

MPVF: 4D Medical Image Inpainting by Multi-Pyramid Voxel Flows

Setup

1. To run this project, install it locally. `$ cd /path/to/MPVF $ pip install -r requirements.txt`
2. Start Visdom server We use **Visdom** to record the training procedure and visualization. `$ python3 -m visdom.server -port 1203`

Preprocess ACDC Dataset

1. Download the ACDC training and testing dataset from <https://acdc.creatis.insa-lyon.fr/#challenge/584e75606a3c77492fe91bba> and place at the `/path/to/MPVF/data` directory.
2. unzip dataset

```
$ unzip training.zip
$ unzip testing.zip
```

3. Preprocess `training` to `training_processed`

- Resample all data to $(160 * 160 * 10)$ and padding 0 to first and latest slices to $(160 * 160 * 12)$.

```
$ python3 preprocess.py --folder data/training --train
```

4. Preprocess `testing` to `testing_4`

- Resample all data to $(160 * 160 * 10)$ and padding 0 to first and latest slices to $(160 * 160 * 12)$.
- Pick the data which only has 5 or 9 or 13 or 17 time points between ED and ES.

```
$ python3 preprocess.py --folder data/testing --test
```

Training

Bilateral Voxel Flow (BVF)

```
$ python3 train_BVF.py --train_folder data/training_processed --test_folder data/testing_4 --model_name BVF -b 3 -l 1e-4 -e 1000 --train --port 1203
```

Pyramid Fusion (PyFu)

```
$ python3 train_PyFu.py --train_folder data/training_processed --test_folder data/testing_4 --flow_model_name BVF --model_name PyFu -b 3 -l 1e-4 -e 1000 --train --port 1203
```

Evaluation

Evaluate certain `t` for specify `-t=0.5` (or 0.25 or 0.75)

```
$ python3 evaluate_t.py --test_folder data/testing_4 --flow_model_name BVF --fusion_model_name PyFu --port 1203 -t 0.5
```

Generate arbitrary time point

Generate certain t for specify `-t=0.6` $t \in (0, 1)$

```
$ python3 generate_t.py --test_folder data/testing_4 --flow_model_name BVF --fusion_model_name PyFu --port 1203 --t 0.6
```

Note

- All the output will be visualized in Visdom