

Shanghai Tech Data

Contents

- `ShanghaiTech`
- `ShanghaiTechDataset`
- `ShanghaiTechTestClipsIndexer`
- `ShanghaiTechTrainClipsIndexer`
- `make_shanghaitech_dataset()`

ShanghaiTech Campus Dataset.

Description:

This module contains PyTorch Dataset and PyTorch Lightning DataModule for the ShanghaiTech Campus dataset. If the dataset is not on the file system, the DataModule class downloads and extracts the dataset and converts video files to a format that is readable by pyav.

License:

ShanghaiTech Campus Dataset is released under the BSD 2-Clause License.

Reference:

- W. Liu and W. Luo, D. Lian and S. Gao. "Future Frame Prediction for Anomaly Detection – A New Baseline." IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.

class

```
anomalib.data.video.shanghaitech.ShanghaiTech(root='./datasets/shanghaitech',
scene=1, clip_length_in_frames=2, frames_between_clips=1,
target_frame=VideoTargetFrame.LAST, task=TaskType.SEGMENTATION,
image_size=None, transform=None, train_transform=None, eval_transform=None,
train_batch_size=32, eval_batch_size=32, num_workers=8,
val_split_mode=ValSplitMode.SAME_AS_TEST, val_split_ratio=0.5, seed=None)
```

Bases: `AnomalibVideoDataModule`

ShanghaiTech DataModule class.

Parameters:

- **root** (*Path | str*) – Path to the root of the dataset
- **scene** (*int*) – Index of the dataset scene (category) in range [1, 13]
- **clip_length_in_frames** (*int, optional*) – Number of video frames in each clip.
- **frames_between_clips** (*int, optional*) – Number of frames between each consecutive video clip.
- **target_frame** ([VideoTargetFrame](#)) – Specifies the target frame in the video clip, used for ground truth retrieval
- **TaskType** (*task*) – Task type, 'classification', 'detection' or 'segmentation'
- **image_size** (*tuple[int, int], optional*) – Size to which input images should be resized. Defaults to `None`.
- **transform** (*Transform, optional*) – Transforms that should be applied to the input images. Defaults to `None`.
- **train_transform** (*Transform, optional*) – Transforms that should be applied to the input images during training. Defaults to `None`.
- **eval_transform** (*Transform, optional*) – Transforms that should be applied to the input images during evaluation. Defaults to `None`.
- **train_batch_size** (*int, optional*) – Training batch size. Defaults to 32.
- **eval_batch_size** (*int, optional*) – Test batch size. Defaults to 32.
- **num_workers** (*int, optional*) – Number of workers. Defaults to 8.
- **val_split_mode** ([ValSplitMode](#)) – Setting that determines how the validation subset is obtained.
- **val_split_ratio** (*float*) – Fraction of train or test images that will be reserved for validation.
- **seed** (*int | None, optional*) – Seed which may be set to a fixed value for reproducibility.

`prepare_data()`

Download the dataset and convert video files.

Return type:

None

```
class anomalib.data.video.shanghaitech.ShanghaiTechDataset(task, split,
root='./datasets/shanghaitech', scene=1, clip_length_in_frames=2,
frames_between_clips=1, target_frame=VideoTargetFrame.LAST, transform=None)
```

Bases: [AnomalibVideoDataset](#)

ShanghaiTech Dataset class.

Parameters:

- **task** (*TaskType*) – Task type, 'classification', 'detection' or 'segmentation'
- **split** ([Split](#)) – Split of the dataset, usually Split.TRAIN or Split.TEST
- **root** (*Path* | *str*) – Path to the root of the dataset
- **scene** (*int*) – Index of the dataset scene (category) in range [1, 13]
- **clip_length_in_frames** (*int*, *optional*) – Number of video frames in each clip.
- **frames_between_clips** (*int*, *optional*) – Number of frames between each consecutive video clip.
- **target_frame** ([VideoTargetFrame](#)) – Specifies the target frame in the video clip, used for ground truth retrieval.
- **transform** (*Transform*, *optional*) – Transforms that should be applied to the input images. Defaults to

None

.

class

```
anomalib.data.video.shanghaitech.ShanghaiTechTestClipsIndexer(video_paths,
mask_paths, clip_length_in_frames=2, frames_between_clips=1)
```

Bases: [ClipsIndexer](#)

Clips indexer for the test set of the ShanghaiTech Campus dataset.

The train and test subsets of the ShanghaiTech dataset use different file formats, so separate clips indexer implementations are needed.

get_clip(*idx*)

Get a subclip from a list of videos.

Parameters:

idx (*int*) – index of the subclip. Must be between 0 and `num_clips()`.

Return type:

`tuple` [`Tensor`, `Tensor`, `dict` [`str`, `Any`], `int`]

Returns:

video (`torch.Tensor`) audio (`torch.Tensor`) info (`Dict`) video_idx (`int`): index of the video in `video_paths`

`get_mask(idx)`

Retrieve the masks from the file system.

Return type:

`Tensor` | `None`

`class`

`anomalib.data.video.shanghaitech.ShanghaiTechTrainClipsIndexer(video_paths, mask_paths, clip_length_in_frames=2, frames_between_clips=1)`

Bases: `ClipsIndexer`

Clips indexer for ShanghaiTech dataset.

The train and test subsets of the ShanghaiTech dataset use different file formats, so separate clips indexer implementations are needed.

`get_mask(idx)`

No masks available for training set.

Return type:

`Tensor` | `None`

`anomalib.data.video.shanghaitech.make_shanghaitech_dataset(root, scene, split=None)`

Create ShanghaiTech dataset by parsing the file structure.

The files are expected to follow the structure:

path/to/dataset/[training_videos|testing_videos]/video_filename.avi path/to/
ground_truth/mask_filename.mat

Parameters:

- **root** (*Path*) – Path to dataset
- **scene** (*int*) – Index of the dataset scene (category) in range [1, 13]
- **split** ([Split](#) | *str* | *None, optional*) – Dataset split (ie., either train or test). Defaults to *None*.

Example

The following example shows how to get testing samples from ShanghaiTech dataset:

```
>>> root = Path('./shanghaiTech')
>>> scene = 1
>>> samples = make_avenue_dataset(path, scene, split='test')
>>> samples.head()
```

	root	image_path	split	mask_path
0	shanghaitech	shanghaitech/testing/frames/01_0014	test	shanghaitech/testing/masks/01_0014
1	shanghaitech	shanghaitech/testing/frames/01_0015	test	shanghaitech/testing/masks/01_0015
...				

Returns:

an output dataframe containing samples for the requested split (ie., train or test)

Return type:

DataFrame

< [Previous](#)
[Avenue Data](#)

[UCSD Data](#) > [Next](#)