VSR-SIM: Spatio-temporal Vision Transformer for Super-resolution Microscopy

Layout of repository

- Powershell script for video dataset sampling:
  - `scripts/sample_documentary_videos.ps1`
- Python code for image formation model:
  - `scripts/im_form_model/SIMulator.py`
- Data generation script:
  - `scripts/datagen_pipeline.py`
- Model architecture based on Pytorch:
  - `basicsr/archs/vsr-sim_arch.py`
- Training code:
  - `basicsr/train.py`
- Inference code for testing:
  - `inference/inference_options.py`
- RBPN code base based on official implementation:
  - `RBPN-PyTorch`

Video sampling

Given a collection of .mp4 and .mkv video containers, we use the FFMPEG library to sample the collection with a time interval of 5 seconds between sequences. The script is launched using Powershell with

```
pwsh scripts/sample_documentary_videos.ps1
```

Data generation

The image formation pipeline can be used as follows

```
python datagen_pipeline.py --root TRAINING_DATA_DIRECTORY \
    --sourceimages_path SAMPLED_IMAGE_SEQUENCE_DIRECTORY --nrep 1\
    --datagen_workers 10 --imageSize 512  --nch_in 9 --nch_out 1\
    --ntrain 100000 --ntest 0 --scale 2 --nepoch 100 --scheduler 20,0.5\
    --norm minmax --workers 6 --dataonly --NoiseLevel 8 \
    --NoiseLevelRandFac 8 --Nangle 3 --Nshift 3 --phaseErrorFac 0.05 \
    --alphaErrorFac 0.05 --seqSIM --ext imagefolder
```

Training

To train a model with the VSR-SIM architecture using options specified in an associated options file, run the following

```
PYTHONPATH="./:${PYTHONPATH}" python basicsr/train.py \
    -opt options/train/VSR-SIM/VSR-SIM.yml
```

Inference on test set

Inference on a test set can be done with

```
PYTHONPATH="./:${PYTHONPATH}" python inference/inference_options.py\
    --task simrec --model_path experiments/VSR-SIM/models/net_g.pth\
    --scale 2 --input testdir/inputs --output testdir/outputs/VSR-SIM \
    -opt options/train/VSR-SIM/VSR-SIM.yml
```