields) in which to store the dataset's labels. By default, this is "ground\_truth" if the dataset contains a single label field. If the dataset contains multiple label fields and this value is not provided, the labels will be stored under dataset-specific field names
- dataset\_name (None) – an optional name to give the returned `fiftyone.core.dataset.Dataset`. By default, a name will be constructed based on the dataset and split(s) you are loading
- dataset\_dir (None) – the directory in which the dataset is stored or will be downloaded. By default, the dataset will be located in `fiftyone.config.dataset_zoo_dir`
- download\_if\_necessary (True) – whether to download the dataset if it is not found in the specified dataset directory
- drop\_existing\_dataset (False) – whether to drop an existing dataset with the same name if it exists
- overwrite (False) – whether to overwrite any existing files if the dataset is to be downloaded
- cleanup (True) – whether to cleanup any temporary files generated during download





## How it works

Datasets information at  
[https://docs.voxel51.com/user\\_guide/dataset\\_zoo/datasets.html#](https://docs.voxel51.com/user_guide/dataset_zoo/datasets.html#)

```
1 dataset = foz.load_zoo_dataset(  
2     "open-images-v6"  
3     , split="validation",  
4     label_types=["detections", "classifications"],  
5     classes=["Fedora", "Piano"],  
6     max_samples=25,  
7 )
```

Docs > FiftyOne User Guide > FiftyOne Dataset Zoo > Available Zoo Datasets

Contents

## AVAILABLE ZOO DATASETS

This page lists all of the datasets available in the Dataset Zoo.

### • NOTE

Check out the [API reference](#) for complete instructions for using the Dataset Zoo.

Dataset name	Tags	
ActivityNet 100	video, classification, action-recognition, temporal-detection	
ActivityNet 200	video, classification, action-recognition, temporal-detection	
BDD100K	image, multilabel, automotive, manual	
Caltech-101	image, classification	
Caltech-256	image, classification	

Available Zoo Datasets  
ActivityNet 100  
ActivityNet 200  
BDD100K  
Caltech-101  
Caltech-256  
CIFAR-10  
CIFAR-100  
Cityscapes  
COCO-2014  
COCO-2017  
Fashion MNIST  
Families in the Wild  
HMBD51  
ImageNet 2012  
ImageNet Sample  
Kinetics 400  
Kinetics 600  
Kinetics 700  
Kinetics 700-2020  
KITTI

("train", "test", "validation")

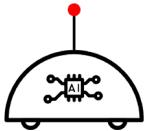
("detections", "classifications", "relationships", "segmentation")



# FiftyOne

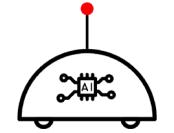
How it works

```
1 fo.launch_app(dataset)
```



The screenshot shows the FiftyOne application interface. At the top, there's a header with the logo, the name "FiftyOne", the dataset name "open-images-v6-train-validation", and various control buttons like "+ add stage", "Have a Team?", and "Show". On the left, a sidebar titled "FILTER" contains sections for "SAMPLE TAGS", "LABEL TAGS", "METADATA" (with options for metadata size, type, width, height, and number of channels), "LABELS" (with a checked checkbox for "segmentations"), and "PRIMITIVES" (with options for id, filepath, and open\_images\_id). The main area displays a 4x5 grid of images of orange cats. Each image has an orange segmentation mask overlaid, highlighting specific parts of the cat or the background. The top right of the interface shows the total sample count "10.214 samples" and a brightness slider.





## How it works

```
1 dataset.export(export_dir='open_data', dataset_type=fo.types.YOLOv5Dataset)
```

```
Found multiple fields ['detections', 'segmentations', 'relationships'] with compatible type <class 'fiftyone.core.labels.Detections'>; exporting 'detections'
```

```
INFO:fiftyone.core.collections:Found multiple fields ['detections', 'segmentations', 'relationships'] with compatible type <class 'fiftyone.core.labels.Detections'>; exporting 'detections'
```

```
100% |██████████| 50/50 [416.6ms elapsed, 0s remaining, 120.0 samples/s]
```

```
INFO:eta.core.utils: 100% |██████████| 50/50 [416.6ms elapsed, 0s remaining, 120.0 samples/s]
```

export(`export_dir=None, dataset_type=None, data_path=None, labels_path=None, export_media=None, rel_dir=None, dataset_exporter=None, label_field=None, frame_labels_field=None, overwrite=False, **kwargs`)

Exports the samples in the collection to disk.

Parameters:

- `export_dir` (None) – the directory to which to export the samples in format `dataset_type`. This parameter may be omitted if you have provided appropriate values for the `data_path` and/or `labels_path` parameters. Alternatively, this can also be an archive path with one of the following extensions: `.zip`, `.tar`, `.tar.gz`, `.tgz`, `.tar.bz`, `.tbz` If an archive path is specified, the export is performed in a directory of same name (minus extension) and then automatically archived and the directory then deleted
- `dataset_type` (None) – the `fiftyone.types.Dataset` type to write. If not specified, the default type for `label_field` is used
  - `COCODetectionDataset`
  - `CVATImageDataset`
  - `YOLOv5Dataset`
  - .... (more available in [https://docs.voxel51.com/api/fiftyone.types.dataset\\_types.html](https://docs.voxel51.com/api/fiftyone.types.dataset_types.html))



# FiftyOne

## How it works

```
1 #Export detections in YOLO format
2 export_dir = "open_data"
3
4 for split in ["train", "val"]:
5     split_view = dataset.match_tags(split)
6     split_view.export(
7         export_dir=export_dir,
8         dataset_type=fo.types.YOLOv5Dataset,
9         split=split
10    )
```

