TVFace Dataset Utilities Documentation

This document provides detailed usage instructions and API reference for the first two modules:

- **tvface_dataset.py**: Loading TVFace images and annotation parsing.
- **compute_statistics.py**: Computing demographic and head-pose statistics.

---

1. tvface_dataset.py

```
from tvface_dataset import TVFaceDataset
```

Overview

`TVFaceDataset` is a custom PyTorch `Dataset` class that:

- Reads images from a specified directory.
- Parses face-related annotations (age, gender, race, expression, pose).
- Returns each sample as a dictionary containing both the image and extracted metadata.

Constructor

```
TVFaceDataset(img_dir: str, annotation_path: str, transform=None)
```

Parameter	Type	Description
`img_dir`	`str`	Path to directory containing `.jpg` images.
`annotation_path`	`str`	Path to the JSON file with `labels` entries for each image.
`transform`	callable	Optional image transform (e.g., `torchvision.transforms.ToTensor()`). May be `None`.

Attributes

- `self.ids` (`List[str]`): List of image IDs (filenames without extension).
- `self.annotations` (`Dict`): Loaded JSON mapping each `img_id` to annotation data.

`__len__`

```
def __len__(self) -> int:
```

- Returns the total number of samples (images) available.

`__getitem__`

```
def __getitem__(self, idx: int) -> dict:
```

Key	Type	Description
`image`	`PIL.Image`	The RGB image, optionally transformed.
`label`	`int`	Face label ID from annotation.
`mask`	`float`	Mask score (if present).
`age`	`str`	Top-probability age bracket (e.g., `'60-69'`).
`gender`	`str`	Top-probability gender (`'Male'` or `'Female'`).
`race`	`str`	Top-probability race (e.g., `'White'`, `'Middle Eastern'`).

Key	Type	Description
`expression`	`str`	Top-probability facial expression (e.g., `'sad'`, `'angry'`).
`pose`	`dict`	Pose angles: keys `['yaw', 'pitch', 'roll']`, each a `float`.
`age_probs`	`dict`	Full distribution of age probabilities.
`gender_probs`	`dict`	Full gender probability distribution.
`race_probs`	`dict`	Full race probability distribution.
`expr_probs`	`dict`	Full expression probability distribution.

Example Usage

```
from torchvision.transforms import ToTensor
from tvface_dataset import TVFaceDataset

dataset = TVFaceDataset(
    img_dir='tvface',
    annotation_path='annotation.json',
    transform=ToTensor()
)
print(len(dataset))              # e.g., 2_600_000
sample = dataset[0]
print(sample['age'], sample['gender'])
```

---

2. compute_statistics.py

```
from compute_statistics import compute_statistics
```

Overview

This script provides a function to aggregate demographic and pose statistics over all samples in a `TVFaceDataset` instance.

`compute_statistics`

```
def compute_statistics(dataset) -> dict:
```

Parameter	Type	Description
`dataset`	`TVFaceDataset` instance	Initialized dataset to compute statistics on.

Returns

A dictionary with the following keys:

- `age_distribution`: `Counter` mapping each age bracket to its count.
- `gender_distribution`: `Counter` of `'Male'` / `'Female'` counts.
- `race_distribution`: `Counter` of race counts.
- `expression_distribution`: `Counter` of expression counts.
- `pose_statistics`: `dict` with keys `['yaw','pitch','roll']`, each mapping to a sub-dict:
  - `mean`: Average angle over dataset.
  - `std`: Standard deviation of the angle.

Example CLI Usage

```
python compute_statistics.py
```

By default, the script:

1. Loads `tvface_dataset.py` with `ToTensor()` transforms.
2. Calls `compute_statistics`.
3. Prints human-readable distributions and pose means/stds.

Python Invocation

```
from tvface_dataset import TVFaceDataset
from compute_statistics import compute_statistics
from torchvision.transforms import ToTensor

dataset = TVFaceDataset('tvface', 'annotation.json', transform=ToTensor())
stats = compute_statistics(dataset)
print(stats['age_distribution'])
```