Appendix A

Datasets.

A.1 Datasets

In this work we adopt several datasets previously used in visual recognition task such as image annotation, image retrieval, scene classification etc. In addition to the existing datasets, we introduce three new datasets — two datasets for the task of image retrieval and one for cross-modal retrieval. Next, we briefly discuss the salient properties of these datasets.

A.1.1 Natural Scene Categories (N8, N13, N15)

The Natural Scene Categories, is a collection of three datasets, viz. "LabelMe Natural Scenes", "Thirteen Natural Scenes" and "Fifteen Natural Scenes", where "LabelMe Natural Scenes" is a subset of "Thirteen Natural Scenes" which itself is a subset of the "Fifteen Natural Scenes" dataset.

LabelMe Natural Scenes (N8)

"LabelMe Natural Scenes" dataset, henceforth referred to as "Natural8" (N8), consists of 2688 images classified into eight classes viz "Coast", "Forest", "Highway", "Inside City", "Mountain", "Open Country", "Street", "Tall Building". This dataset was first proposed in [104] and has been later used in several scene classification literatures [114, 17, 80, 67] etc. Although the images are available with color, in this work as is commonly done we convert all the images to gray scale. The average size of each image is 250×250 pixels. N8 dataset is primarily used for scene classification task, where 100 images per class serve as the training set and the rest of the images as the test set. A.1 provides a detailed description of various classes.

Thirteen Natural Scenes (N13)

"Thirteen Natural Scenes" dataset here referred to as "Natural13 (N13)", was first proposed in [77] where five more scene categories, viz. "Bedroom", "Suburb", "Kitchen", "Livingroom", "Office", were added to the N8 dataset. N13 dataset has been used by several authors to evaluate scene classification systems

Table A.1: Summary of the Natural Scene datasets.

Natural8 (N8)			
Category	Training set	Test set	Total
Coast	100	260	360
Forest	100	228	328
Highway	100	160	260
Inside City	100	208	308
Mountain	100	274	374
Open Country	100	310	410
Street	100	192	292
Tall Building	100	256	356
total	800	1888	2688
Natural13 (N13) Additional Classes			
Bedroom	100	116	216
Suburb	100	141	241
Kitchen	100	110	210
Livingroom	100	189	289
Office	100	115	215
total	1300	2559	3859
Natural15 (N15) Additional Classes			
Store	100	215	315
Industrial	100	211	311
total	1500	2985	4485

[17, 114, 74, 6, 65]. A.1 provides a detailed description of various additional classes of the N13 dataset.

Fifteen Natural Scenes (N15)

“Fifteen Natural Scenes” dataset, here referred to as “Natural15” (N15) is currently one of the most popular dataset used for the evaluation of scene recognition systems. N15 dataset was first proposed in [74], where two more scene categories, viz. “Store”, “Industrial” were added to the N13 dataset. Thus, N15 dataset consists of fifteen classes of natural scenes where each class contains 200 to 400 images, of average size 270×250 pixels. In all the experiments using N15 dataset, 100 images per scene are used to learn the model, the remaining being used as test set. A.1 provides a detailed description of the additional classes in the N15 dataset.

A.1.2 UIUC Sports Dataset (S8)

UIUC Sports dataset, henceforth refered to as ”Sports8” (S8), consists of 1579 images classified into eight sports categories, viz. {“badminton”, “bocce”, “croquet”, “polo”, “rock climbing”, “rowing”, “sailing”, “snowboarding”}. It was first proposed in [79] for Latent Dirichlet Allocation based (LDA) based classification, and subsequently used by [167] to evaluate supervised-LDA. Each category has 137 to 250 images with an average size of over 1000 × 1000 pixels. For our experiments, the images were resized to a maximum of 256 pixels along the larger border. In all, there are 1579 images. In this work S8 dataset is used to evaluate scene classification systems. As in [79], 70 images per scene are used to learn the model, and 60 images are used as test set. A.2 provides a detailed description of all the classes in the S8 dataset.

A.1.3 Corel Image Collection (C371, C50, C43, C15)

The Corel Image Collection consists of the Corel Stock Photo CDs. Each CD includes 100 images of a common topic. We construct four different datasets from this collection.

Table A.2: Summary of the UIUC Sports dataset.

Category	Training set	Test set	Total
Coast	70	60	200
Forest	70	60	137
Highway	70	60	236
Inside City	70	60	182
Mountain	70	60	194
Open Country	70	60	250
Street	70	60	190
Tall Building	70	60	190
total	560	480	1579

Corel371 (C371)

The first dataset is “Corel371” (C371) which was first proposed in [35] for the task of automatic image annotation. C371 consists of $5,000$ images from 50 Corel Stock Photo CDs. Each image is further labeled with 1-5 semantic concepts. Overall there are 371 concepts in the vocabulary. C371 has since then been used to evaluate several other image annotation systems [41, 72, 21, 22] etc where 4500 images are used to train the system and the rest 500 for evaluation. A.3 provides a list of the annotation available for the C371 dataset along with the number of training and testing images per concept (in brackets). All images in this collection are available with color information. In this work, all the images from the Corel Collection are normalized to size 181×117 or 117×181 and converted from RGB to the YBR color space.

Table A.3: Summary of the C371 dataset.

water (1005,116); sky (883,105); tree (854,93); people (670,74); grass (446,51); buildings (408,54); mountain (307,38); flowers (269,27); snow (267,31); clouds (254,

Table A.3: (continued)

26); rocks (228,22); stone (212,20); street (203,26); plane (199,25); bear (198,22);
field (198,17); sand (184,19); birds (179,17); beach (177,18); boats (155,15);
jet (147,19); leaf (136,12); cars (134,17); plants (129,15); house (124,19);
bridge (123,15); polar (122,13); valley (122,11); garden (117,10); hills (113,18);
close-up (112,10); ruins (107,12); statue (106,11); horses (103,12); tracks (103,11);
sun (101,10); ice (99,12); wall (98,14); ocean (96,9); cat (96,11);
temple (94,10); train (94,11); tiger (91,10); coral (89,9); scotland (89,11);
swimmers (85,8); coast (84,5); window (79,8); branch (78,2); pool (77,11);
foals (77,9); sunset (76,7); sculpture (76,10); frost (74,7); head (71,2);
forest (71,11); fox (71,9); nest (71,7); mare (69,9); city (67,10);
railroad (63,8); ground (60,4); horizon (59,4); shops (59,4); petals (59,4);
arch (57,4); reefs (56,5); palace (56,4); reflection (55,9); park (55,2);
desert (55,11); skyline (53,6); locomotive (53,9); shore (51,8); castle (49,6);
pillar (49,9); river (48,4); town (48,9); road (47,4); deer (47,4);
waves (45,4); smoke (44,10); sea (43,2); church (42,6); market (40,2);
tower (40,7); coyote (37,2); light (37,6); courtyard (37,2); sign (37,2);
zebra (37,4); bush (36,1); fence (35,2); village (35,7); door (35,2);
landscape (35,4); pyramid (35,3); black (34,2); roofs (34,2); tundra (33,9);
display (32,1); shadows (32,3); elk (32,6); island (31,2); flight (30,1);
grizzly (30,7); harbor (30,4); rodent (30,4); runway (29,1); stems (29,2);
palm (28,3); tulip (28,3); antlers (28,4); dunes (28,1); man (28,1);
woman (28,1); turn (28,3); fish (27,6); restaurant (27,4); formula (27,4);
buddha (26,1); white-tailed (26,2); kauai (26,4); hut (25,6); herd (25,4);
formation (24,2); wood (24,4); food (24,2); museum (23,4); indian (22,3);
oahu (22,1); ships (21,3); flag (21,2); prop (21,1); hillside (21,3);
farms (21,2); bengal (21,6); cliff (21,0); hats (21,2); lizard (21,1);
prototype (21,4); gate (20,2); shrine (20,0); frozen (20,4); face (19,2);
log (18,2); arctic (18,3); bulls (18,5); caribou (18,4); moose (18,1);
canyon (18,3); baby (18,1); buddhist (18,3); straightaway (18,0); tables (17,2);
costume (17,3); hotel (17,2); fountain (17,1); night (17,2); tortoise (17,0);
path (16,1); stairs (16,2); figures (16,0); lawn (16,2); giant (16,0);

Table A.3: (continued)

giraffe (16,1); steel (16,0); hawaii (16,3); land (15,1); meadow (15,3);
cubs (15,1); autumn (15,0); umbrella (15,0); crystals (15,1); booby (15,5);
seals (15,0); maui (15,2); lake (14,1); windmills (14,2); monastery (14,2);
facade (14,0); mule (14,2); tusks (14,1); sphinx (14,1); anemone (13,1);
clothes (13,1); writing (13,0); ceremony (13,1); cottage (13,3); elephant (13,3);
monks (13,3); iguana (13,3); marine (13,3); reptile (13,1); f-16 (12,1);
tails (12,1); pagoda (12,0); fruit (12,2); poppies (12,0); pots (12,3);
albatross (12,1); girl (12,3); cow (11,4); guard (11,0); athlete (11,3);
steps (11,0); horns (11,1); fly (11,1); prayer (11,0); shrubs (10,3);
post (10,2); crab (10,1); entrance (10,1); column (10,2); relief (10,1);
penguin (10,0); row (10,0); antelope (10,2); bay (9,0); fan (9,1);
sunrise (9,1); vegetation (9,1); sailboats (9,0); chapel (9,0); paintings (9,0);
plaza (9,1); pond (9,0); vines (9,1); bench (9,0); waterfalls (9,0);
slope (9,1); goat (9,2); wolf (9,0); dog (8,0); stream (8,0);
lion (8,3); barn (8,2); glass (8,1); architecture (8,1); fog (8,0);
stick (8,0); wings (8,0); blooms (8,1); mosque (8,1); squirrel (8,2);
rainbow (7,0); dress (7,1); run (7,0); sheep (7,2); detail (7,1);
room (7,0); cathedral (7,2); monument (7,3); canal (7,1); interior (7,3);
mist (7,2); vineyard (7,1); lynx (7,1); african (7,1); pups (7,0);
carvings (6,0); kit (6,1); den (6,1); balcony (6,1); art (6,2);
decoration (6,2); chairs (6,0); crowd (6,0); cheese (6,0); silhouette (6,1);
terrace (6,1); cactus (6,2); outside (6,1); basket (5,1); drum (5,0);
winter (5,0); rockface (5,0); pair (5,0); nets (5,1); pattern (5,0);
blossoms (5,0); store (5,1); needles (5,1); designs (5,0); lily (5,0);
lighthouse (5,2); truck (5,1); marsh (5,1); porcupine (5,1); range (5,0);
pole (5,0); dance (5,1); plain (4,0); peaks (4,1); helicopter (4,0);
fall (4,0); sponges (4,0); star (4,0); cave (4,2); vegetables (4,0);
rose (4,0); dock (4,1); pottery (4,0); fawn (4,0); chrysanthemums (4,0);
trunk (4,2); eagle (4,0); whales (4,1); rabbit (4,0); animals (4,0);
shell (3,0); storm (3,0); crafts (3,1); festival (3,1); mural (3,0);
butterfly (3,1); carpet (3,0); floor (3,0); vendor (3,1); parade (3,0);

Table A.3: (continued)

doorway (3,1); texture (3,0); dust (3,0); pack (3,0); dall (3,0);
trail (3,0); shirt (3,0); pebbles (3,0); snake (3,1); moon (2,0);
cafe (2,1); angelfish (2,0); perch (2,0); sidewalk (2,2); spider (2,0);
tent (2,0); clearing (2,0); hands (2,0); crops (2,0); vehicle (2,1);
rice (2,0); tomb (2,0); calf (2,1); school (2,0); boeing (1,0);
diver (1,0); sails (1,1); model (1,0); railing (1,0); ladder (1,0);
rapids (1,0); military (1,0); mushrooms (1,0); hawk (1,0); orchid (1,1);
saguaro (1,0); mast (1,0); pepper (1,0); insect (1,0); glacier (1,0);
harvest (1,0); shade (1,0); ceiling (1,0); furniture (1,0); lichen (1,0);
remains (1,0); leopard (1,0); jeep (1,0); cougar (1,1); canoe (1,0);
race (1,0); grouper (0,1); moss (0,1); aerial (0,1);

Corel50,Corel43 (C50,C43)

We also use the image from C371 dataset to construct two more dataset, “Corel50” (C50) and “Corel43” (C43) for the task of scene classification task, relying on the CD labels for groundtruth instead of the image annotations. C50 contains 50 scene classes, each corresponding to one CD in the collection. For each CD, 90 images are used to learn class models and the remaining for testing. It has been argued that CD labels lead to an easy classification problem [173] as there is high variability between images from different CDs and high similarity among those from the same CD. To address these concerns, we construct another dataset from this collection, C43 that uses a set of manual annotations (disjoint from the CD labels) as ground truth. 43 semantic concepts are chosen from the set of annotations of [35] (those with a minimum of 100 annotated images) and 100 images are randomly selected per concept. Since an image can be labeled with more than one concept, this results in a total of 3102 images. Of these, 2766 are randomly selected to create a test set with approximately 90 images per label, and the remainder are used for testing. A correct classification is declared whenever the top predicted label matches any of the groundtruth labels.

Corel15 (C15)

Corel15 (C15) consists of $1,500$ images from another fifteen previously unused Corel Stock Photo CDs, viz. “Adventure Sailing”, “Autumn”, “Barnyard Animals”, “Caves”, “Cities of Italy”, “Commercial Construction”, “Food”, “Greece”, “Helicopters”, “Military Vehicles”, “New Zealand”, “People of World”, “Residential Interiors”, “Sacred Places”, “Soldier”. Once again, the CD themes (non-overlapping with those of C50 served as the ground truth. This dataset is used for the evaluation of image retrieval systems where $1,200$ images serve as the retrieval set and the remaining 300 images as the query set.

A.1.4 Flickr Images (F18)

To address some criticism that ‘Corel is easy’ [98, 172], we collected a second database from the online photo sharing website `www.flickr.com`. The images in this database were extracted by placing queries on the flickr search engine, and manually pruning images that appeared irrelevant to the specified queries. Note that the judgments of relevance did not take into account how well a content-based retrieval system would perform on the images, simply whether they appeared to be search errors (by flickr) or not. The images are shot by flickr users, and hence differ from the Corel Stock photos, which have been shot professionally. This database, “Flickr18” (F18), contains 1800 images divided into 18 classes viz. “Automobiles”, “Building and Landscapes”, “FacialCloseUp”, “Flora”, “FlowersCloseup”, “Food and Fruits”, “Frozen”, “Hills and Valley”, “Horsesl and Foal”, “JetPlanes”, “Sand”, “Sculpture and Statues”, “SeaandWaves”, “Solar”, “Township”, “Train”, “Underwater”, “Waterfun”, according to the manual annotations provided by the online users. F18 is again used for evaluating image retrieval systems where 20% of randomly selected images served as the query set and the remaining 80% as the retrieval set.

Table A.4: Summary of the TVGraz dataset.

Category	Training set	Test set	Total
Brain	109	47	156
Butterfly	195	51	246
Cactus	137	37	174
Deer	223	51	274
Dice	169	50	219
Dolphin	163	59	222
Elephant	120	54	174
Frog	215	67	282
Harp	131	42	173
Pram	96	42	138
total	1558	500	2058

A.1.5 TVGraz

The TVGraz dataset is a collection of web-pages compiled by Khan *et al.* [66]. The Google Image search engine was used to retrieve $1,000$ web-pages for each of ten categories from the Caltech-256 [52] dataset. The results were filtered into a set of $2,592$ positive web-pages, containing both text and image data, for which the image belonged to the query category. Due to copyright issues, the TVGraz database is stored as a list of URLs, and must be recompiled by each new user. We collected $2,058$ image-text pairs, since some URLs were defunct and we discarded web-pages that did not contain at least 10 words and one image. The median text length, per web-page, is 289 words. A random split was used to produce $1,558$ training and 500 test documents, as summarized in A.4.

A.1.6 Wikipedia

A novel dataset was assembled from the "Wikipedia featured articles", a continually updated collection of Wikipedia articles, which contained $2,669$ entries

Table A.5: Summary of the Wikipedia dataset.

Category	Training set	Test set	Total
Art & architecture	138	34	172
Biology	272	88	360
Geography & places	244	96	340
History	248	85	333
Literature & theatre	202	65	267
Media	178	58	236
Music	186	51	237
Royalty & nobility	144	41	185
Sport & recreation	214	71	285
Warfare	347	104	451
total	2173	693	2866

when the data was collected, in October 2009. These articles, which are selected and reviewed for style and quality by Wikipedia's editors, are often accompanied by one or more pictures from the Wikimedia Commons, supplying a text-image pairing. The Wikipedia featured articles are divided into 29 categories, but some contain very few entries. We considered only articles from the 10 most populated categories, which were used as a semantic vocabulary. Since the featured articles tend to have multiple images and span multiple topics, each article was split into sections, based on its section headings. Each image was assigned to the section in which it was placed by the author(s). This produced a total of $7,114$ sections, which are internally more coherent and usually contain a single picture. The dataset was then pruned, by keeping only sections with exactly one image and at least 70 words. The final corpus contains a total of $2,866$ documents. The median text length is 200 words. A random split was used to produce a training set of $2,173$ documents and a test set of 693 documents, as summarized in A.5.

Appendix B

Generalized Expectation maximization (GEM)

The parameters $\Lambda^w = \{\beta_k^w, \boldsymbol{\alpha}_k^w\}$ of the contextual class models of (6.1) are learned using GEM. This is an extension of the well known EM algorithm, applicable when the M-step of the latter is intractable. It consists of two steps. The E-Step is identical to that of EM, computing the expected values of the component probability mass β_k. The generalized M-step estimates the parameters $\boldsymbol{\alpha}_k$. Rather than solving for the parameters of maximum likelihood, it simply produces an estimate of higher likelihood than that available in the previous iteration. This is known to suffice for convergence of the overall EM procedure [30]. We resort to the Newton-Raphson algorithm to obtain these improved parameter estimates, as suggested in [95] for single component Dirichlet distributions. Omitting the dependence on the concept index w for brevity, the algorithm iterates between two steps,

E-step: compute

$$h_{dk} = \frac{\mathcal{D}ir(\boldsymbol{\pi}_d; \boldsymbol{\alpha}_k)\beta_k}{\sum_l \beta_l \mathcal{D}ir(\boldsymbol{\pi}_d; \boldsymbol{\alpha}_l)} \tag{B.1}$$

M-step: set

$$(\beta_k)^{new} = \frac{N_k}{N}, \quad \text{where} \quad N = \sum_{dk} h_{dk}, N_k = \sum_{d} h_{dk} \tag{B.2}$$

$$(\boldsymbol{\alpha}_k)^{new} = (\boldsymbol{\alpha}_k)^{old} + \mathcal{H}^{k^{-1}}\mathbf{g}^k \tag{B.3}$$

$$\text{where} \quad \mathbf{g}_i^k = N_k(\Psi(\sum_{p=1}^{L}\alpha_p) - \Psi(\alpha_i)) + \sum_d h_{dk}\log\pi_{id} \tag{B.4}$$

$$\text{and} \quad \mathcal{H}_{ii}^k = N_k(\Psi'(\sum_{p=1}^{L}\alpha_p) - \Psi'(\alpha_i)) \tag{B.5}$$

$$\mathcal{H}_{ij}^k = N_k(\Psi'(\sum_{p=1}^{L}\alpha_p)), \tag{B.6}$$

Ψ and Ψ' are the Digamma and Trigamma functions [95].

Appendix C

Computation of Image-SMNs

Given N patch-based SMNs, $\boldsymbol{\pi}^{(n)}$, the Image-SMN $\boldsymbol{\pi}^*$ is

$$\begin{aligned}
\boldsymbol{\pi}^* &= \arg\min_{\boldsymbol{\pi}} \frac{1}{N}\sum_{n=1}^{N} KL(\boldsymbol{\pi}||\boldsymbol{\pi}^{(n)}) \\
&= \arg\min_{\boldsymbol{\pi}} \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{L} \pi_i \log\frac{\pi_i}{\pi_i^{(n)}} \\
&= \arg\min_{\boldsymbol{\pi}} \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{L} \left[\pi_i \log \pi_i - \pi_i \log \pi_i^{(n)}\right]
\end{aligned}$$

subject to $\sum_{i=1}^{L} \pi_i = 1$. This has Lagrangian

$$\mathcal{L}(\pi, \lambda) = \sum_{i=1}^{L} \pi_i \log \pi_i - \frac{1}{N}\sum_{i=1}^{L} \pi_i \sum_{n=1}^{N} \log \pi_i^{(n)} + \frac{\lambda}{N}(1 - \sum_{i=1}^{L} \pi_i).$$

Setting derivatives with respect to π_i to zero leads to

$$1 + \log \pi_i - \frac{1}{N}\sum_{n=1}^{N} \log \pi_i^{(n)} - \frac{\lambda}{N} = 0, \tag{C.1}$$

$$\text{or} \qquad \pi_i = \exp\left(\hat{\lambda} + < \log \pi_i >\right) \tag{C.2}$$

where $< \log \pi_i >= \frac{1}{N}\sum_{n=1}^{N} \log \pi_i^{(n)}$ and $\hat{\lambda} = \frac{\lambda}{N} - 1$. Summing over i and using the constraint $\sum_i \pi_i = 1$,

$$1 = \exp(\hat{\lambda}) \sum_{i=1}^{L} \exp < \log \pi_i > \tag{C.3}$$

$$\exp(\hat{\lambda}) = \frac{1}{\sum_{i=1}^{L} \exp < \log \pi_i >}. \tag{C.4}$$

Substituting (C.4) in (C.2),

$$\pi_i^* = \frac{\exp < \log \pi_i >}{\sum_{i=1}^{L} \exp < \log \pi_i >} \tag{C.5}$$

$$= \frac{\exp \frac{1}{N} \sum_n \log \pi_i^{(n)}}{\sum_i \exp \frac{1}{N} \sum_n \log \pi_i^{(n)}}. \tag{C.6}$$

Appendix D

Variational Approximation

Variational methods approximate the posterior $P(\boldsymbol{\pi}, w_{1:N}|x_{1:N})$ by a mean-field variational distribution $q(\boldsymbol{\pi}, w_{1:N})$, indexed by free variational parameters, within some class of tractable probability distributions $\mathcal{F}$. These distributions usually assume independent factors,

$$q(\boldsymbol{\pi}, w_{1:N}) = q(\boldsymbol{\pi}; \boldsymbol{\gamma}) \prod_n q(w_n; \boldsymbol{\phi}_n) \tag{D.1}$$

where $q(y)$ and $q(z_i)$ are categorical models, and $q(\boldsymbol{\pi})$ a Dirichlet distribution. Given an observation $x_{1:N}$, the optimal variational approximation minimizes the Kullback-Leibler (KL) divergence between the two posteriors

$$\begin{aligned} q^* &= \arg\min_{q \in \mathcal{F}} KL(q(\boldsymbol{\pi}, w_{1:N}) || P(\boldsymbol{\pi}, w_{1:N}|x_{1:N})) && \text{(D.2)} \\ &= \mathcal{L}(q(\boldsymbol{\pi}, w_{1:N})) + \log P(x_{1:N}) && \text{(D.3)} \end{aligned}$$

where,

$$\mathcal{L}(q(\boldsymbol{\pi}, w_{1:N})) = E_q[\log q(\boldsymbol{\pi}, w_{1:N})] - E_q[\log P(\boldsymbol{\pi}, w_{1:N}, x_{1:N})]. \tag{D.4}$$

Since the data likelihood $P(x_{1:N})$ is constant for an observed image, the optimization problem is identical to

$$q^*(\boldsymbol{\pi}, w_{1:N}) = \arg\min_{q \in \mathcal{F}} \mathcal{L}(q(\boldsymbol{\pi}, w_{1:N})), \tag{D.5}$$

From Appendix A.3 of [14], the update rule for coordinate descent of the variational parameters is

$$\gamma_i^* = \sum_n \phi_{ni} + \alpha_i \tag{D.6}$$

$$\phi_{ni}^* \propto P_{X|W}(x_n|w_n = i)\ e^{\psi(\gamma_i) - \psi(\sum_j \gamma_j)} \tag{D.7}$$

such that $\sum_i \phi_{ni} = 1$ and, where α_i are the parameters of the prior class distribution $P(\boldsymbol{\pi}; \boldsymbol{\alpha})$ and ψ is the Digamma function [95]. Once the parameters of the variational distribution are obtained, the SMN for an image can be computed as,

$$\begin{aligned} \boldsymbol{\pi}^* &= \arg\max_{\boldsymbol{\pi}} q(\boldsymbol{\pi}; \boldsymbol{\gamma}) && \text{(D.8)} \\ &= \arg\max_{\boldsymbol{\pi}} \log q(\boldsymbol{\pi}; \boldsymbol{\gamma}) && \text{(D.9)} \\ &= \arg\max_{\boldsymbol{\pi}} \sum_j^L (\gamma_j - 1) \log \pi_j && \text{(D.10)} \\ \text{such that, } & \sum_j \pi_j = 1 && \text{(D.11)} \end{aligned}$$

Using the Lagrange multiplier, λ, we get

$$J(\boldsymbol{\pi}, \lambda) = \sum_j^L (\gamma_j - 1) \log \pi_j + \lambda(1 - \sum_j^L \pi_j) \tag{D.12}$$

Taking partial derivatives with respect to, π_j and λ and setting them to zero we get,

$$\frac{\partial J}{\partial \pi_j} = \frac{(\gamma_j - 1)}{\pi_j} - \lambda = 0, \forall j \tag{D.13}$$

$$\frac{\partial J}{\partial \lambda} = 1 - \sum_j^L \pi_j = 0 \tag{D.14}$$

From (D.13) and (D.14) we get,

$$\pi_j = \frac{\gamma_i - 1}{\sum_j \gamma_j - L} \tag{D.15}$$

Appendix E

Parameter Estimation in cLDA

The parameters $(\boldsymbol{\eta}, \boldsymbol{\alpha}_{1:C}, \Lambda_{1:K})$ of cLDA are learned using variational Expectation Maximization (EM) algorithm. This iterates between:

Variational E-Step consists of approximating the posterior $P(\boldsymbol{\pi}^d, z_{1:N}^d | \mathcal{I}^d, y^d)$ for an image $\mathcal{I}^d = \{w_1^d, \ldots, w_N^d\}$ using the variational distribution,

$$q(\boldsymbol{\pi}^d, z_{1:N}^d) = q(\boldsymbol{\pi}^d; \boldsymbol{\gamma}^d) \prod_n q(z_n^d; \boldsymbol{\phi}_n^d) \tag{E.1}$$

Similar to the variational inference of LDA (see Appendix D), the variational parameters can be computed using the update rules,

$$\gamma_k^{d*} = \sum_n \phi_{nk}^d + \alpha_{y^d k} \tag{E.2}$$

$$\phi_{nk}^{d*} \propto \Lambda_{k w_n^d} \, \exp\left[\psi(\gamma_k^d)\right] \tag{E.3}$$

where, $\sum_k \phi_{nk}^d = 1$. Note that in cLDA, since each class is associated with a separate prior over the topic simplex, (E.2) differs from (D.6), in that $\boldsymbol{\alpha}$ parameters are class specific.

M-Step consists of computing the values of the parameters $(\boldsymbol{\alpha}_{1:C}, \Lambda_{1:K})$, where $\boldsymbol{\alpha}_y$ is obtained by maximizing,

$$\begin{aligned}\boldsymbol{\alpha}_y^* = \arg\max_{\boldsymbol{\alpha}_y} - \sum_d \delta(y^d, y) \log \mathcal{B}(\boldsymbol{\alpha}_y) \\ + \sum_d \sum_k \delta(y^d, y)(\alpha_{y^d k} - 1) E_q[\log \pi_k^d] \quad \text{(E.4)}\end{aligned}$$

where,

$$E_q[\log \pi_k^d] = \psi(\sum_l \gamma_l^d) - \psi(\gamma_k^d) \quad \text{(E.5)}$$

$$\mathcal{B}(\boldsymbol{\alpha}_y) = \frac{\prod_k (\Gamma(\alpha_{yk})}{\Gamma(\sum_k \alpha_{yk})} \quad \text{(E.6)}$$

and $\Gamma()$ is the Gamma function. The above optimization can be carried out using the method of Newton-Raphson gradient ascent as detailed in [95].

Λ_k is obtained by maximizing,

$$\Lambda_{kv}^* = \arg\max_{\Lambda_k} \sum_d \sum_n \delta(w_n^d, v) \phi_{nk}^d \log \Lambda_{kv} \quad \text{(E.7)}$$

such that $\sum_{v=1}^{|\mathcal{V}|} \Lambda_{kv} = 1$, using the method of Lagrange multipliers which results in the closed form update,

$$\Lambda_{kv} \propto \sum_d \sum_n \delta(w_n^d, v) \phi_{nk}^d \quad \text{(E.8)}$$

where, proportionality symbols means that Λ_k is normalized to sum to 1. Note that its common to assume a uniform class prior and we assume $\boldsymbol{\eta}_y = \frac{1}{C}, \forall y \in \mathcal{Y}$.

Appendix F

Parameter Estimation in topic-supervised LDA models

In this section, we discuss the parameter estimation for ts-cLDA. The parameter for other topic-supervised models can be computed using a similar approach. Topic supervision decouples cLDA learning into two steps: 1) learning of the parameters $\Lambda_{1:K}$ of the topic-conditional distributions, and 2) learning of the parameters $\boldsymbol{\alpha}_{1:C}$ of the class-conditional distributions[1].

F.1 Learning Topic Conditional Distributions

As discussed in Section 7.5, since the topics are defined over the class vocabulary $\mathcal{T} = \mathcal{V}$, in absence of the individual topic labels z_n^d for the visual words w_n^d during learning, we assume all topic labels are equal to the image class y^d, i.e. $z_n^d = y^d \ \forall n, d$. Although, this is not true in reality, such an approximation has been shown to be effective both through the design of image labeling systems [21] and through theoretical connections to multiple instance learning. Infact, this is an implicit assumption in learning the parameters of the flat model. Thus, the ML estimates of Λ_k can be obtained from

$$\Lambda_{kv}^{*} = \arg\max_{\Lambda_k} \sum_d \sum_n \delta(y^d, k)\delta(w_n^d, v) \log \Lambda_{kv} \tag{F.1}$$

[1] Note that $\boldsymbol{\eta}$ is again assumed to follow a uniform distribution

such that $\sum_{v=1}^{|\mathcal{V}|} \Lambda_{kv} = 1$. Using the method of Lagrange multipliers, the solution to the optimization problem is given by,

$$\Lambda_{kv} = \frac{\sum_d \sum_n \delta(y^d, k)\delta(w_n^d, v)}{\sum_j \sum_d \sum_n \delta(y^d, j)\delta(w_n^d, v)} \tag{F.2}$$

F.2 Learning Class Conditional Distribution

Once the topic-conditional distributions are learned, the parameters $\boldsymbol{\alpha}_c$ of the class-conditional distributions can be learned by the maximizing the likelihood of the data, $P(y^d, w_{1:N}^d)$ using the standard variational EM algorithm, this approach iterates between two steps:

Variational E-Step consists of computing,

$$\gamma_k^{d*} = \sum_n \phi_{nk}^d + \alpha_{y^d k} \tag{F.3}$$

$$\phi_{nk}^{d*} \propto \Lambda_{k w_n^d} \ \exp\left[\psi(\gamma_k^d)\right] \tag{F.4}$$

where, proportionality symbols means that ϕ_n^d is normalized to sum to 1.

M-Step consists of computing the values of the parameters $\boldsymbol{\alpha}_{1:C}$ (note that $\Lambda_{1:K}$ is already computed) similar to (E.4).

Appendix G

Implementation Details of the various systems

We conclude this thesis by discussing some implementation details of various recognition systems proposed in this work. This discussion is intended mostly for those interested in replication portions or the entirety of the work described in the thesis. Although, many of the details have been mentioned in the previous chapters, we believe that it is useful to present a cohesive summary of the most important points.

G.1 Image Representation

Given a database of images, images are represented either using DCT or SIFT descriptors, where both BoF and BoW models are employed.

G.1.1 SIFT Features

To compute SIFT, image patches are selected either 1) by interest point detection, referred to as SIFT-INTR, or 2) on a dense regular grid, referred to as SIFT-GRID. For SIFT-INTR, interest points computed using three operators — Harris-Laplace, Laplace-of-Gaussian, and Difference-of-Gaussian — which are then merged. These measures also provide scale information, which is used in the computation of SIFT features. For SIFT-GRID, feature points are sampled every 8 pixels and the descriptor is computed over a 16×16 neighborhood around each feature point. Both interest points and SIFT features are computed with the implementation of LEAR — `http://lear.inrialpes.fr/people/dorko/downloads.html`. On average, the two strategies yield similar number of samples per image. The SIFT descriptors are scaled by a factor of 100 to prevent numerical instabilities during learning of the Gaussian mixture models.

G.1.2 DCT Features

DCT features are computed on a dense regular grid, with a step of 1-to-8 pixels (usually improved performance is obtained with lower step size, but at the cost of computation). 8×8 image patches are extracted around each grid point, and 8×8 DCT coefficients computed per patch and color channel. The DCT coefficients are vectorized into a row vector using the coefficient scanning mechanism defined by the MPEG standard. For monochrome images this results in a feature space of 64 dimensions. For color images the space is 192 dimensional, where the vectors for corresponding channels are interleaved. We currently use the YBR color-space defined by MPEG, but this selection has not been subject to detailed scrutiny.

G.1.3 Bag-of-Features

Using the bag of features representation, each image is modeled as a Gaussian Mixture Model (GMM) with a fixed number C of mixture components. The default value is $C = 8$ for DCT and $C = 16$ for SIFT, but can be modified when the database is initialized. In general using more mixture components result in improved performance, however the gains diminish over 16 mixture components. All Gaussian mixture parameters are estimated using the EM algorithm. The implementation is fairly standard, the only details worth mentioning are the following.

1. All Gaussians have diagonal covariances.

2. In order to avoid singularities, a variance limiting constrain is applied. This constrain places a minimum value of 10(0.01) on the variance along each dimension of the DCT(SIFT) feature space. Note, if new features are being introduced, a good estimate of minimum covariance using cross-validation techniques should be obtained.

3. Initialization is performed with a vector quantizer designed by the Linde-Buzo-Gray (LBG) algorithm, using a variation of the cell splitting method described in [81]. For more details please see [162].

4. The EM iterations for DCT(SIFT) features are restricted to 5(15) iterations. More iterations are required for SIFT features as 1) there are more mixture components, 2) unlike DCT, every dimension of SIFT has high variance.

G.1.4 Bag-of-Words

To obtain the bag-of-words representation, the space of image features is quantized using the LBG algorithm with a fixed number B of clusters. The default value is $B = 256$ codeword for both DCT and SIFT, although several experiments are performed with codewords as high as 4096 (e.g. topic-supervised LDA models). In general increasing the size of codebook leads to improved performance. Note that the initialization of GMM uses the codebooks learned with LBG algorithm.

G.1.5 Semantic Multinomial

To compute the Semantic Multinomial(SMN), the posterior probability of the concepts given an image, is computed using (2.21) for BoF and using (2.23) for BoW. (2.21) yields SMN which are almost uniform for BoW. SMN are regularized using $\pi_0 = 1$ for QBSE and $\pi_0 = 0.0001$ for holistic context models. Unless otherwise mentioned, similarity between two SMNs is computed using KL divergence.

G.2 Concept/Category Models

G.2.1 Appearance Based Models

GMM is the choice of probability distribution for the appearance based concept models. Given a training set of images, along with their GMM learned using the approach described above, appearance models are learned using hierarchical estimation technique proposed in [162]. For DCT(SIFT), 128(512) mixture components are used, although more mixture components leads to better performance.

G.2.2 Holistic Context Models

Contextual class models are learned using the outputs of appearance based models. Dirichlet Mixture Model (DMM) is the choice of probability distribution. Given a training set of images, DMM can be learned using image-SMNs, however since most datasets used in this work has only $\sim$ 100 images per concept available for training, data augmentation techniques of Section6.3.5 is employed. To increase the cardinality of the training sets used for contextual modeling, 800 random sets of 30 patches are sampled per image, yielding 800 patch-SMNs per image. Image-SMNs are then computed from these, with (2.21) or (2.23).

The parameters of the contextual class models are learned using GEM as described in Appendix B. The implementation is fairly standard, the only details worth mentioning are the following.

1. The number of mixture components is set to 42. Given sufficient training data, in general higher number of mixture components yield better recognition accuracies, however the benefits are limited over 40 mixture components.

2. In order to avoid singularities, a variance limiting constrain is applied. Since both high and low values of α parameter can lead to low variance, α values are constrained to a minimum and a maximum value of 0.01 and 100 respectively. The performance of the system is not very sensitive to the choice of these values, however a sanity check must be performed to ensure that they are not beyond reasonable limits.

3. Initialization is performed with a vector quantizer designed by the Linde-Buzo-Gray (LBG) algorithm, using a variation of the cell splitting method described as described above. The similarity between different SMN is computed using KL divergence.

4. The GEM iterations are restricted to 15 iterations.

G.3 Topic Supervised LDA

Topic supervised LDA models are built using BoW representation with the vocabulary size ranging from 128 to 4096. For each dataset, codebooks are generated from a random collection of 300 examples per training image. For experiments using LDA and sLDA we use the code available online[1]. This code was modified for cLDA and topic-supervised LDA. The number of topics is varied from 10 to 100 for topic discovery approaches. For topic-supervised models, the number of topics is equal to the number of classes. The α_k parameter is set to 1 in all experiments except for cLDA and ts-cLDA, where an asymmetric $\boldsymbol{\alpha}_y$ parameter is learned per class. Although not explicitly shown in Figure Figure 7.1, we use the "smoothed" version of various LDA models with a Dirichlet prior on the topic-distributions [14], using a symmetric hyper-parameter of 0.001. The performance of various models is not very sensitive to the choice of both α_k and the smoothing parameter.

[1] http://www.cs.princeton.edu/~blei/lda-c/ and http://www.cs.princeton.edu/~chongw/slda/ respectively.