Chapter 2

Semantic Image Representation

In this chapter we first present a review of the existing solutions to the problem of image retrieval and scene classification followed by a brief review of low level image representation. We then introduce the semantic image representation for scene classification.

2.1 Preliminaries

We start by briefly reviewing appearance-based modeling and the design of visual recognition systems for image retrieval and scene classification.

2.1.1 Notations

Consider a image database $\mathcal{D} = \{\mathcal{I}_1, \ldots, \mathcal{I}_D\}$ where images $\mathcal{I}_i$ are observations from a random variable $\mathbf{X}$, defined on some feature space $\mathcal{X}$. For example, $\mathcal{X}$ could be the space of discrete cosine transform (DCT), or SIFT descriptors. Each image is represented as a set of N *low-level feature vectors* $\mathcal{I} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathcal{X}$, assumed to be sampled independently. This is commonly referred to as the "bag-of-features" (BoF) representation, since the image is represented as an orderless collection of visual features. A popular extension of the BoF representation is the "bag-of-words" (BoW) [27, 74] representation. In BoW representation, the feature space $\mathcal{X}$ is further quantized into $|\mathcal{V}|$ unique bins, defined by a collection of centroids, $\mathcal{V} = \{1, \ldots, \mathcal{V}|\}$, and each feature vector $\mathbf{x}_n,\ n \in \{1, \ldots, N\}$ is mapped to its closest centroid. Each image is then represented as a collection of *visual words*, $\mathcal{I} = \{v_1, \ldots, v_N\}, v_n \in \mathcal{V}$, where v_n is the bin that contains the feature vector $\mathbf{x}_n$. This facilitates the representation of the image as a vector in $\Re^{|\mathcal{V}|}$, however it has been argued that feature quantization leads to significant degradation in its discriminative power [15]. In this work, we rely on both BoF and BoW representation, BoF being the default choice of image representation.

2.1.2 Image Retrieval Systems

The starting point for any retrieval system is the image database $\mathcal{D} = \{\mathcal{I}_1, \ldots, \mathcal{I}_D\}$. Although several image retrieval formulations are possible, in this work, the framework underlying all query paradigms is that of minimum probability of error retrieval, as introduced in [156]. Under this formulation, each image is considered as an observation from a different class, determined by a random variable Y defined on $\{1, \ldots, D\}$. Given a query image $\mathcal{I}$, the MPE retrieval decision is to assign it to the class of largest posterior probability, i.e.

$$y^* = \arg\max_y P_{Y|\mathbf{X}}(y|\mathcal{I}). \tag{2.1}$$

and image retrieval is based on the mapping $g : \mathcal{X} \to \{1, \ldots, D\}$ of (2.1). Using Bayes rule and under the assumption of independent samples this is equivalent to,

$$y^* = \arg\max_y P_{\mathbf{X}|Y}(\mathcal{I}|y)P_Y(y). \tag{2.2}$$

$$= \arg\max_y \prod_j P_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}_j|y)P_Y(y). \tag{2.3}$$

where $P_{\mathbf{X}|\mathbf{Y}}(x|y)$ is the class conditional density, which serves as the *appearance model* for the y^{th} image and $P_Y(y)$ the class prior. Although any prior class distribution $P_Y(y)$ can be supported, we assume a uniform distribution in what follows.

To model the appearance distribution, we rely on Gaussian mixture models (GMM). These are popular models for the distribution of visual features [21, 57, 145, 12] and have the form

$$P_{\mathbf{X}|Y}(\mathbf{x}|y;\Gamma_y) = \sum_j \alpha_y^j \mathcal{G}(\mathbf{x}, \mu_y^j, \mathbf{\Sigma}_y^j) \tag{2.4}$$

where, α_y is a probability mass function such that $\sum_j \alpha_y^j = 1$, $\mathcal{G}(\mathbf{x}, \mu, \Sigma)$ a Gaussian density of mean μ and covariance Σ, and j an index over the mixture components. Some density estimation [33] procedure can be used to estimate the parameters of this distribution. In this work we use the well known expectation-maximization (EM) algorithm [30]. Henceforth, we refer to the above retrieval paradigm as *query-by-visual-example* (QVBE).

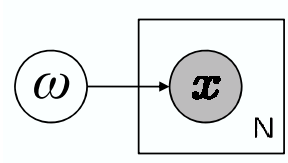

Figure 2.1: The generative model underlying image formation at the appearance level. w represents a sample from a vocabulary of scene categories or semantic concepts, and an image $\mathcal{I}$ is composed of N patches, $\mathbf{x}_n$, sampled independently from $P_{\mathbf{X}|W}(\mathbf{x}|w)$. Note that, throughout this work, we adopt the standard plate notation of [14] to represent graphical models.

2.1.3 Scene Classification Systems

A scene classification system appends the database $\mathcal{D}$ with a vocabulary of scene category $\mathcal{W} = \{1, \ldots, K\}$ and each image with a scene label $\mathbf{w}_i$, making $\mathcal{D}^{\mathcal{W}} = \{(\mathcal{I}_1, \mathbf{w}_1), \ldots, (\mathcal{I}_D, \mathbf{w}_D)\}$. The scene label $\mathbf{w}_i$ is considered to be an observation from a scene category random variable W defined on $\mathcal{W}$. Note that, for scene classification systems, the label $\mathbf{w}_i$ is an indicator vector such that $\mathbf{w}_{i,j} = 1$ if the i^{th} image is an observation from the j^{th} scene category. Each scene category induces a probability density $\{P_{\mathbf{X}|W}(\mathbf{x}|w)\}_{w=1}^{K}$ on $\mathcal{X}$, from which feature vectors are drawn. This is denoted as the *appearance model* for the category w which describes how observations are drawn from the low-level visual feature space $\mathcal{X}$. As shown in Figure 2.1, the generative model for a feature vector $\mathbf{x}$ thus consists of two steps: first a category label w is selected, with probability $P_W(w) = \pi_w$, and the feature vector then drawn from $P_{\mathbf{X}|W}(\mathbf{x}_n|w)$. Both concepts and feature vectors are drawn independently, with replacement.

Given a new image $\mathcal{I}$, classification is performed using the minimum probability of error framework, where the optimal decision rule is to assign it to the category of largest posterior probability

$$w^* = \arg\max_w P_{W|\mathbf{X}}(w|\mathcal{I}). \tag{2.5}$$

where $P_{W|\mathbf{X}}(w|\mathcal{I})$ is posterior probability of category w given $\mathcal{I}$ and can be com-

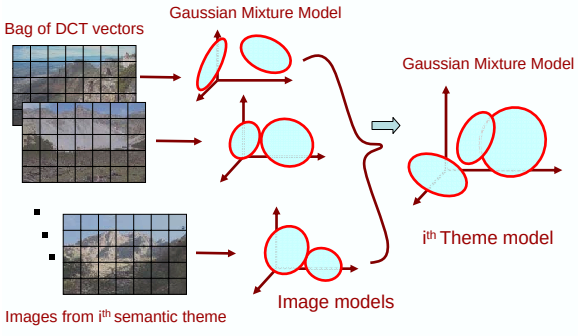

Figure 2.2: Learning the scene category (semantic concept) density from the set $\mathcal{D}_w$ of all training images annotated with the w^{th} caption in $\mathcal{W}(\mathcal{L})$, using hierarchical estimation [21]

.

puted used Bayes rule under the assumption of independent samples as,

$$P_{W|\mathbf{X}}(w|\mathcal{I}) = \frac{P_{\mathbf{X}|W}(\mathcal{I}|w)P_W(w)}{P_{\mathbf{X}}(\mathcal{I})}. \tag{2.6}$$

$$= \frac{\prod_j P_{\mathbf{X}|\mathbf{W}}(\mathbf{x}_j|w)P_W(w)}{\prod_j P_{\mathbf{X}}(\mathbf{x}_j)} \tag{2.7}$$

Although any prior class distribution $P_W(w)$ can be supported, we assume a uniform distribution in what follows. This leads to

$$P_{W|\mathbf{X}}(w|\mathcal{I}) \propto \frac{\prod_j P_{\mathbf{X}|\mathbf{W}}(\mathbf{x}_j|w)}{\prod_j P_{\mathbf{X}}(\mathbf{x}_j)} \tag{2.8}$$

The appearance model $P_{\mathbf{X}|\mathbf{W}}(x|w)$ is modeled using a GMM, defined by the parameters $\Omega_w = \{\nu_w^j, \mathbf{\Phi}_w^j, \beta_w^j\}$,

$$P_{\mathbf{X}|W}(\mathbf{x}|w;\Omega_w) = \sum_j \beta_w^j \mathcal{G}(\mathbf{x}, \nu_w^j, \mathbf{\Phi}_w^j) \tag{2.9}$$

where, β_w is a probability mass function such that $\sum_j \beta_w^j = 1$ and j an index over the mixture components. The parameters Ω_w are learned from the set $\mathcal{D}_w^{\mathcal{W}}$ of all training images annotated with the w^{th} category using some density estimation procedure. In this work we rely on a *hierarchical estimation* procedure first proposed in [159], for image indexing. As shown in Figure 2.2, this procedure is itself composed of two steps. First, a Gaussian mixture is learned for each image in $\mathcal{D}_w^{\mathcal{W}}$,

producing a sequence of mixture densities

$$P_{\mathbf{X}|Y,W}(\mathbf{x}|y,w) = \sum_k \alpha_{w,y}^k \mathcal{G}(\mathbf{x}, \mu_{w,y}^k, \Sigma_{w,y}^k), \tag{2.10}$$

where Y is a hidden variable that indicates the index of the image in $\mathcal{D}_w^{\mathcal{W}}$. Note that, if a QBVE has already been implemented, these densities are just replicas of the ones of (2.4). In particular, if the mapping $\mathcal{M} : \{1,\ldots,L\} \times \{1,\ldots,D\} \rightarrow \{1,\ldots,D\}$ translates the index (w,y) of the y^{th} image in $\mathcal{D}_w^{\mathcal{W}}$ into the image's index w on $\mathcal{D}^{\mathcal{W}}$, i.e. $w = \mathcal{M}(w,y)$, then

$$P_{\mathbf{X}|Y,W}(\mathbf{x}|y,w) = P_{\mathbf{X}|W}(\mathbf{x}|\mathcal{M}(w,y)).$$

Omitting, for brevity, the dependence of the mixture parameters on the semantic class w, assuming that each mixture has κ components, and that the cardinality of $\mathcal{D}_w^{\mathcal{W}}$ is D_w, this produces $D_w\kappa$ mixture components of parameters $\{\alpha_y^k, \mu_y^k, \Sigma_y^k\}, y = 1,\ldots,D_w$, $k = 1,\ldots,\kappa$. The second step is an extension of the EM algorithm, which clusters the Gaussian components into the mixture distribution of (2.9), using a hierarchical estimation technique (see [21, 159] for details). Because the number of parameters in each image mixture is orders of magnitude smaller than the number of feature vectors extracted from the image, the complexity of estimating concept mixtures is negligible when compared to that of estimating the individual image mixtures.

2.1.4 Image Representation

The literature on image representation is vast and goes back over five decades [1]. Although any type of visual features are acceptable, we only consider *localized features*, i.e., features of limited spatial support [153, 94, 150, 117]. Thus, a localized feature is a representation of a collection of adjoining image pixels, separating it from its immediate neighborhood. Usually image properties — such as intensity, color, texture, edges, edge orientations, frequency spectrum — change across these features. Localized features do not require sophisticated image segmentation procedures, which makes them computationally efficient and robust to scene clutter. Owing to these benefits, in recent years, they have been

quite successful for visual recognition tasks [94]. A large number of localized features have been proposed in the literature, the simplest being a vector of image pixel intensities [77]. Other descriptors emphasize different image properties like color [153, 49], texture [117, 111, 34], shape [49, 7], edges [85, 94], frequency spectrum [46, 62, 118, 156] etc. A comparison of these features for visual recognition tasks was presented in [153, 94]. In this work, since the main aim is to present an image representation that incorporates semantic cues, we do not debate on the choice of low-level feature representation, and rely on two popular localized image representations viz. scale invariant feature transform (SIFT) and discrete cosine transform (DCT). Infact, in Chapter 6 we show that, the semantic image representation improves over low-level visual features and moreover, the choice of low-level feature representation is not critical to the gains achieved. Next we present a brief description of both DCT and SIFT.

Discrete Cosine Transform

The discrete cosine transform (DCT) [62] expresses an image patch in terms of sum of cosine functions oscillating at different frequencies. A DCT of an image patch of size (N_1, N_2) is obtained as,

$$X_{k_1,k_2} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x_{n_1,n_2} \cos\left[\frac{\pi}{N_1}\left(n_1 + \frac{1}{2}\right)k_1\right] \cos\left[\frac{\pi}{N_2}\left(n_2 + \frac{1}{2}\right)k_2\right]. \quad (2.11)$$

The DCT is widely used in image compression, and previous recognition experiments have shown that DCT features can lead to recognition rates comparable to those of many features proposed in the recognition literature [162]. It has also been shown that, for local image neighborhoods, DCT features approximates principal component analysis (PCA). This makes the space of DCT coefficients a natural choice for the feature space, $\mathcal{X}$, for visual recognition.

In this thesis, DCT features are computed on a dense regular grid, with a step of 8 pixels. 8×8 image patches are extracted around each grid point, and 8×8 DCT coefficients computed per patch and color channel. For monochrome images this results in a feature space of 64 dimensions. For color images the space is 192 dimensional.

Scale Invariant Feature Transform

The scale invariant feature transformation (SIFT), was proposed in [85] as a feature representation invariant to scale, orientation, and affine distortion, and partially invariant to illumination changes. SIFT is a measure of the orientations of the edges pixels in a given image patch. To compute the SIFT, 8-bin orientation histograms are computed in a 4×4 grid. This leads to a SIFT feature vector with $4 \times 4 \times 8 = 128$ dimensions. This vector is normalized to enhance invariance to changes in illumination.

SIFT can computed for image patches which 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. While several interest point detectors are available in the literature, in this thesis SIFT-INTR is computed using interest points obtained with three saliency measures — Harris-Laplace, Laplace-of-Gaussian, and Difference-of-Gaussian — which are merged. These measures also provide scale information, which is used in the computation of SIFT features. For a dense grid, SIFT-GRID, feature points are sampled every 8 pixels. For both the strategies, SIFT features[1] are then computed over a 16×16 neighborhood around each feature point. On average, the two strategies yield similar number of samples per image.

2.2 Semantic Image Representation

While appearance features are intensity, texture, edge orientations, frequency bases, etc. those of the semantic representation are concept probabilities. Semantic image representation differs from appearance based representation in that, images are represented by vectors of *concept counts* $\mathcal{I} = (c_1, \ldots, c_L)^T$, rather than being sampled from low-level feature space $\mathcal{X}$. Each low level feature vector $\mathbf{x}$ for a given image, is assumed to be sampled from the probability distribution of a semantic concept and c_i is the number of low level feature vectors drawn from the i^{th} concept. The count vector for the y^{th} image is drawn from a multinomial

[1]Computed using the SIFT implementation made available by LEAR at `http://lear.inrialpes.fr/people/dorko/downloads.html`

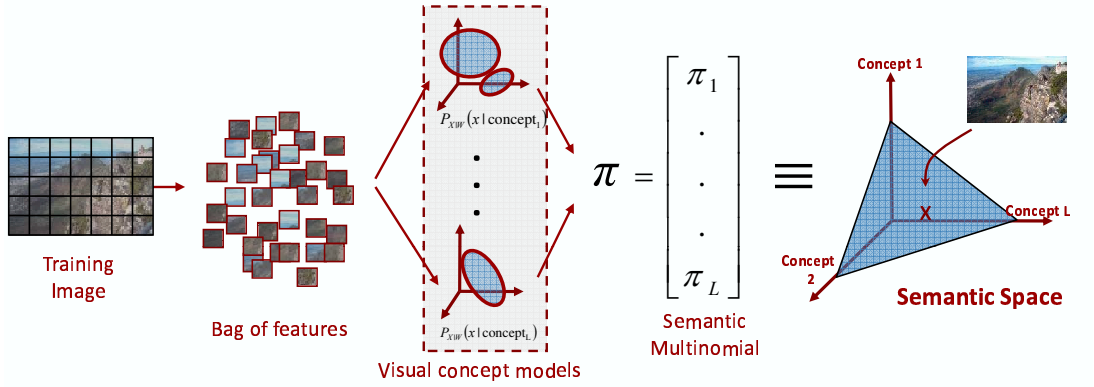

Figure 2.3: Image representation in semantic space $\mathcal{S}$, with a semantic multinomial (SMN) distribution. The SMN is a vector of posterior concept probabilities which encodes the co-occurrence of various concepts in the image, based on visual appearance.

variable $\mathbf{T}$ of parameters $\boldsymbol{\pi}_y = (\pi_y^1, \ldots, \pi_y^L)^T$

$$P_{\mathbf{T}|Y}(\mathcal{I}|y;\boldsymbol{\pi}_y) \quad = \quad \frac{n!}{\prod_{k=1}^{L} c_k!} \prod_{j=1}^{L} (\pi_y^j)^{c_j}, \tag{2.12}$$

where π_y^i is the probability that a feature vector is drawn from the i^{th} concept. The random variable $\mathbf{T}$ can be seen as the result of a feature transformation from the space of visual features $\mathcal{X}$ to the L-dimensional probability simplex $\mathcal{S}_L$. This mapping, $\boldsymbol{\Pi} : \mathcal{X} \rightarrow \mathcal{S}_L$ such that $\boldsymbol{\Pi}(\mathbf{X}) = \mathbf{T}$, maps the image $\mathcal{I} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, thereby the distribution $P_{\mathbf{X}|Y}(\mathcal{I}|y)$, into the multinomials $P_{\mathbf{T}|Y}(\mathcal{I}|y)$, and establishes a correspondence between images and points $\boldsymbol{\pi}_y \in \mathcal{S}_L$, as illustrated by Figure 2.3. We refer to each concept probability $\pi_y^i, i = 1, \ldots, L$ a *semantic feature* and the probability vector $\boldsymbol{\pi}_y$ as a *semantic multinomial* (SMN) distribution. The probability simplex $\mathcal{S}_L$ is itself referred to as the *semantic space* [119], which unlike $\mathcal{X}$ has explicit semantics. Semantic features, or concepts, outside the vocabulary simply define directions orthogonal to the learned semantic space. In the example of 1.1, the mapping of the image onto the semantic simplex assigns high probability to (known) concepts such as 'train', 'smoke', 'railroad' etc.

2.2.1 The Semantic Multinomial

Learning the semantic space requires an image database $\mathcal{D}$ and a vocabulary of semantic concepts, $\mathcal{L} = \{1, \ldots, L\}$, where each image is labeled with a label vector, $\mathbf{c}_d$ according to $\mathcal{L}$, making $\mathcal{D}^{\mathcal{L}} = \{(\mathcal{I}_1, \mathbf{c}_1), \ldots, (\mathcal{I}_D, \mathbf{c}_D)\}$. $\mathbf{c}_d$ is a binary L-dimensional vector such that $c_{d,i} = 1$ if the d^{th} image was annotated with the i^{th} keyword in $\mathcal{L}$. The dataset is said to be weakly labeled if absence of a keyword from caption $\mathbf{c}_d$ does not necessarily mean that the associated concept is not present in $\mathcal{I}_d$. For example, an image containing "sky" may not be explicitly labeled with that keyword. This is usually the case in practical scenarios, since each image is likely to be annotated with a small caption that only identifies the semantics deemed as most relevant to the labeler. We assume weak labeling throughout this work. Note that, the vocabulary of scene categories $\mathcal{W}$ can readily serve as a substitute for the vocabulary of semantic concepts $\mathcal{L}$. Infact, in absence of datasets annotated with semantic concepts, this is often the modus operandi to learn the semantic space. The only difference between the annotated datasets $\mathcal{D}^{\mathcal{W}}$ and $\mathcal{D}^{\mathcal{L}}$ is that in $\mathcal{D}^{\mathcal{W}}$ an image can be annotated with a single scene category (semantic concept) whereas in $\mathcal{D}^{\mathcal{L}}$ each image can be labeled with multiple concepts.

Given an annotated dataset $\mathcal{D}^{\mathcal{L}}$, appearance based concept models are learned for all the concepts in $\mathcal{L}$ similar to that of learning appearance models for the scene categories. Next, the posterior concept probabilities $P_{W|\mathbf{X}}(w|\mathbf{x}_k), w \in \{1, \ldots, L\}$ is computed for *each* feature vector $\mathbf{x}_k$, $k \in \{1, \ldots, N\}$, and $\mathbf{x}_k$ is assigned to the concept of largest probability. Denoting, c_w as the total count of feature vectors assigned to the w^{th} concept in a given image, the maximum likelihood estimate of the semantic feature π_w is then given by [33]

$$\pi_w^{ML} = \arg\max_{\pi_w} \prod_{j=1}^{L} \pi_j^{c_j} = \frac{c_w}{\sum_j c_j} = \frac{c_w}{N}. \tag{2.13}$$

The vector, $\pi^{ML} = \{\pi_1^{ML}, \ldots, \pi_L^{ML}\}$, is the ML estimate of the SMN for a given image.

2.2.2 Robust estimation of SMNs

As is usual in probability estimation, these posterior probabilities can be inaccurate for concepts with a small number of training images. Of particular concern are cases where some of the π_w are very close to zero, and can become ill-conditioned when used for recognition problems, where noisy estimates are amplified by ratios or logs of probabilities. A common solution is to introduce a prior distribution to regularize these parameters. Regularization can then be enforced by adopting a Bayesian parameter estimation viewpoint, where the parameter $\boldsymbol{\pi}$ is considered a random variable, and a prior distribution $P_{\mathbf{\Pi}}(\boldsymbol{\pi})$ introduced to favor parameter configurations that are, a priori, more likely.

Conjugate priors are frequently used, in Bayesian statistics [48], to estimate parameters of distributions in the exponential family, as is the case of the multinomial. They lead to a closed-form posterior (which is in the family of the prior), and *maximum aposteriori probability* parameter estimates which are intuitive. The conjugate prior of the multinomial is the Dirichlet distribution

$$\boldsymbol{\pi} \sim \mathbf{Dir}(\alpha) = \frac{\Gamma\left(\sum_j^L \alpha_j\right)}{\prod_{j=1}^L \Gamma(\alpha_j)} \prod_{j=1}^L \pi_j^{\alpha_j - 1}, \tag{2.14}$$

of *hyper-parameters* α_i, and where $\Gamma(.)$ is the Gamma function. Setting[2] $\alpha_i = \alpha$, the maximum aposteriori probability estimates are

$$\begin{aligned} \pi_w^{posterior} &= \arg\max_{\pi_w} P_{\mathbf{T}|\mathbf{\Pi}}(c_1, \ldots, c_L|\boldsymbol{\pi}) P_{\mathbf{\Pi}}(\boldsymbol{\pi}) \\ &= \arg\max_{\pi_w} \prod_{j=1}^L \pi_j^{c_j} \prod_{j=1}^L \pi_j^{\alpha-1} \\ &= \frac{c_w + \alpha - 1}{\sum_{j=1}^L (c_j + \alpha - 1)}. \end{aligned} \tag{2.15}$$

This is identical to the maximum likelihood estimates obtained from a sample where each count is augmented by $\alpha - 1$, i.e. where each image contains $\alpha - 1$ more feature vectors from each concept. The addition of these vectors prevents zero counts, regularizing $\boldsymbol{\pi}$. As α increases, the multinomial distribution tends to uniform.

[2] Different hyper-parameters could also be used for the different concepts.

Noting, from (2.13), that $c_w = N\pi_w^{ML}$, the regularized estimates of (2.15) can be written as

$$\pi_w^{posterior} = \frac{\pi_w^{ML} + \pi_0}{\sum_j^L (\pi_j^{ML} + \pi_0)}.$$

with $\pi_0 = \frac{\alpha - 1}{N}$.

2.2.3 SMNs as Posterior Probability Vector

The data processing theorem [88] advises against making hard decisions until the very last stages of processing. This suggests that thresholding the individual feature vector posteriors and counting is likely to produce worse probability estimates than those obtained without any thresholding. Motivated by the above argument, it is worth considering an alternative procedure for the estimation of π_w. Instead of (2.13), this consists of equating the semantic features π_w *directly with the posterior probability of the* w^{th} *semantic concept given the entire image*, i.e.

$$\pi_w^{direct} = P_{W|\mathbf{X}}(w|\mathcal{I}) \tag{2.16}$$

Thus, while in (2.13), posterior probability vector for each feature vector is threshold and aggregated over the entire image, in (2.16) the posterior probability vector is computed directly from the entire collection of the feature vectors. Thus, given an image $\mathcal{I} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ the vector of posterior probabilities

$$\boldsymbol{\pi}^{direct} = (P_{W|\mathbf{X}}(1|\mathcal{I}), \ldots, P_{W|\mathbf{X}}(L|\mathcal{I}))^T \tag{2.17}$$

provides a rich description of the image semantics and a robust alternative to the estimation of its SMN. Furthermore, regularized estimates of (2.17) can be obtained with

$$\pi_w^{reg} = \frac{\pi_w^{direct} + \pi_0}{1 + L\pi_0} \tag{2.18}$$

which is equivalent to using maximum aposteriori probability estimates, in the thresholding plus counting paradigm, with the Dirichlet prior of (2.14). In this work we rely on (2.18) to obtain a SMN of a given image.

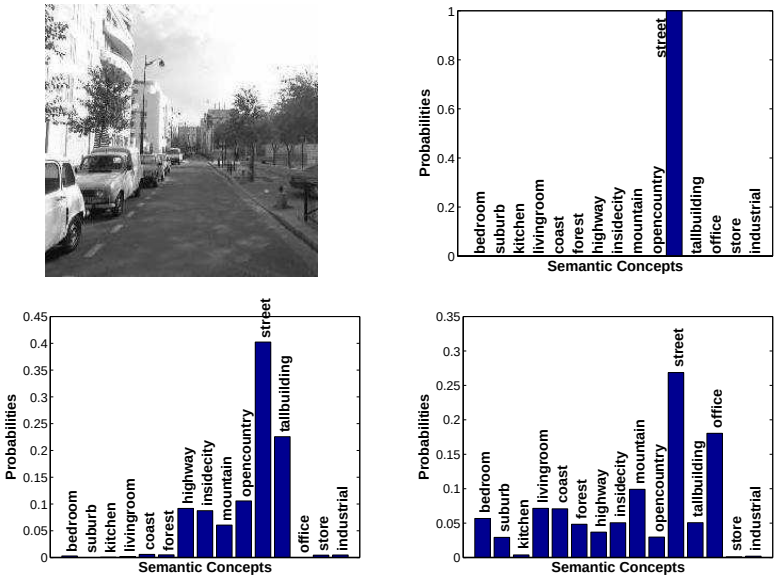

Figure 2.4: SMN for the image shown on the top left computed using (top-right) (2.8), (bottom-left) (2.21) and (bottom-right) (2.23).

2.3 Computing the Semantic Multinomial

It should be noted that the architecture proposed above is generic, in the sense that any appearance recognition system that produces a vector of posterior probabilities $\boldsymbol{\pi}$, can be used to learn the proposed contextual models. In fact, these probabilities can even be produced by systems that do not model appearance explicitly, e.g. multi-class logistic regression, multi-class SVM etc. This is achieved by converting classifier scores to a posterior probability distribution, using probability calibration techniques. For example, the distance from the decision hyperplane learned by an SVM can be converted to a posterior probability using a sigmoidal transform [110]. In practice, however, care must be taken to guarantee that the appearance classifiers are not too strong. If they make very hard decisions, e.g. assign images to a single class, the SMN would simply indicate the presence of a single concept and would not be rich enough to build visual recognition systems. Infact, in Chapter 5 we use multi-class logistic regression to compute the SMNs.

In the MPE implementation above, it is natural to use the posterior probabilities of (2.18) as the SMN of image $\mathcal{I}$. However, as N tends to be large, there

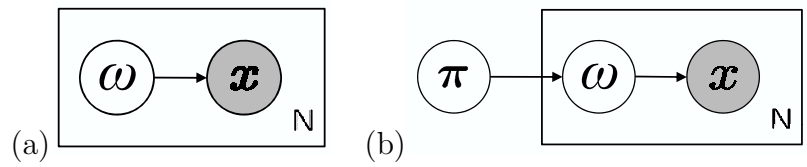

Figure 2.5: Alternative generative models for image formation at the appearance level. (a) A concept is sampled per appearance feature vector rather than per image, from $P_{\mathbf{X}|W}(\mathbf{x}|w)$. (b) Explicit modeling of the contextual variable Π from which a single SMN is drawn per image.

Table 2.1: SMN Entropy.

Model	Entropy
Figure 2.1, Eq (2.8)	0.003 ± 0.044
Figure 2.5(a), Eq (2.21)	2.530 ± 0.435
Figure 2.5(b), Eq (2.23)	2.546 ± 0.593

is usually very strong evidence in favor of one concept, not always that of greatest perceptual significance. For example, if the image has a large region of "sky", the existence of many sky patches makes the posterior probability of the "sky" concept close to one. This is illustrated in Figure 2.4 (top-right) where the SMN assigns all probability to a single concept. Table 2.1 shows that this happens frequently: the average entropy of the SMNs computed on the N15 Dataset (to be introduced later) is very close to 0. Note that this is the property that enables the learning of the appearance based models from the weakly supervised datasets: when all images containing "sky" are grouped, the overall feature distribution is very close to that of the "sky" concept, despite the fact that the training set contains spurious image patches from other concepts. This is an example of the multiple instance learning paradigm [155], where an image, consisting of some patches from the concept being modeled and some spurious patches from other concepts, serves as the positive bag. Although this dominance of the strongest concept is critical for learning, the data processing theorem advises against it during inference. Or, in other words, while multiple instance learning is required, multiple instance in-

ference is undesirable. In particular, modeling images as bags-of-features from a *single concept*, as in Figure 2.1, does not lend to contextual inference.

One alternative is to perform inference with the much looser model of Figure 2.5(a), where a concept is sampled *per appearance feature vector*, rather than per image. Note that, because labeling information is not available per vector, the models $P_{\mathbf{X}|W}(\mathbf{x}|w)$ are still learned as before, using the multiple instance learning principle. The only difference is the inference procedure. In this case, SMNs are available per image patch denoted as patch-SMN, $\boldsymbol{\pi}^n = P_{W|X}(w_n|x_n), n \in \{1, \ldots, N\}$. Determining an SMN, denoted the *Image-SMN*, for the entire image requires computing a representative for this set of patch-SMNs. One possibility is the multinomial of minimum average Kullback-Leibler divergence with all patch-SMNs

$$\boldsymbol{\pi}^* = \arg\min_{\boldsymbol{\pi}} \frac{1}{N} \sum_{n=1}^{N} KL(\boldsymbol{\pi}||\boldsymbol{\pi}^n) \quad \text{s.t} \sum_{i=1}^{L} \pi_i = 1. \tag{2.19}$$

As shown in Appendix C, this is the representative

$$\pi_i^* = \frac{\exp \frac{1}{N} \sum_n \log \pi_i^n}{\sum_i \exp \frac{1}{N} \sum_n \log \pi_i^n}, \tag{2.20}$$

which reduces to

$$\pi_i^* = \frac{\exp \left\{ \frac{1}{n} \sum_n \log P_{X|W}(x_n|i) \right\}}{\sum_j \exp \left\{ \frac{1}{n} \sum_n \log P_{X|W}(x_n|j) \right\}} \tag{2.21}$$

for a uniform prior. This is in contrast to the posterior estimate of (2.8). Note that while (2.8) computes a product of likelihoods, (2.21) computes their geometric mean.

A second possibility is to adopt the generative model of Figure 2.5(b). This explicitly accounts for the contextual variable $\boldsymbol{\Pi}$, from which a single SMN is drawn per image. A concept is then drawn per image patch. In this case, the Image-SMN is

$$\boldsymbol{\pi}^* = \arg\max_{\boldsymbol{\pi}} P_{\Pi|X}(\boldsymbol{\pi}|\mathcal{I}). \tag{2.22}$$

However, this optimization is intractable, and only approximate inference is possible. A number of approximations can be used, including Laplace or variational

approximations, sampling, etc. In Appendix D we show that, for a variational approximation,

$$\pi_i^* = \frac{\gamma_i - 1}{\sum_j \gamma_j - L} \tag{2.23}$$

where, γ_i is computed with the following iteration,

$$\gamma_i^* = \sum_n \phi_{ni} + \alpha_i \tag{2.24}$$

$$\phi_{ni}^* \propto P_{X|W}(x_n|w_n = i)\ e^{\psi(\gamma_i) - \psi(\sum_j \gamma_j)}. \tag{2.25}$$

Here, α_i is the parameter of the prior $P_\Pi(\pi)$ which, for compatibility with the assumption of uniform class priors, we set to 1, $\psi(\cdot)$ the Digamma function, and $\gamma_i,\ \phi_{ni}$ the parameters of the variational distributions. Figure 2.4 shows that the SMNs obtained with (2.21) and (2.23) are rich in contextual information. Table 2.1 shows that the two models lead to approximately the same average SMN entropy on N15, which is much higher than that of (2.8).

Since (2.23) involves an iterative procedure, which is more expensive than the closed form of (2.21), (2.21) is the default choice for computing the SMNs in this work. In Chapter 6 we will show that (2.21) also yield marginally better performance over (2.23), in a scene classification task.

2.4 Related Work

The idea of representing documents as weighted combinations of the words in a pre-defined vocabulary is commonly used in information retrieval. In fact, the classic model for information retrieval is the vector space model of Salton [125, 126]. Under this model, documents are represented as collections of keywords, weighted by importance, and can be interpreted as points in the semantic space spanned by the vocabulary entries. In image retrieval, there have been some proposals to represent images as points in a semantic vector space. The earliest among these efforts [68, 54] were based on semantic information extracted from metadata - viz. origin, filename, image url, keywords from surrounding webpage text, manual annotations, etc.

The closest works, in the literature, to the semantic image representation proposed here, are the systems proposed by Smith *et al.* in [137, 135] and Lu *et al.* in [86]. To the best of our knowledge, [137] pioneered the idea of learning a semantic space by learning a separate statistical model for each concept. The vector of semantic weights, denoted as the 'model vector', is learned from the image content. Each image receives a confidence score per semantic concept, based on the proximity of the image to the decision boundary of a support vector machine (SVM) trained to recognize the concept. While laying the foundations for the semantic image representation, [137] does not present any formal definition or systematic analysis of the semantic image representation, as presented in Section 2.2. Moreover in [137], the model vector is used solely for the task of retrieving images known to the system (that were used to learn the SVM classifiers). In Chapter 3 we show that the benefits of semantic representation goes beyond that, and propose image retrieval systems that can generalize well beyond the known vocabulary. Furthermore, we present two novel visual recognition systems, viz. scene classification and cross-modal multimedia retrieval based on the semantic image representation. Infact, the problem of cross-modal multimedia is itself in its nascency and no formal analysis has been presented in the literature, which we do in Chapter 5. Finally, in [137] the model vector is simply used as an alternative image representation, without any analysis of their ability to model semantic "gist" and context of an image. In Chapter 6 we introduce "contextual models" and show that the proposed representation is successful in modeling the "gist" of an image.

2.5 Acknowledgments

The text of Chapter 2, in part, is based on the material as it appears in: N. Rasiwasia, P. J. Moreno and N. Vasconcelos, *'Bridging the Semantic Gap: Query by Semantic Example'*, IEEE Transactions on Multimedia, 9(5), 923-938, August 2007. and N. Rasiwasia, P. J. Moreno and N. Vasconcelos, *'Query by Semantic Example'*, ACM International Conference on Image and Video Retrieval, LNCS

51-60, Phoenix, 2006. The dissertation author was a primary researcher and an author of the cited material.