evaluation

September 12, 2017

1 Evaluation

```
In [ ]: get_ipython().magic('run util_notebook.py')
        get_ipython().magic('run -i geometry.py')
```

```
In [ ]: get_ipython().magic('run -i config.py')
        get_ipython().magic('run -i datasets.py')
        get_ipython().magic('run -i planes.py')
```

```
In [ ]: get_ipython().magic('run -i keypoint_projection.py')
        get_ipython().magic('run -i patches.py')
        get_ipython().magic('run -i detectors.py')
        get_ipython().magic('run -i matching.py')
```

```
In [ ]: # T-feat networks
        get_ipython().magic('run -i descriptor_tfeat.py')
```

```
In [ ]: # Networks trained on 7sc
        tfeat_7sc_flat = TFeatRunner(name='net_7sc_int', model_file= pp(DIR_OUT_MODELS, '7sc_fl
        tfeat_7sc_depth = TFeatRunner(name='net_7sc_depth', model_file= pp(DIR_OUT_MODELS, '7sc
        tfeat_7sc_norm = TFeatRunner(name='net_7sc_norm', model_file= pp(DIR_OUT_MODELS, '7sc_n
        tfeat_7sc_unw = TFeatRunner(name='net_7sc_unw', model_file= pp(DIR_OUT_MODELS, '7sc_unw
```

```
In [ ]: # Networks trained on synth
        tfeat_syn_flat = TFeatRunner(name='net_syn_int', model_file= pp(DIR_OUT_MODELS, 'synth_
        tfeat_syn_depth = TFeatRunner(name='net_syn_depth', model_file= pp(DIR_OUT_MODELS, 'syn
        tfeat_syn_norm = TFeatRunner(name='net_syn_norm', model_file= pp(DIR_OUT_MODELS, 'synth
        tfeat_syn_unw = TFeatRunner(name='net_syn_unw', model_file= pp(DIR_OUT_MODELS, 'synth_u
```

```
In [ ]: # Networks trained on arch
        tfeat_arch_flat = TFeatRunner(name='net_arch_int', model_file= pp(DIR_OUT_MODELS, 'arch
        tfeat_arch_depth = TFeatRunner(name='net_arch_depth', model_file= pp(DIR_OUT_MODELS, 'a
        tfeat_arch_norm = TFeatRunner(name='net_arch_norm', model_file= pp(DIR_OUT_MODELS, 'arc
        tfeat_arch_unw = TFeatRunner(name='net_arch_unw', model_file= pp(DIR_OUT_MODELS, 'arch_
```

```
In [ ]: # Descriptor networks to be evaluated:
        flat_nets_arch = [tfeat_arch_flat, tfeat_arch_depth, tfeat_arch_norm]
        flat_nets_7sc = [tfeat_7sc_flat, tfeat_7sc_depth, tfeat_7sc_norm]
```

```
flat_nets_syn = [tfeat_syn_flat, tfeat_syn_depth, tfeat_syn_norm]
flat_nets_all = flat_nets_arch + flat_nets_7sc + flat_nets_syn
unw_nets_all = [tfeat_arch_flat, tfeat_arch_unw, tfeat_7sc_flat, tfeat_7sc_unw, tfeat_s

def describe_all(fr):
        """
        Runs the whole pipeline of detecting describing and matching keypoints
        Then accumulates the accuracy metrics.

        patches: flat unwarp unwarp_det
        descriptors:
                sift
                nets
        """
        # planes
        frame_detect_planes(fr, b_normals=True)

        # copy the sequence to use with 2 detectors
        fr_flat_det = fr
        fr_unw_det = frame_clone(fr)

        # detect points on original frames
        #patches: original
        frame_detect_sift_flat(fr_flat_det, b_cut_patches=True)
        p_flat = fr_flat_det.patch_lists['flat']

        # patches: unwarp after detection
        frame_describe_unwarp(fr_flat_det)
        p_unw = fr_flat_det.patch_lists['unwarp']

        # patches: unwarp before detection
        frame_detect_sift_unwarp(fr_unw_det)
        p_unwdet = fr_unw_det.patch_lists['unwarp_det']

        # networks trained on original patches
        for net in flat_nets_all:
                net.describe_patch_list(p_flat)

        # networks trained on unwarped patches
        for net in unw_nets_all:
                net.describe_patch_list(p_unw)
                net.describe_patch_list(p_unwdet)

        return [fr_flat_det, fr_unw_det]
```

In [ ]:

In [ ]:

```
In [ ]: # DATASETS: synth
        dsets_synth = discover_synthetic_dsets(
                CFG_BASE,
                pp(DIR_DATA, 'synthetic')
        )
        dsets_synth_train = dsets_synth[:300]
        dsets_synth_test = dsets_synth[300:]
```

```
In [ ]: # DATASETS: 7SC
        dsets_7sc = discover_7scenes(CFG_BASE,
                scenes_train=['pumpkin', 'office', 'stairs', 'chess'],
                scenes_test=['fire', 'redkitchen', 'heads'],
        )
        dsets_7sc_train = dsets_7sc['train']
        dsets_7sc_test = dsets_7sc['test']
```

```
In [ ]: # DATASETS: arch
        dsets_arch_test = [
                DatasetSouthBuilding(CFG_BASE),
                DatasetPersonHall(CFG_BASE),
        ]
```

```
In [ ]: # Eval on: SYNTH SHORT
        evaluation_global(
                dsets_synth_test[:5],
                pp(DIR_OUT_EVAL, 'eval_synth_syn-7sc_short'),
                describe_all,
                fixed_seq_len=5,
                win_size=5,
                multiproc=False,
                multidet=True,
        )
        None
```

```
In [ ]: # Eval on: SYNTH LONG
        evaluation_global(
                dsets_synth_test[:50],
                pp(DIR_OUT_EVAL, 'eval_synth_all_long'),
                describe_all,
                fixed_seq_len=5,
                win_size=5,
                multiproc=False,
                multidet=True,
        )
        None
```

```
In [ ]: # Eval on: 7SC short
        evaluation_global(
                dsets_7sc_test[::5],
                pp(DIR_OUT_EVAL, 'eval_7sc_arch-7sc_short'),
                describe_arch,
                (1, 15),
                win_size=6,
                multiproc=False,
                multidet=True,
                dset_max_frame_count=60,
        )
        None
```

```
In [ ]: # Eval on: 7SC long
        evaluation_global(
                dsets_7sc_test,
                pp(DIR_OUT_EVAL, 'eval_7sc_syn-7sc_long'),
                describe_all,
                (1, 15),
                win_size=6,
                multiproc=False,
                multidet=True,
        )
```

```
In [ ]: # Eval on: arch short
        evaluation_global(
                dsets_arch_test[::-1],
                pp(DIR_OUT_EVAL, 'eval_arch_arch-7sc_short'),
                describe_arch,
                (1, 3),
                win_size=5,
                multiproc=False,
                multidet=True,
                dset_max_frame_count=20,
        )
```

```
In [ ]: # Eval on: arch long
        evaluation_global(
                dsets_arch_test[:1],
                pp(DIR_OUT_EVAL, 'eval_arch_all_long'),
                describe_all,
                (1, 3),
                win_size=5,
                multiproc=False,
                multidet=True,
        )
```