

# CC\_homework2 Report

資工碩一 r09922055 陳柏妤

## 1. Color feature

Regional color feature performs better than global color feature, following is the result.

# of grid	1	4	9	16	25
Max MAP	0.7315	0.7727	0.8682	0.8699	0.8741
Mean MAP	0.2853	0.2933	0.3230	0.3220	0.3200

Slice image into 9 smaller grids, the weight of each grid is related to its distance to the center (i.e. The closer, the bigger), since the information in the center is more important than that of outer. Without this method, the MAP will reduce about 2%. Then, compute the following features:

- hsv color histogram  
768 dimensions(bin) =  $256H + 256S + 256V$
- hsv color space  
162 dimensions(bin) =  $18H * 3S * 3V$
- bgr color space  
512 dimensions(bin) =  $8R * 8G * 8B$

Concat these 3 features of each grid, so the image can be represented by 23072-dimensional vector ( $23072 = (768+162+512) * 9$ )

## 2. Edge feature

By observing the images in dataset, I think the edge feature may be more powerful than texture feature (since there seems no special texture in these image), and select PHOG as implemented approach.

PHOG is representing histogram of gradient (HOG) in spatial pyramid, and there are four parameter used in my implementation.

- scale  
To construct the pyramid with different resolution, resize the image by ratio of scale parameter. Do not make much difference, but scale=2 performs better.
- kernel  
Use Sobel kernel to compute HOG, the kernel size can be  $1*1, 3*3, 5*5\dots$ . Smallest kernel performs best.
- angel\_slicing  
Number of bins in HOG is controlled by “angel\_slicing” parameter. I tried parameters of 20, 24, 36, the larger parameter performs better.
- edge\_slicing  
It is for local edge feature. Slicing images in pyramid into smaller girds make the performance better, and the image in higher resolution is sliced into more grid.

HOG is weighted by magnitude, concat the HOGs of grids in each resolution level as edge feature.

### 3. Local feature

- sift stage : training

In the training process, I gather features from 700 image. The feature is extract by SIFT, and each image will be represented by n 128-dimensional descriptors (in my experiment n=130 performs the best), and we have about  $130 \times 700$  128-dimensional descriptors.

- k-means

Then cluster the  $130 \times 700$  descriptors into 80 class by k-means.

- sift stage : testing

In the prediction process, use sift to gather features from 700 images again. Each image will be represented by m 128-dimensional descriptors (in my experiment m=280 performs the best), and classify these descriptors into 80 class by kmeans.predict().

Finally, every image can be represented by 80-bin histogram, which is the local feature.

### 4. Fusion feature

I apply the technique of ensemble learning. For each image  $i$ , rank other 699 images,  $j$ , by color feature, color-and-edge feature and color-and-local feature respectively.

- color feature : only use color feature
- color-and-edge feature : color feature + edge feature weighted by 3
- color-and-local feature : color feature + local feature

After the ranking process, we will get 3 sorted-699-dimensional ranking-list, each represents the similarity between  $i$  and other 699 images.

Then, for each image  $j$  we sum its orders (rank) in the 3 ranking-lists, and get 699-dimensional total ranking-list. The image with the highest total rank is considered to be the most related image to image  $i$ .

### 5. Similarity

- Bhattacharyya distance

MAP of color feature is better when similarity is sorted by Bhattacharyya distance.

- cosine similarity

MAP of edge&local feature is better when similarity is sorted by cosine similarity.

	Color	Edge	Local
MAP by Bhattacharyya distance	0.3230	0.054	0.1702
MAP by cosine similarity	0.2623	0.1918	0.2151

### 6. Result of MAP

	Color	Edge	Local	Fusion
Best two	Korean_snack : 0.8682 gge_snack : 0.7222	gge_snack : 0.7825 children_dress : 0.4139	korean_snack : 0.9584 gge_snack : 0.8709	korean_snack : 0.9210 gge_snack : 0.8835
Worst two	sweeping_robot : 0.0487 nba_jersey : 0.0371	blue_pillow : 0.0628 mouse : 0.0504	sweeping_robot : 0.0340 lollipop: 0.0317	sweeping_robot : 0.0515 nba_jersey : 0.0488
Mean	0.3230	0.1918	0.2151	0.3658

## 7. Some Result of Running

```
#####
evaluate color #####
color_featuresize : (700, 23072)
===== Color MAP from high to low =====
korean_snack : 0.8682248986294485
gge_snack : 0.7222368774298321
minnie_dress : 0.6772891072500516
aloe_vera_gel : 0.6561327788721282
sprite : 0.6158781571365526
skirt : 0.5150003690425773
garment : 0.5120307285259846
hand_cream : 0.5027433614604135
orange : 0.46800239345810163
tennis_ball : 0.45827150187741826
cartoon_purse : 0.4480310180595735
women_clothes : 0.41221595841266157
goggles : 0.4097120194813074
lollipop : 0.3786350571634674
cup : 0.3257382082785237
minnie_shoes : 0.3168750031075753
baby_shoes : 0.31407963068376843
bottle : 0.2979916388028838
bracelet : 0.27260152184179076
men_clothes : 0.26121496388062093
drum : 0.24841748197713615
blue_pillow : 0.24362288466402626
children_dress : 0.2049939473802166
bicycle : 0.1880784448892786
overalls : 0.14698967838310134
ice_cream : 0.13935600581695823
leather_purse : 0.12249661467470256
glasses : 0.1149672987801939
mouse : 0.1077105581219832
chair : 0.08263511988252861
suitcase : 0.07589068328040033
clock : 0.0673281813474186
trousers : 0.056795226403437024
sweeping_robot : 0.04869723843575238
nba_jersey : 0.037055406793727416
mean: 0.3230134093292392
max: 0.8682248986294485
```

mean: 0.19175054734759203  
max: 0.7825110807317721

```
#####
evaluate edge #####
edge_featuresize : (700, 1332)
===== Edge MAP from high to low =====
korean_snack : 0.7825110807317721
children_dress : 0.41393643522786
bottle : 0.3612446668736604
korean_snack : 0.3406014138758565
aloe_vera_gel : 0.2889467183636509
goggles : 0.28864654072661927
cup : 0.2622591628438726
overalls : 0.25709300171869716
suitcase : 0.24297745777638072
trousers : 0.21301040259821273
minnie_shoes : 0.2125052032322606
garment : 0.2058339216689173
clock : 0.18270004105462592
minnie_dress : 0.17789476187748127
baby_shoes : 0.1778947311215798
leather_purse : 0.17206706559831798
nba_jersey : 0.16365125764141694
women_clothes : 0.15688831834077926
sprite : 0.15613329804733998
skirt : 0.15026332976704992
lollipop : 0.134883599348679
bicycle : 0.1294499523527522
chair : 0.1244095294797397
orange : 0.12146463426822351
cartoon_purse : 0.11785929347691315
sweeping_robot : 0.1142599487589346
glasses : 0.10696914046316171
men_clothes : 0.10352162265718172
hand_cream : 0.09805800715303747
tennis_ball : 0.09568315587498094
drum : 0.088318301774460974
bracelet : 0.08451877762837842
ice_cream : 0.0723755022694479
blue_pillow : 0.0627504638263609
mouse : 0.05044065973191515
mean: 0.2150614458991161
max: 0.9584305721243529
```

mean: 0.19175054734759203  
max: 0.7825110807317721

```
#####
evaluate local #####
local_featuresize : (700, 80)
===== Local MAP from high to low =====
korean_snack : 0.9584305721243529
gge_snack : 0.8708842515015565
children_dress : 0.45523089279042395
garment : 0.38632170880872113
minnie_dress : 0.3713020323874443
bracelet : 0.37000757734504075
women_clothes : 0.35143193772725156
blue_pillow : 0.3378126317928464
aloe_vera_gel : 0.30243291078442497
chair : 0.251569771652842
cartoon_purse : 0.22752369556311786
baby_shoes : 0.2230799788820073
men_clothes : 0.21668617274696161
skirt : 0.2121279468173938
hand_cream : 0.19454642220015615
ice_cream : 0.14636464418218942
minnie_shoes : 0.14340356045136227
cup : 0.14171274276872326
overalls : 0.13731866043761243
clock : 0.1308893887265256
bicycle : 0.13811010729652334
drum : 0.11508407426527026
orange : 0.10044434602983196
goggles : 0.09844854236070592
nba_jersey : 0.0971297889762525
bottle : 0.09233063301863663
sprite : 0.090410209566780506
glasses : 0.0798174838861114
trousers : 0.07069252963469858
mouse : 0.0631843404495197
leather_purse : 0.06568145761777415
suitcase : 0.04291455888128492
tennis_ball : 0.04114060615939019
sweeping_robot : 0.03402802493937461
lollipop : 0.031721799396595786
mean: 0.2150614458991161
max: 0.9584305721243529
```

mean: 0.2150614458991161  
max: 0.9584305721243529

```
#####
evaluate fusion #####
===== Fusion MAP from high to low =====
gge_snack : 0.9210122286974374
korean_snack : 0.8834600461019504
aloe_vera_gel : 0.77361464449301539
minnie_dress : 0.7963430449528314
sprite : 0.6880011199555545
garment : 0.552174055902865
goggles : 0.5511899078638929
skirt : 0.53007046695734842
hand_cream : 0.5282885691928636
orange : 0.5264227716741668
cartoon_purse : 0.4998515076409361
women_clothes : 0.4754836334524575
tennis_ball : 0.40591862400070144
cup : 0.4002771051472126
minnie_shoes : 0.39658473257478793
bottle : 0.3940792966831946
lollipop : 0.3789224411881751
baby_shoes : 0.376916699805629
drum : 0.3021202149284466
children_dress : 0.2969447617146387
bicycle : 0.2822316337598112
men_clothes : 0.281478386847035
bracelet : 0.2651608543915683
blue_pillow : 0.20774364119120978
overalls : 0.1870375452383134
ice_cream : 0.1500232836078142
leather_purse : 0.13881681748415537
suitcase : 0.12294482842777246
glasses : 0.11258458482980775
trousers : 0.10618856337084563
chair : 0.09427442129519283
clock : 0.08894002365477169
mouse : 0.0795266676856875
sweeping_robot : 0.05146447105097769
nba_jersey : 0.04879849590153205
mean: 0.3658082868010923
max: 0.9210122286974374
```

mean: 0.3658082868010923  
max: 0.9210122286974374

