

plot_psnr_nb_boat_Run

June 3, 2019

```
In [1]: %matplotlib inline
# N=256, L=8, J=8, dk=1, dl=L/2, bump steerable
N=256
# obtained with compute_coef_ps2.py
nb_dict={
    0:5890,
    1:17426,
    2:27314,
    3:35714,
    4:42434,
    5:47474,
}
RUNs = 10
pt_dict = dict()
import os.path
for dj in nb_dict.keys():
    pt_dict[dj]=[]
    for runid in range(1,RUNs+1):
        ptfile = './boat/Run' + str(runid) + '/' + \
            'test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_' + 'dj' +str(dj) + '.pt'
        print('check',ptfile)
        assert os.path.isfile(ptfile)
        pt_dict[dj].append(ptfile)
csvfile = 'plot_pnsr_nb_boat_Run.csv'

check ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run3/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run4/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run5/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run6/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run7/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run8/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run9/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run10/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt
check ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt
check ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt
```

[illegible]

```

In [3]: import scipy.io as sio
        droot0='../kymatio_wph_data/'
        datalabel='demo_boat1_N' + str(N)
        data = sio.loadmat(droot0 + datalabel)
        im = data['imgs']

In [4]: import numpy as np
        def minMse_match(img1,img2):
            assert len(img1.shape)==2, 'input img should be dim 2'
            N=img1.shape[0]
            M=img1.shape[1]
            assert(img2.shape[0]==N and img2.shape[1]==M)
            minMse = -1
            minC0 = 0
            minC1 = 0
            for cn in range(N):
                oimg1 = np.roll(img1,cn,axis=0)
                for cm in range(M):
                    ooimg1 = np.roll(oimg1,cm,axis=1)
                    Mse = np.linalg.norm(ooimg1-img2,'fro')**2
                    if minMse<0:
                        minMse = Mse
                    elif minMse>Mse:
                        minMse = Mse
                        minC0 = cn
                        minC1 = cm
                    #print('minMSE',minMse,'cn',cn,'cm',cm)
            return minMse, minC0, minC1

In [5]: import torch
        psnr_dict = {}
        for dj in pt_dict.keys():
            psnr_dict[dj] = []
            print(dj)
            for i, ptfile in enumerate(pt_dict[dj]):
                saved_result = torch.load(ptfile)
                print('normalized_loss',saved_result['normalized_loss'])
                im_opt = saved_result['tensor_opt'].numpy()
                minMSE,minC0,minC1 = minMse_match(im_opt[0,0,:,:],im)
                psnr_dict[dj].append( -10*np.log10(minMSE/(N*N)) )
                print('eval', ptfile, 'PSNR',-10*np.log10(minMSE/(N*N)))

        print("PSNR eval complete")

```

0

normalized_loss 4.170789907220751e-15

eval ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 21.701828231511463

normalized_loss 4.157084549660795e-15
eval ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 20.81940882744247
normalized_loss 6.75214032526128e-15
eval ./boat/Run3/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 20.89047727419252
normalized_loss 7.182567787822336e-15
eval ./boat/Run4/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 20.70653035581725
normalized_loss 3.031435880984645e-16
eval ./boat/Run5/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 22.17776174793866
normalized_loss 2.485409459040966e-15
eval ./boat/Run6/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 21.16022001503572
normalized_loss 6.983390449022409e-16
eval ./boat/Run7/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 22.469446692657826
normalized_loss 8.9454706539982e-16
eval ./boat/Run8/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 21.76321353466499
normalized_loss 5.517682438949123e-15
eval ./boat/Run9/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 21.115285605602473
normalized_loss 5.153030542714987e-16
eval ./boat/Run10/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj0.pt PSNR 21.695163438429176
1
normalized_loss 5.4789745718153424e-17
eval ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 30.31069887488038
normalized_loss 7.441646516781475e-17
eval ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 29.96067910744887
normalized_loss 5.320530362951104e-17
eval ./boat/Run3/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 29.97179127292474
normalized_loss 5.577077786256268e-17
eval ./boat/Run4/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 30.606954244741836
normalized_loss 9.12805546704476e-17
eval ./boat/Run5/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 29.93763064416799
normalized_loss 5.660294846165926e-17
eval ./boat/Run6/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 30.516087599211662
normalized_loss 3.874141896176297e-17
eval ./boat/Run7/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 30.729163990762537
normalized_loss 6.20832054210041e-17
eval ./boat/Run8/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 30.07395357920299
normalized_loss 5.952012429588649e-17
eval ./boat/Run9/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 30.771135093822988
normalized_loss 5.3996052429283733e-17
eval ./boat/Run10/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj1.pt PSNR 29.72121343484256
2
normalized_loss 3.4012202831945613e-17
eval ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 34.932879428147245
normalized_loss 4.0496630049347004e-17
eval ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 34.7518565222327
normalized_loss 3.855145678244298e-17
eval ./boat/Run3/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 32.99159248506274
normalized_loss 3.548943823261652e-17
eval ./boat/Run4/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 33.80928721189635

normalized_loss 2.467508295467269e-17
eval ./boat/Run5/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 33.861810545292116
normalized_loss 4.1612727841311427e-17
eval ./boat/Run6/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 33.74780276082441
normalized_loss 4.2233159547322427e-17
eval ./boat/Run7/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 34.72742299055348
normalized_loss 2.702296058032516e-17
eval ./boat/Run8/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 35.38054809871916
normalized_loss 5.008924972571549e-17
eval ./boat/Run9/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 32.36809995038132
normalized_loss 2.943930894616642e-17
eval ./boat/Run10/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj2.pt PSNR 33.24962849579629
3
normalized_loss 1.1086582674124657e-17
eval ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 37.21955125271084
normalized_loss 1.0897053925873478e-17
eval ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 37.19090975589714
normalized_loss 1.403695932822302e-17
eval ./boat/Run3/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 36.55130148439704
normalized_loss 1.3475464299972372e-17
eval ./boat/Run4/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 34.41712995397998
normalized_loss 1.6059294694059643e-17
eval ./boat/Run5/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 33.71818970316744
normalized_loss 1.1164130597762777e-17
eval ./boat/Run6/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 35.43433996913744
normalized_loss 1.1100257779617095e-17
eval ./boat/Run7/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 36.68051840041363
normalized_loss 1.2837116969421913e-17
eval ./boat/Run8/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 37.0928801626609
normalized_loss 2.3503005763814144e-17
eval ./boat/Run9/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 35.021401537194635
normalized_loss 1.0862267885158871e-17
eval ./boat/Run10/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj3.pt PSNR 38.14211871443517
4
normalized_loss 1.1176385328326432e-17
eval ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 38.68231399400278
normalized_loss 3.391903646843275e-17
eval ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 34.29348963854924
normalized_loss 6.8802407326984395e-18
eval ./boat/Run3/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 40.96740382781428
normalized_loss 1.038265224906354e-17
eval ./boat/Run4/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 40.53667033216934
normalized_loss 7.767221177346073e-18
eval ./boat/Run5/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 41.1134750120777
normalized_loss 1.0975841746585502e-17
eval ./boat/Run6/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 37.66112430863199
normalized_loss 1.249243410939016e-17
eval ./boat/Run7/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 39.692340097274204

```

normalized_loss 7.425238379710209e-18
eval ./boat/Run8/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 38.83242061170827
normalized_loss 1.4302212036909623e-17
eval ./boat/Run9/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 38.830339399388436
normalized_loss 3.556492913503462e-17
eval ./boat/Run10/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj4.pt PSNR 33.138814880500995
5
normalized_loss 3.38384836595651e-17
eval ./boat/Run1/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 33.453165290051246
normalized_loss 1.9121807781630194e-17
eval ./boat/Run2/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 37.41681867314143
normalized_loss 1.1741423833200315e-17
eval ./boat/Run3/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 39.005221867499195
normalized_loss 5.293236426950898e-18
eval ./boat/Run4/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 42.05933233374882
normalized_loss 5.02012653669226e-18
eval ./boat/Run5/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 42.4652477753656
normalized_loss 6.280392028656933e-18
eval ./boat/Run6/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 40.719672349496754
normalized_loss 5.329833641098958e-18
eval ./boat/Run7/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 41.232002690085146
normalized_loss 5.14617042313148e-18
eval ./boat/Run8/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 42.93658396650707
normalized_loss 7.553712322305728e-18
eval ./boat/Run9/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 40.012663565363965
normalized_loss 6.888763692813882e-18
eval ./boat/Run10/test_rec_bump_chunkid_lbfgs_gpu_N256_ps2par_dn0_dj5.pt PSNR 41.747817419614236
PNSR eval complete

```

```

In [6]: # save to csv file
import csv

myData = [["ptfile", "dj", "M", "PSNR"]]

for dj in psnr_dict.keys():
    for i, psnr in enumerate(psnr_dict[dj]):
        M = nb_dict[dj]
        myData.append([pt_dict[dj][i], dj, M, psnr])

myFile = open(csvfile, 'w')
with myFile:
    writer = csv.writer(myFile)
    writer.writerows(myData)
print("Writing complete")

```

Writing complete

```
In [ ]: # LOAD CSV
```

```
In [2]: import pandas as pd
import numpy as np
df = pd.read_csv(csvfile)
psnr = np.array(df['PSNR'].astype(float))
M = np.array(df['M'].astype(int))
dj = np.array(df['dj'].astype(int))
# select smallest PSNR for each dj
lnb = []
lpsnr=[]
Nd=float(N*N)
for delta_j in range(6):
    Mj = np.min(M[dj==delta_j])
    lnb.append(np.log10(Mj/Nd))
    lpsnr.append(np.max(psnr[dj==delta_j]))
lnb=np.array(lnb)
lpsnr=np.array(lpsnr)
print(lnb, lpsnr)
```

```
[-1.04636464 -0.57528222 -0.38009463 -0.26364144 -0.18876596 -0.14002411] [22.46944669 30.771135
```

```
In [3]: import matplotlib
#matplotlib.style.use('ggplot')
import matplotlib.pyplot as plt
#print matplotlib.__version__
do_color = False
blue = 'b' if do_color else 'k'
```

```
In [4]: plt.figure()
plt.axhline(y=10, Color='lightgray', Linestyle=":", Linewidth=1)
plt.axhline(y=20, Color='lightgray', Linestyle=":", Linewidth=1)
plt.axhline(y=30, Color='lightgray', Linestyle=":", Linewidth=1)
plt.axhline(y=40, Color='lightgray', Linestyle=":", Linewidth=1)
plt.axhline(y=50, Color='lightgray', Linestyle=":", Linewidth=1)
plt.plot(lnb, lpsnr) #, Linewidth=3, Color=blue)

plt.xticks([-1, 0],fontSize=20)
plt.yticks([0, 10, 20, 30, 40, 50, 60],fontSize=20)
plt.xlim(xmin=-1.25, xmax=0)
plt.ylim(ymin=10, ymax=60)

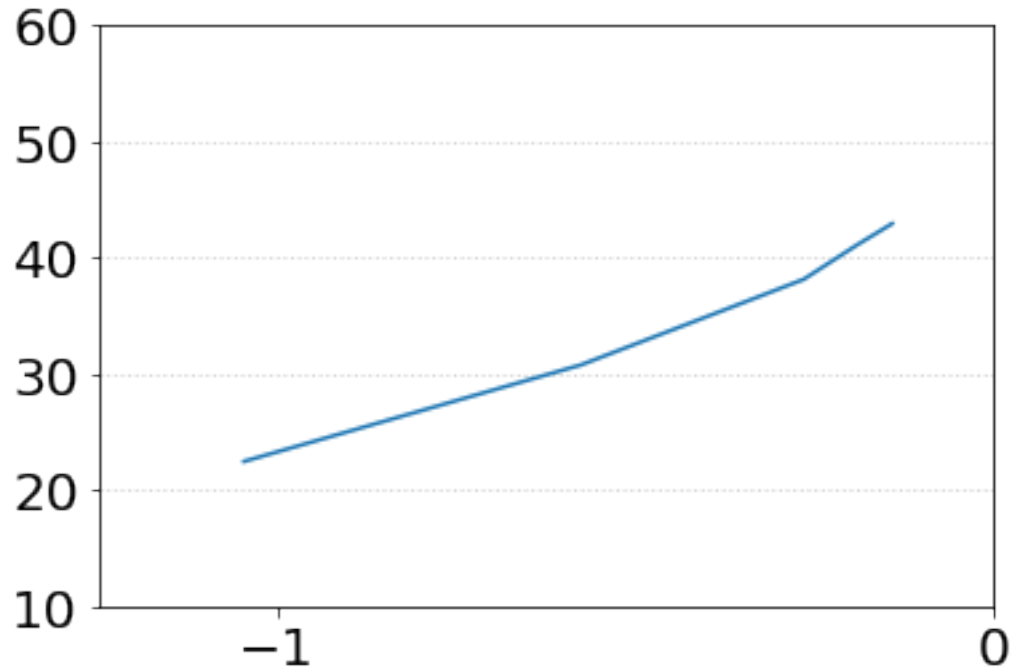
plt.savefig('fig8_boat_psnr.pdf')
```

```
/home/zsx/anaconda3/envs/py36/lib/python3.6/site-packages/matplotlib/axes/_base.py:3215: MatplotlibDeprecationWarning:
The `xmin` argument was deprecated in Matplotlib 3.0 and will be removed in 3.2. Use `left` instead.
alternative='`left`', obj_type='argument')
/home/zsx/anaconda3/envs/py36/lib/python3.6/site-packages/matplotlib/axes/_base.py:3221: MatplotlibDeprecationWarning:
```

The ``xmax`` argument was deprecated in Matplotlib 3.0 and will be removed in 3.2. Use ``right`` instead.
 alternative=``right``', obj_type='argument')

/home/zsx/anaconda3/envs/py36/lib/python3.6/site-packages/matplotlib/axes/_base.py:3604: MatplotlibDeprecationWarning: The ``ymin`` argument was deprecated in Matplotlib 3.0 and will be removed in 3.2. Use ``bottom`` instead.
 alternative=``bottom``', obj_type='argument')

/home/zsx/anaconda3/envs/py36/lib/python3.6/site-packages/matplotlib/axes/_base.py:3610: MatplotlibDeprecationWarning: The ``ymax`` argument was deprecated in Matplotlib 3.0 and will be removed in 3.2. Use ``top`` instead.
 alternative=``top``', obj_type='argument')



```
In [5]: import scipy.stats as sps
        slope, intercept, r_value, p_value, std_err = sps.linregress(lnb, lpsnr)
        print(slope/20)
```

1.1036847961114882