0) Measure **BoW** on Holidays **Done**

sift=128, dictionary=64
norm2 (8 tryals) ~~0.3009±0.002~~
norm1 (8 tryals) ~~0.2925±0.002~~
term frequency (3 tryals) ~~0.2710±0.005~~
term frequency + norm2 (3 tryals) ~~0.2808±0.0038~~

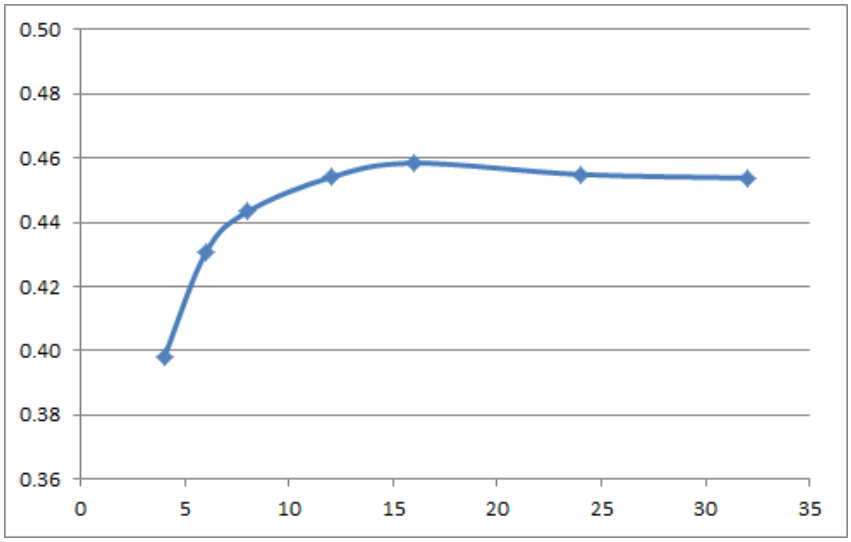
norm2 (10 tryals) 0.6129/0.6167+0.0024/0.6212
term frequency + norm2 (10 tryals) 0.6293/0.6343+0.0024/0.6372

rootSifts=128, dictionary=64
term frequency (10 tryals) 0.6264/0.6309±0.0025/0.6350
term frequency + norm2 (10 tryals) 0.6293/**0.6343**±0.0024/0.6372

sift=128, dictionary=256
term frequency + norm2 (10 tryals) 0.6293/0.6343+0.0024/0.6372
sift=128, dictionary=1024
term frequency + norm2 - launch failed, cause is yet to be uncovered.

0) Measure **color histograms** on Holidays **Done**

Histogram size	4	6	8	12	16	24	32
~~mAP~~	~~0.3049~~	~~0.3274~~	~~0.3381~~	~~0.3418~~	~~0.3395~~	~~0.3378~~	~~0.3356~~
mAP	0.3980	0.4306	0.4433	0.4540	0.4584	0.4549	0.4538



1) Measure **VLAD** on Holidays **Done**
sift=128
dictionary=64

using soft assignment [w=1/(d^2+1)]
tested 100 clusterizations:
mAP = ~~0.3837/0.4088±0.0119/0.4387~~ (=min/avg±std/max)

using hard assignment [w=0/1]
tested 100 clusterizations:
mAP = ~~0.3837/0.4088±0.0119/0.4387~~ (the same)
rootSifts, tested 7 clusterizations:
mAP = ~~0.3684/0.3709±0.002/0.3743~~
rootSifts (shift -127), tested 9 clusterizations:
mAP = ~~0.3723/0.3921±0.0105/0.4124~~

using hard assignment [w=0/1]
rootVLAD
mAP = ~~0.4279~~ 0.7166 (preliminary)
VLAD+norm2
mAP = ~~0.4088~~ (preliminary)
rootVLAD+component norm2 (options 'SquareRoot' and 'NormalizeComponents')
mAP = ~~0.2950~~ (preliminary)
VLAD 'NormalizeComponents'
mAP = ~~0.2868~~ (preliminary)

rootSifts, rootVLAD
mAP = 0.7166 (preliminary)

2) Implement **FV**, measure on Holidays **Skipped**

3) **Multiple codebook VLAD +PCA Partially**

- Created 100 different clusterization
- Created function to measure distance between them and measured 100x100
- Sergey and me implemented bunch of algorithms to select the best sample/selection.
  We used two metrics to score samples:
  1. sum(d(i,j)) -> max
  2. min(d(i,j)) -> max
  And several approaches to select the best sample:
  1. sort clusterizations by distance from the rest of examples (sum_d, min_d) and select clusterizations that are the farest
  2. 1st clusterization = the farest from the whole set, 2nd - the farest from the 1st, and so on - next is the farest(sum_d, min_d) from previously selected

clusterizations.

3. brute force = exhausting search
4. random search - pick the best from several (about 1M) random samples.

So far looks like 2nd metric and 2nd approach are the best.

4) vl_sift vs vl_phow **Skipped**

5) **rootSIFT** **Done**
(6.1 here:
http://www.robots.ox.ac.uk/~vgg/publications/2012/Arandjelovic12/arandjelovic12.pdf)

6) **Intranormalization** **Done**

http://lear.inrialpes.fr/pubs/2010/JDSP10/jegou_compactimagerepresentation.pdf
http://www.axes-project.eu/wp-content/uploads/2013/08/arandjelovic13.pdf
http://hal.inria.fr/docs/00/86/46/84/PDF/iccv13_tolias.pdf

7) **Product quantization (16x16) on PCA VLAD RootSITFs** **Done**

Exhaustive search (1M + Holidays): (measuring expense 0.599 s)

	mAP	time per query, sec
no PQ (pca baseline)	0.5134	3.701
no PQ (pca, dot product dist)	0.3887	0.429
symmetric	0.3597	0.736
asymmetric	0.3953	0.852

BoW, Phow, VLAD, FV
VLAD adapt+innorm
MultiVLAD, MultivocVLAD
Discriminative Query Expansion + Spatial Augmentation