

VSFA

- conda create -n pip python=3.9
- conda activate reproducibleresearch
- git clone <https://github.com/lidq92/VSFA.git>
- cd VSFA-master
- pip install torch torchvision torchaudio --index-url <https://download.pytorch.org/whl/cu118>
- pip install sk-video==1.1.10
- pip install h5py==3.11.0
- pip install tensorboardX==2.6.2.2
- Go to anaconda3\envs\reproducibleresearch\Lib\site-packages\skvideo\io\ffmpeg.py and replace every occurrence of np.float and np.int by float and int respectively.
- Go to root/VSFA.py and replace every occurrence of np.int by int.
- python test_demo.py --video_path=test.mp4
- Download datasets from: <http://database.mmsp-kn.de/konvid-1k-database.html>, <https://zenodo.org/record/2646315#.X6OmVC-1H3Q>, <http://live.ece.utexas.edu/research/incaptureDatabase/index.html>
- Open CNNfeatures.py and replace the path in videos_dir by your dataset's path.
- Run in terminal the command "set CUDA_VISIBLE_DEVICES=1"
- If your ds is KONVID, run `python CNNfeatures.py --database=KoNViD-1k --frame_batch_size=64`
- If your ds is CVD2014, run `python CNNfeatures.py --database=CVD2014 --frame_batch_size=32`
- If your ds is LIVE-Qualcomm, run `python CNNfeatures.py --database=LIVE-Qualcomm --frame_batch_size=32`
- After that, run `python VSFA.py --database=CVD2014 --exp_id=0` if you used CVD2014 dataset

MDTVSFA

- git clone <https://github.com/lidq92/MDTVSFA.git>
- conda create -n pip python=3.9
- conda activate reproducibleresearch
- cd MDTVVSFA-master
- pip install torch torchvision torchaudio --index-url <https://download.pytorch.org/whl/cu118>
- pip install sk-video==1.1.10
- pip install h5py==3.11.0
- pip install tensorboardX==2.6.2.2
- Go to anaconda3\envs\reproducibleresearch\Lib\site-packages\skvideo\io\ffmpeg.py and replace every occurrence of np.float and np.int by float and int respectively.
- Go to root/VSFA.py and replace every occurrence of np.int by int.
- python test_demo.py --video_path=test.mp4

- Download datasets from: <http://database.mmsp-kn.de/konvid-1k-database.html>, <https://zenodo.org/record/2646315#.X6OmVC-1H3Q>, <http://live.ece.utexas.edu/research/incaptureDatabase/index.html>
- Place the datasets on the root folder.
- Run in terminal the command “set CUDA_VISIBLE_DEVICES=1”
- If your ds is KONVID, run `python CNNfeatures.py --database=KoNViD-1k --frame_batch_size=64`
- If your ds is CVD2014, run `python CNNfeatures.py --database=CVD2014 --frame_batch_size=32`
- If your ds is LIVE-Qualcomm, run `python CNNfeatures.py --database=LIVE-Qualcomm --frame_batch_size=32`
- After that training, cross evaluate by:
 - Replace every `np.int` instance by `int`.
 - `pip install pytorch-ignite==0.5.0.post2`
 - executing: `python cross_dataset_evaluation.py --model MDTVSA --feature_extractor ResNet-50 --trained_model_file path/to/your/model.pt`
- The rest are class files.

nr-vqa-consumervideo

- git clone <https://github.com/jarikorhonen/nr-vqa-consumervideo.git>
- `cd nr-vqa-consumervideo`
- Open `compute_features_example.m`
- Mod the line 18 `yuv_path = '.'` with the path to your yuv files.
- Run it by pressing the green play button. That will save `LIVE_features.csv` file
- `conda create -n nr python=3.9`
- `conda activate nr`
- `pip install pandas==2.2.2`
- `pip install matplotlib==3.8.4`
- `pip install scipy==1.13.0`
- `pip install scikit_learn==1.4.2`
- `python nr_vqm_train_and_validate_example.py` (after you run the `compute_features_example.m`)

VIDEVAL

- git clone <https://github.com/vztu/VIDEVAL.git>
- `cd VIDEVAL`
- `conda create -n videval python=3.9`
- `conda activate videval`
- `pip install scikit-learn==1.4.2`
- `pip install pandas==2.2.2`
- Replace every instance of `np.float` by `float` and `np.int` by `int` in all `.py` files.
- Replace in `kfold_train_test_svr_all_combined_export_model.py` the line 33 by “import joblib”.
- Train with `python kfold_train_test_svr_all_combined_export_model.py`

- Run python demo_eval_BVQA_feats_all_combined.py
- Replace in file demo_eval_BVQA_feats_one_dataset.py line 68 “if color_only:” by “if color_only and data_name == 'YOUTUBE_UGC':”
- Run python demo_eval_BVQA_feats_one_dataset.py
- Open demo_compute_VIDEVAL_feats and run it!
- It'll display an error in the console and you must install the package by clicking the link “Statistics and Machine Learning Toolbox” that appears there as shown in the image.

```
Computing features for 1-th sequence: videos\5115335471.mp4
Video file size: 186624000 bytes (240 frames)
Computing frame features every second frame on frames 1..237
hist3 requires Statistics and Machine Learning Toolbox.
```