Variational Bayesian Methods For Multimedia Problems

Zhaofu Chen, S. Derin Babacan, *Member, IEEE*, Rafael Molina, *Member, IEEE*, and Aggelos K. Katsaggelos, *Fellow, IEEE*

***Abstract*—In this paper we present an introduction to Variational Bayesian (VB) methods in the context of probabilistic graphical models, and discuss their application in multimedia related problems. VB is a family of deterministic probability distribution approximation procedures that offer distinct advantages over alternative approaches based on stochastic sampling and those providing only point estimates. VB inference is flexible to be applied in different practical problems, yet is broad enough to subsume as its special cases several alternative inference approaches including Maximum A Posteriori (MAP) and the Expectation-Maximization (EM) algorithm. In this paper we also show the connections between VB and other posterior approximation methods such as the marginalization-based Loopy Belief Propagation (LBP) and the Expectation Propagation (EP) algorithms. Specifically, both VB and EP are variational methods that minimize functionals based on the Kullback-Leibler (KL) divergence. LBP, traditionally developed using graphical models, can also be viewed as a VB inference procedure. We present several multimedia related applications illustrating the use and effectiveness of the VB algorithms discussed herein. We hope that by reading this tutorial the readers will obtain a general understanding of Bayesian methods and establish connections among popular algorithms used in practice.**

***Index Terms*—Bayes methods, graphical models, multimedia signal processing, variational Bayes, inverse problems.**

I. Introduction

A GOOD part of the research and applications covered by the IEEE Transactions on Multimedia deal with inverse problems, that is, moving from known events back to their most probable causes. Although solutions to inverse problems have been originally derived using numerous approaches, many of them can be developed and formulated in a systematic fashion within the Bayesian framework.

Multimedia data processing tasks have made extensive use of probabilistic machine learning models in domains such as content-based image and video retrieval, biometrics, semantic labeling, human-computer interaction, and data mining in text and music documents (see for instance [1]–[3]). Multimedia data, such as digital images, audio streams, motion video programs, etc., exhibit much richer structures than simple, isolated data items. Probabilistic machine learning techniques can explicitly exploit the spatial and temporal structures, and model the correlations among different elements of the inverse problems.

Among the wide range of multimedia related applications of diverse origins, recently there has been a significant interest in problems involving the estimation of low-rank matrices. A typical example is the matrix completion problem, where an unknown (approximately) low-rank matrix is estimated from its limited set of observed entries. Matrix completion finds application in many areas of engineering, including computer vision [4], [5], medical imaging [6], machine learning [7], system identification [8], sensor networks [9], video compression [10], image denoising [11], and video error concealment [12] (see [13] and the references therein). A related and important problem is Robust Principal Component Analysis (RPCA), where the high dimensional data is assumed to lie in a lower dimensional subspace with some data points corrupted with (arbitrarily) large errors. Widely used classical methods, such as Principal Component Analysis (PCA), often fail to provide meaningful results in these cases. Robust PCA has many important multimedia related applications, such as video surveillance (foreground/background separation in video) [13], face recognition [14], [15], latent semantic indexing [16], image alignment [17], voice separation [18], error concealment [19], motion segmentation [20] and network monitoring [21] among many others.

All of the problems above can be approached by using Bayesian modeling and inference. A fundamental principle of the Bayesian philosophy is to regard all parameters and unobservable variables of a given problem as unknown stochastic quantities, assigning probability distributions based on beliefs. Thus, for instance, in the notationally simple image recovery problem, the original image(s), the observation noise, and even the function(s) defining the acquisition process are all treated as samples of random variables, with corresponding prior Probability Density Functions (PDFs) that model our knowledge about the nature of images and the imaging process.

The recently developed VB methods have attracted a lot of interest in Bayesian statistics, machine learning and related

Manuscript received July 03, 2013; revised December 16, 2013; accepted February 18, 2014. Date of publication February 24, 2014; date of current version May 13, 2014. This work was supported in part by a grant from the Department of Energy (DE-NA0000457), the Spanish Ministry of Economy and Competitiveness under project TIN2010-15137, the European Regional Development Fund (FEDER), and the CEI BioTic at the Universidad de Granada. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yiannis Andreopoulos.

Z. Chen and A. K. Katsaggelos are with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60208 USA (e-mail: zhaofuchen2014@u.northwestern.edu; aggk@eecs.northwestern.edu).

S. D. Babacan is with Google Inc., Google Inc., Mountain View, CA 94043 USA (e-mail: dbabacan@gmail.com).

R. Molina is with the Departamento de Ciencias de la Computación e I. A. E.T.S. Ing. Informática y Telecomunicación. Universidad de Granada, 18071 Granada, Spain (e-mail: rms@decsai.ugr.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2014.2307692

areas. A major disadvantage of traditional methods (such as EM) is that they generally require exact knowledge of the posterior distributions of the unknowns, or poor approximations of them are used. Variational Bayesian methods [22]–[27] overcome this limitation by approximating the unknown posterior distributions with simpler, analytically tractable distributions, which allow for the computation of the needed expectations, and therefore extend the applicability of Bayesian inference to a much wider range of modeling options: More complex priors (which are very often needed in multimedia problems) modeling the unknowns can be utilized with ease, resulting in improved estimation accuracy.

It is highly advantageous to augment the aforementioned probabilistic framework with diagrammatic representations of probability distributions called probabilistic graphical models, since they provide a simple way to visualize the structure of the probabilistic model. Furthermore, the required inference can be expressed in terms of graphical manipulations.

In this paper, we provide an overview of Bayesian modeling and inference methods for multimedia and related areas based on the use of graphical models. Emphasis will be placed on the pros and cons of variational posterior distribution approximations, and their connections to other inference methods.

The rest of the paper is structured as follows. In Section II we provide a preliminary introduction to Bayesian modeling and graphical models. The inference of the unknown quantities within the Bayesian framework is treated in Section III, with the focus on variational methods and their relations to alternative approaches. Section IV discusses local bounds on probability distributions and their application in facilitating variational analysis. The concepts of factor graphs and LBP are introduced in Section V and its connection with variational inference is analyzed. In Section VI we present the EP algorithm as an alternative variational approach to inference. Finally, the paper is concluded in Section VII. In addition to the main text, we include at the end of the paper four appendices, which exemplify the variational analysis applied to solve multimedia problems.

Notation: We use lowercase boldface letters to denote vectors or sets of items, whose specific meanings will be clear from the context. Matrices are in general represented by uppercase boldface letters, unless otherwise noted. $\mathrm{Tr}(\cdot)$ is the trace operator on a square matrix. Given a matrix $\mathbf{X}$, we denote as $\mathbf{x}_{i\cdot}$, $\mathbf{x}_{\cdot j}$ and X_{ij} its ith row, jth column, and (i,j)th element, respectively.

II. Bayesian Modeling

A. Notations and Preliminaries

As mentioned above, in many multimedia applications, the solution to an inverse problem is sought. In general, the underlying system generates a set of observed variables, and the goal is to infer a set of latent/hidden variables from these observations. For instance, in blind deconvolution for image recovery, the camera provides a blurred and noisy version of the scene, and an estimate of the original sharp and noiseless image is desired. As another example, in audio-visual speech recognition multiple related data streams of different modalities are used simultaneously to recognize the uttered words [28], [29].

To facilitate the exposition below, we introduce a unified set of notations for the inverse problem as follows. Let $\mathbf{y} = \{\mathbf{y}_i\}_{i=1}^N$ denote the set of N observed variables, where each $\mathbf{y}_i$ is in general a vector (or a scalar as a degenerated vector). Examples of $\mathbf{y}_i$ can be vectorized image frames in image processing, binary class labels for classification, recorded speech sequence for speech recognition, etc. Note that two observed variables, e.g., $\mathbf{y}_i$ and $\mathbf{y}_j$ do not have to be of the same size, although in many practical scenarios they do. In addition to the observed variables, a set of hidden variables are denoted by $\mathbf{z}$, which can be considered as driving the data generation process. Other parameters, such as the observation noise variance, that affect the modeling are grouped and denoted as $\mathbf{\Omega}$.

Since $\mathbf{y}$ is assumed to be stochastic, its probability conditioned on the hidden variables $\mathbf{z}$ and parameters $\mathbf{\Omega}$ is known as the data likelihood $\mathrm{p}(\mathbf{y}|\mathbf{z},\mathbf{\Omega})$. If the data $\mathbf{y}$ are independently observed, it follows that

$$\mathrm{p}(\mathbf{y}|\mathbf{z},\mathbf{\Omega}) = \prod_{i=1}^{N} \mathrm{p}(\mathbf{y}_i|\mathbf{z},\mathbf{\Omega}). \tag{1}$$

The prior distribution $\mathrm{p}(\mathbf{z}|\mathbf{\Omega})$ is employed to model our knowledge about the hidden variables $\mathbf{z}$ prior to seeing the observation $\mathbf{y}$, and

$$\mathbf{\Theta} = \{\mathbf{z},\mathbf{\Omega}\} \tag{2}$$

is the set of all unknown variables. Note that $\mathbf{\Omega}$ can be treated as either deterministic or stochastic. In a fully Bayesian model, $\mathbf{\Omega}$ is treated as stochastic and it is assigned a hyperprior distribution $\mathrm{p}(\mathbf{\Omega})$.

As a concrete example of the notations introduced above, let us consider a simple linear logistic regression problem. In this case, the set of D-dimensional feature vectors $\mathbf{x} = \{\mathbf{x}_i\}_{i=1}^N$ are assumed to be fixed. The observed data are the associated binary class labels and are denoted as $\mathbf{y} = \{y_i\}_{i=1}^N$. The logistic regression expresses the class likelihood $\mathrm{p}(y_i|\mathbf{z})$ as a function of the weight vector $\mathbf{z} = [z_1,\cdots,z_D]^{\mathrm{T}}$, i.e.,

$$\mathrm{p}(y_i|\mathbf{z}) = \begin{cases} \sigma(\mathbf{z}^{\mathrm{T}}\mathbf{x}_i), & y_i = 1 \\ \sigma(-\mathbf{z}^{\mathrm{T}}\mathbf{x}_i), & y_i = 0 \end{cases}, \tag{3}$$

where $\sigma(\cdot)$ is the logistic sigmoid function. Under the assumption that the data are independent of each other, the complete observation model can then be expressed using (1) and (3) by

$$\mathrm{p}(\mathbf{y}|\mathbf{z}) = \prod_{i=1}^{N} \exp(\mathbf{z}^{\mathrm{T}}\mathbf{x}_i y_i)\sigma(-\mathbf{z}^{\mathrm{T}}\mathbf{x}_i). \tag{4}$$

To estimate $\mathbf{z}$, we use the prior distribution $\mathrm{p}(\mathbf{z}|\alpha)$ based on the l_p-quasinorm

$$\mathrm{p}(\mathbf{z}|\alpha) \propto \alpha^{\frac{D}{p}} \exp\left[-\alpha\sum_{i=1}^{D}|z_i|^p\right] = \alpha^{\frac{D}{p}} \prod_{i=1}^{D} \exp\left(-\alpha|z_i|^p\right) \tag{5}$$

where $\alpha > 0$ and $0 < p \leq 1$. This type of prior has been shown to enforce sparsity in estimation problems like logistic regression (see [30] and [31] for a regularization point of view). Finally, we assume that α has a Gamma hyperprior $\mathrm{p}(\alpha) = \Gamma(\alpha|a_\alpha^o, b_\alpha^o)$, where the hyperparameters $a_\alpha^o > 0$ and $b_\alpha^o > 0$ are assumed to be deterministic and fixed. In this classification example, $\mathbf{\Omega} = \{\alpha\}$, and $\mathbf{\Theta} = \{\mathbf{z},\alpha\}$, respectively.

With the definition above, the global modeling of the inverse problem can be expressed as the joint distribution

$$\mathrm{p}(\mathbf{z},\boldsymbol{\Omega},\mathbf{y}) = \mathrm{p}(\mathbf{y}|\mathbf{z},\boldsymbol{\Omega})\mathrm{p}(\mathbf{z}|\boldsymbol{\Omega})\mathrm{p}(\boldsymbol{\Omega}). \tag{6}$$

The objective of Bayesian analysis in general is to infer the unknown $\boldsymbol{\Theta}$ given the observed $\mathbf{y}$.

B. Graphical Models

Effective estimation of the unknown variables $\boldsymbol{\Theta} = \{\mathbf{z},\boldsymbol{\Omega}\}$ is possible only through the utilization of models that accurately represent the nature of these variables and their relationships. In this regard, graphical models provide a systematic method for expressing the relationship between the unknown and observed quantities, which is embedded in the structure of a graph. Graphical models are especially useful for incorporating probabilistic conditional independencies into the modeling and estimation procedures [32]. In the following paragraphs we introduce two types of graphical models, namely directed graphical models and undirected graphical models, and describe their applications in Bayesian analysis.

A directed graphical model, also called a Bayesian network [33] represents a joint probability distribution as the product of conditional distributions in the form of a Directed Acyclic Graph (DAG):

$$\mathrm{p}(\mathbf{s}) = \prod_i \mathrm{p}(s_i|\mathbf{s}_{\pi(i)}) \tag{7}$$

where $\mathbf{s} = \{s_i\}_i$ is the set of all random variables involved in the joint distribution, and $\mathbf{s}_{\pi(i)} \subseteq \mathbf{s}$ represents the set of s_i's parents. Although we use non-boldface lowercase letters s_i to denote the random variables involved, the reader is advised that they are not restricted to be scalar-valued and can represent general random quantities of arbitrary dimensions. In the inverse problem introduced above, $\mathbf{s}$ usually corresponds to $\{\mathbf{z},\boldsymbol{\Omega},\mathbf{y}\}$, or to $\{\mathbf{z},\boldsymbol{\Omega}\}$ when $\mathbf{y}$ is considered observed and fixed.

To visualize the factorization in (7) as a directed graph we add one node per s_i and draw to s_i a directed link from each $s_j \in \mathbf{s}_{\pi(i)}$. Conversely, to obtain the probability distribution factorization from its graphical representation, we introduce a factor for every node in the graph. If the node s_i has no parents the factor is simply $\mathrm{p}(s_i)$, otherwise it is $\mathrm{p}(s_i|\mathbf{s}_{\pi(i)})$ where the parents $\mathbf{s}_{\pi(i)}$ are all the nodes that point to s_i.

As two illustrative examples, Fig. 1 depicts the DAG corresponding to the distribution

$$\mathrm{p}(s_1,s_2,s_3,s_4) = \mathrm{p}(s_1)\mathrm{p}(s_2)\mathrm{p}(s_3|s_1,s_2)\mathrm{p}(s_4|s_3) \tag{8}$$

and Fig. 2 depicts the DAG corresponding to the Hidden Markov Model (HMM)

$$\begin{aligned}&\mathrm{p}(z_1,z_2,z_3,z_4,y_1,y_2,y_3,y_4)\\&= \mathrm{p}(z_1)\prod_{i=2}^{4}\mathrm{p}(z_i|z_{i-1})\prod_{i=1}^{4}\mathrm{p}(y_i|z_i).\end{aligned} \tag{9}$$

A directed graphical model (or its equivalent probability factorization) implies a set of independence and conditional independence relations between the variables, see [32] for details.

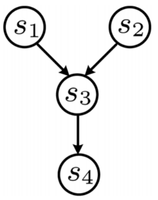


Fig. 1. DAG representing $\mathrm{p}(s_1)\mathrm{p}(s_2)\mathrm{p}(s_3|s_1,s_2)\mathrm{p}(s_4|s_3)$.

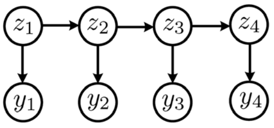


Fig. 2. DAG representing $\mathrm{p}(z_1)\prod_{i=2}^{4}\mathrm{p}(z_i|z_{i-1})\prod_{i=1}^{4}\mathrm{p}(y_i|z_i)$.

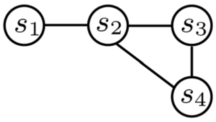


Fig. 3. Undirected graph representing $\frac{1}{Z}f_{12}(s_1,s_2)f_{234}(s_2,s_3,s_4)$ and $\frac{1}{Z}f_{12}(s_1,s_2)f_{23}(s_2,s_3)f_{34}(s_3,s_4)f_{24}(s_2,s_4)$.

Consequently we can think about the structure of a joint distribution in three different ways, that is, its factorization, its directed graphical model or the conditional independence relations. There is a one-to-one mapping between factorizations and directed acyclic graphs. Unfortunately there are conditional independence relations that cannot be represented by directed acyclic graphs.

Let us return to the inverse problem discussed above. For the joint probability distribution in (6), the associated directed graphical model can be constructed in a straightforward way. Note, however, that the relationship between the parameters, hidden variables and observed variables leads in general to much richer representations. While in our classification problem $\mathrm{p}(\mathbf{z}|\boldsymbol{\Omega})$ in (5) and $\mathrm{p}(\mathbf{y}|\mathbf{z},\boldsymbol{\Omega})$ in (4) are products of independent distributions, in many real world problems some of the conditional distributions themselves may also be modeled naturally using directed or very often using undirected graphical models, as we describe next.

Let $\mathbf{s}$ be a set of random variables whose distribution takes the form of the product of positive potential functions $\{f_m(\mathbf{s}_m)\}_m$, where each $f_m(\mathbf{s}_m)$ operates on a subset of $\mathbf{s}$ called a clique and denoted as $\mathbf{s}_m \subseteq \mathbf{s}$. The joint distribution can therefore be expressed as

$$\mathrm{p}(\mathbf{s}) = \frac{1}{Z}\prod_m f_m(\mathbf{s}_m), \tag{10}$$

where Z is the normalization constant that makes the probability distribution integrate to 1. To visualize the associated undirected graphical model, also called Markov Random Field (MRF), we draw one node per random variable and for every clique $\mathbf{s}_m$ we draw an undirected link between every pair of nodes in it.

Fig. 3 depicts the undirected graph corresponding to the distribution

$$\mathrm{p}(s_1,s_2,s_3,s_4)=\frac{1}{Z}f_{12}(s_1,s_2)f_{23}(s_2,s_3)f_{34}(s_3,s_4)f_{24}(s_2,s_4) \tag{11}$$

which is also the representation of

$$\mathrm{p}(s_1, s_2, s_3, s_4) = \frac{1}{Z} f_{12}(s_1, s_2) f_{234}(s_2, s_3, s_4). \tag{12}$$

Notice that, for clarity, we have replaced m in (10) by the indices of the variables in the corresponding potential function.

Given an undirected graph we build a probability factorization by adding one term to the factorization per maximal clique (a subset of nodes which are fully connected and no additional node can be added to the subset so that the subset remains fully connected).

Notice that, as is the case for DAGs, the undirected graphical model (or the probability factorization) implies a set of independence and conditional independence relations among the variables [32]. Consequently we can think about the structure of a joint distribution as its factorization, its undirected graphical model or the conditional independence relations.

Let us complete this section by considering a simple image denoising problem. Our goal is to recover the original image pixels $\mathbf{z} = \{z_{i,j}\}_{i,j}$ from the observed noisy image

$$\mathbf{y} = \mathbf{z} + \epsilon, \tag{13}$$

where $\mathrm{p}(\epsilon|\beta) = \mathcal{N}(\epsilon|\mathbf{0}, \beta^{-1}\mathbf{I})$. This gives rise to the data likelihood

$$\mathrm{p}(\mathbf{y}|\mathbf{z}, \beta) = \mathcal{N}(\mathbf{y}|\mathbf{z}, \beta^{-1}\mathbf{I}). \tag{14}$$

For the noiseless image $\mathbf{z}$ we assume a prior based on the Conditional Auto-Regressive (CAR) model, that is,

$$\mathrm{p}(\mathbf{z}|\alpha) \propto \alpha^{\frac{N}{2}} \exp\left\{-\frac{\alpha}{2}\sum_{i,j}\left[(z_{i,j} - z_{i+1,j})^2 + (z_{i,j} - z_{i,j+1})^2\right]\right\}, \tag{15}$$

which captures the smoothness property of natural images according to which the intensities of neighboring pixels are expected to be close to each other. The parameter α controls the level of smoothness imposed by the prior, e.g., a larger α imposes a heavier penalty to image discontinuities and hence promotes smoothness in a stronger way. Note α plays a similar role as the regularization parameter in regularized optimization problems.

The Bayesian modeling of our problem has the form in (6) with $\mathbf{\Omega} = (\alpha, \beta)$. Notice that $\mathrm{p}(\mathbf{z}|\mathbf{\Omega}) = \mathrm{p}(\mathbf{z}|\alpha)$ corresponds to an undirected graphical model. Furthermore

$$\mathrm{p}(\mathbf{z}|\mathbf{y}, \mathbf{\Omega}) \propto \mathrm{p}(\mathbf{z}, \mathbf{\Omega}, \mathbf{y}) = \mathrm{p}(\mathbf{y}|\mathbf{z}, \beta)\mathrm{p}(\mathbf{z}|\alpha) \propto \prod_{i,j} \phi(z_{i,j})\psi(z_{i,j}, z_{i+1,j})\psi(z_{i,j}, z_{i,j+1}) \tag{16}$$

is an MRF known as conditional random field, where $\phi(z_{i,j}) = \exp[-\frac{\beta}{2}(y_{i,j} - z_{i,j})^2]$ and $\psi(a, b) = \exp[-\frac{\alpha}{2}(a - b)^2]$.

III. Bayesian Inference

Recall from the description above that $\mathbf{\Theta} = \{\mathbf{z}, \mathbf{\Omega}\}$ denotes the set of all unknown variables (i.e., hidden variables and unknown parameters). In the Bayesian framework, the inference on the unknowns is performed using the posterior distribution $\mathrm{p}(\mathbf{\Theta}|\mathbf{y})$, expressed using Bayes' rule as

$$\mathrm{p}(\mathbf{\Theta}|\mathbf{y}) = \frac{\mathrm{p}(\mathbf{\Theta}, \mathbf{y})}{\mathrm{p}(\mathbf{y})} = \frac{\mathrm{p}(\mathbf{y}|\mathbf{\Theta})\mathrm{p}(\mathbf{\Theta})}{\mathrm{p}(\mathbf{y})}, \tag{17}$$

where $\mathrm{p}(\mathbf{y}) = \int \mathrm{p}(\mathbf{y}|\mathbf{\Theta})\mathrm{p}(\mathbf{\Theta})\mathrm{d}\mathbf{\Theta}$ is called the model evidence.

In many applications the posterior $\mathrm{p}(\mathbf{\Theta}|\mathbf{y})$ is intractable since $\mathrm{p}(\mathbf{y})$ cannot be computed analytically. In these situations, one has to resort to approximation methods. In the following we will briefly review the most common ones.

Stochastic sampling methods such as Markov Chain Monte Carlo (MCMC) represent the most general approaches to performing inference. These methods generate a sequence of samples from the intractable posterior distribution using tractable conditional or joint distributions. These samples are then used to approximate the intractable posterior distribution. In theory, sampling methods can find the exact form of the posterior distribution, but in practice they are computationally intensive (especially for multidimensional signals such as images and videos) and their convergence is hard to establish. Among the MCMC methods, the Gibbs sampler is probably the best known one for fitting a Bayesian model [34], which aims at obtaining a sufficient number of samples from the posterior distribution to accurately characterize it. The assessment of the convergence of the chain and the compulsory use of burn-in iterations are two important problems to be faced, especially in complex models applied to large data sets, which is the case of multimedia applications. See [35] for a comparison of sampling and variational methods in the context of political analysis.

In addition to sampling-based approaches, most methods in the literature seek point estimates of the unknowns, which are generally obtained by maximizing the posterior distribution

$$\hat{\mathbf{\Theta}} = \arg\max_{\mathbf{\Theta}} \mathrm{p}(\mathbf{\Theta}|\mathbf{y}) = \arg\max_{\mathbf{\Theta}} \mathrm{p}(\mathbf{y}, \mathbf{\Theta}) \tag{18}$$

MAP solutions, or Maximum Likelihood (ML) solutions when flat priors are used, fall in this category. These methods reduce the inference problem to an optimization problem. From the deterministic perspective, these methods can be considered as regularized data fitting problems, which are extensively studied in the literature. Methods providing point estimates to the unknowns are computationally very efficient, but they might exhibit in some cases a number of disadvantages. Common problems include over-fitting in the presence of high noise, error propagation among the estimates of different unknowns, and lack of uncertainties of the estimates. Another fundamental problem with the MAP method is that the maximum might not represent the data well in some cases. For instance, in multimodal data, some of the modes might have very high magnitudes but very limited support, while most data points are represented by other modes with much larger support but lower peaks. In this case, the MAP method will provide the largest mode which has almost no representation of the data. In extreme cases, these methods might result in a trivial global maximum [36]. Bayesian methods, on the other hand, seek the full posterior distributions of the data, from which not only representative statistics such as means but also uncertainty information associated with the estimates can be obtained.

Let us illustrate this problem of MAP estimation with a simple example. Assume that the observed signal is the convolution of a sparse signal with a known blurring kernel plus additive Gaussian noise with known variance β^{-1}. The observation model can then be written as

$$\mathrm{p}(\mathbf{y}|\mathbf{z}) = \mathcal{N}(\mathbf{y}|\mathbf{H}\mathbf{z}, \beta^{-1}\mathbf{I}), \tag{19}$$

where $\mathbf{H}$ is the matrix constructed from the known blurring kernel.

To promote sparsity we use the following hierarchical model as in [37]

$$\mathrm{p}(\mathbf{z}|\boldsymbol{\xi}) = \prod_i \mathrm{p}(z_i|\xi_i) = \prod_i \mathcal{N}(z_i|0, \xi_i^{-1}) \tag{20}$$

$$\mathrm{p}(\boldsymbol{\xi}) = \prod_i \mathrm{p}(\xi_i) \propto \prod_i \xi^{-1} \tag{21}$$

Then the MAP approach would lead to

$$(\hat{\mathbf{z}}, \hat{\boldsymbol{\xi}}) = \arg\min_{\mathbf{z},\boldsymbol{\xi}} \frac{\beta}{2}\|\mathbf{y} - \mathbf{H}\mathbf{z}\|^2 + \frac{1}{2}\sum_i \xi_i z_i^2 + \sum_i \ln \xi_i \tag{22}$$

which produces a trivial solution when all ξ_i's are equal to zero. This is not alleviated by integrating over $\boldsymbol{\xi}$ since

$$-2\ln \mathrm{p}(\mathbf{y}, \mathbf{z}) = \beta\|\mathbf{y} - \mathbf{H}\mathbf{z}\|^2 + \sum_i \ln|z_i| \tag{23}$$

also leads to a trivial solution. This behavior of MAP has also been reported in problems such as blind image restoration [36] and centralized and distributed processing [38].

The Bayesian framework also provides other methodologies for estimating the distributions of the unknowns (i.e., more than merely point estimates), which provide more information about the uncertainties. A commonly used method is marginalization, where some of the unknowns are integrated/summed out from the joint distribution to obtain a marginal distribution, and the remaining unknowns are estimated by maximizing this distribution. Evidence-based (or Laplace) and empirical approaches fall into this category, where the marginalized variables are sometimes called hidden variables. The EM algorithm, first described in [39], is a very popular method in signal processing for iteratively solving ML and MAP problems that include hidden variables, and it is also based on the marginalization principle. In all methods based on marginalization, the prior distributions of the hidden variables are chosen such that the marginalization is tractable. In most cases, however, this limits the form of the prior models to simple or even unrealistic ones.

As mentioned above, one would like to use realistic models for the unknowns $\boldsymbol{\Theta}$ and meanwhile have an efficient inference procedure. However, arbitrarily complex models render fully Bayesian treatment impossible in most cases, and limit the inference options to point-estimation methods such as MAP or sampling approaches. Variational Bayes is a powerful alternative to these methods, as it provides more accurate approximations to the posterior distribution than point estimation methods, and is computationally much more efficient than sampling approaches. In the following, we will present a general outline of the variational methods.

VB methods provide analytically tractable approximations $\mathrm{q}(\boldsymbol{\Theta})$ to the true posterior distribution $\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$ by assuming $\mathrm{q}(\boldsymbol{\Theta})$ has specific parametric or factorized forms. In the following, we show how this approximating distribution $\mathrm{q}(\boldsymbol{\Theta})$ can be found. Let us first consider the following decomposition of the logarithm of the model evidence (see, e.g., [24] for a reference)

$$\begin{aligned}\ln \mathrm{p}(\mathbf{y}) &= \int \mathrm{q}(\boldsymbol{\Theta}) \ln\left(\frac{\mathrm{p}(\mathbf{y})\mathrm{q}(\boldsymbol{\Theta})}{\mathrm{q}(\boldsymbol{\Theta})}\right) d\boldsymbol{\Theta} \\ &= \int \mathrm{q}(\boldsymbol{\Theta}) \ln\left(\frac{\mathrm{p}(\boldsymbol{\Theta}, \mathbf{y})}{\mathrm{q}(\boldsymbol{\Theta})} \times \frac{\mathrm{q}(\boldsymbol{\Theta})}{\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})}\right) d\boldsymbol{\Theta} \\ &= \mathcal{L}(\mathrm{q}(\boldsymbol{\Theta})) + \mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})),\end{aligned} \tag{24}$$

where

$$\mathcal{L}(\mathrm{q}(\boldsymbol{\Theta})) = \int \mathrm{q}(\boldsymbol{\Theta}) \ln\left(\frac{\mathrm{p}(\boldsymbol{\Theta}, \mathbf{y})}{\mathrm{q}(\boldsymbol{\Theta})}\right) d\boldsymbol{\Theta}, \tag{25}$$

and

$$\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})) = \int \mathrm{q}(\boldsymbol{\Theta}) \ln\left(\frac{\mathrm{q}(\boldsymbol{\Theta})}{\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})}\right) d\boldsymbol{\Theta} \tag{26}$$

is the (reverse) Kullback-Leibler (KL) divergence between $\mathrm{q}(\boldsymbol{\Theta})$ and the true posterior $\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$.

Since $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})) \geq 0$ with equality if and only if $\mathrm{q}(\boldsymbol{\Theta}) = \mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$, we have that $\mathcal{L}(\mathrm{q}(\boldsymbol{\Theta})) \leq \ln \mathrm{p}(\mathbf{y})$. So, for any distribution $\mathrm{q}(\boldsymbol{\Theta})$, the quantity $\mathcal{L}(\mathrm{q}(\boldsymbol{\Theta}))$ represents a lower bound of $\ln \mathrm{p}(\mathbf{y})$. We can then maximize this lower bound with respect to $\mathrm{q}(\boldsymbol{\Theta})$ to obtain an approximation of $\ln \mathrm{p}(\mathbf{y})$.

Equivalently, we can minimize $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y}))$, which represents a variational problem. Variational methods have their origins in the 18th century with the work of Euler, Lagrange, and others on the calculus of variations. To obtain some insight into the kind of VB distributions that minimization of $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y}))$ provides, consider the following cases. If $\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$ is small in a given area, then $-\ln \mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$ is large and so $\mathrm{q}(\boldsymbol{\Theta})$ will assign low mass to that area to avoid large values of the KL divergence. In particular, minimizing $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y}))$ will lead to distributions $\mathrm{q}(\boldsymbol{\Theta})$ that assign zero probability mass to areas outside the support of $\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$, hence $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y}))$ is "zero-forcing" for $\mathrm{q}(\boldsymbol{\Theta})$. Notice that other functionals can also be used to measure the similarity between $\mathrm{q}(\boldsymbol{\Theta})$ and $\mathrm{p}(\Theta|\mathbf{y})$. In particular, we can minimize $\mathrm{KL}(\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})\|\mathrm{q}(\boldsymbol{\Theta}))$, which leads to a $\mathrm{q}(\boldsymbol{\Theta})$ that covers the support of $\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$. The variational posterior approximation method based on the minimization of $\mathrm{KL}(\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})\|\mathrm{q}(\boldsymbol{\Theta}))$ is known as Expectation Propagation, the details of which will be presented in Section VI.

Since the minimum of the KL divergence is achieved at $\mathrm{q}(\boldsymbol{\Theta}) = \mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$, which can not be calculated, some assumptions on $\mathrm{q}(\boldsymbol{\Theta})$ have to be made. Before discussing several possible assumptions on $\mathrm{q}(\boldsymbol{\Theta})$, we notice from (24) that

$$\begin{aligned}&\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})) \\ &\quad= \int \mathrm{q}(\boldsymbol{\Theta}) \ln\left(\frac{\mathrm{q}(\boldsymbol{\Theta})}{\mathrm{p}(\boldsymbol{\Theta}, \mathbf{y})}\right) d\boldsymbol{\Theta} + \ln \mathrm{p}(\mathbf{y}),\end{aligned} \tag{27}$$

and therefore minimizing the KL divergence with respect to $\mathrm{q}(\boldsymbol{\Theta})$ does not require knowledge of $\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})$ (otherwise we end up in a loop).

One possible assumption on $\mathrm{q}(\boldsymbol{\Theta})$ is that it assumes specific parametric forms, for instance a Gaussian distribution. Another (very commonly used) assumption is that $\mathrm{q}(\boldsymbol{\Theta})$ factorizes into M disjoint groups, i.e.,

$$\mathrm{q}(\boldsymbol{\Theta}) = \prod_{i=1}^{M} \mathrm{q}_i(\boldsymbol{\Theta}_i), \tag{28}$$

where each factor q_i depends on a subset $\boldsymbol{\Theta}_i \subseteq \boldsymbol{\Theta}$. This factorized form of variational inference is called mean field theory in physics [40].

Using (28), the KL divergence can be minimized with respect to each of the factors q_i separately while holding the other factors fixed. The optimal solution for each of the factors can then be derived as [24]

$$\mathrm{q}_i(\boldsymbol{\Theta}_i) = Z_i \exp\left\{\mathrm{E}_{j\neq i}\left[\ln \mathrm{p}(\boldsymbol{\Theta}, \mathbf{y})\right]\right\}, \tag{29}$$

where Z_i is the normalization constant, and $\mathrm{E}_{j\neq i}[\cdot]$ denotes the expectation taken with respect to all the approximating factors $\{\mathrm{q}_j\}_{j\neq i}$. Note that (29) defines a system of M nonlinear equations in $\{\boldsymbol{\Theta}_i\}_{i=1}^{M}$. One way to solve this system of equations is via an alternating optimization procedure, where the distribution of each factor $\mathrm{q}_i(\Theta_i)$ is iteratively updated using the most recent distributions of all the other factors. This update process is cyclic and is repeated until convergence. Since the KL divergence (26) is convex with respect to $\mathrm{q}_i(\Theta_i)$ [41], the convergence is guaranteed.

Let us now examine how the MAP solution in (18) can be obtained as a particular variational distribution approximation. First we use the same distribution factorization as in (28) with the additional assumption that all the factors $\mathrm{q}_i(\boldsymbol{\Theta}_{ii})$ are degenerate at the peak $\boldsymbol{\Theta}_i^{\mathrm{d}}$, i.e.,

$$\mathrm{q}_i(\boldsymbol{\Theta}_i) = \begin{cases} 1, & \text{if} \quad \boldsymbol{\Theta}_i = \boldsymbol{\Theta}_i^{\mathrm{d}} \\ 0, & \quad \text{otherwise.} \end{cases} \tag{30}$$

Note that by imposing this constraint we ignore the uncertainty information provided by non-degenerate distributions. It then follows from (27)

$$\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta}) \| \mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})) = -\ln \mathrm{p}(\boldsymbol{\Theta}^{\mathrm{d}}, \mathbf{y}) + \ln \mathrm{p}(\mathbf{y}), \tag{31}$$

where $\boldsymbol{\Theta}^{\mathrm{d}} = (\boldsymbol{\Theta}_1^{\mathrm{d}}, \Theta_2^{\mathrm{d}}, \ldots, \boldsymbol{\Theta}_M^{\mathrm{d}})$. By iterating the minimization of $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta}) \| \mathrm{p}(\boldsymbol{\Theta}|\mathbf{y}))$ in (31) through all distributions $\mathrm{q}_i(\boldsymbol{\Theta}_i)$ we obtain the MAP estimates.

Let us now examine how the EM algorithm is another particular case of the variational approximation to posterior distribution. We consider here the partition of the unknown variables as in (2), and assume that the posterior distribution approximation has the form

$$\mathrm{q}(\boldsymbol{\Theta}) = \mathrm{q}(\mathbf{z})\mathrm{q}(\boldsymbol{\Omega}), \tag{32}$$

where we have removed the subscripts $\mathbf{z}$ and Ω on the right hand side of the above equation for simplicity. Then we have

$$\begin{aligned} &\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta}) \| \mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})) \\ &= \mathrm{KL}(\mathrm{q}(\mathbf{z})\mathrm{q}(\boldsymbol{\Omega}) \| \mathrm{p}(\boldsymbol{\Omega}, \mathbf{z}|\mathbf{y})) \\ &= \int \mathrm{q}(\mathbf{z})\mathrm{q}(\boldsymbol{\Omega}) \ln\left(\frac{\mathrm{q}(\mathbf{z})\mathrm{q}(\boldsymbol{\Omega})}{\mathrm{p}(\mathbf{z}, \boldsymbol{\Omega}, \mathbf{y})}\right) d\mathbf{z} d\boldsymbol{\Omega} + \ln \mathrm{p}(\mathbf{y}). \end{aligned} \tag{33}$$

TABLE I
Comparison of Inference Algorithms

	MAP	EM	VB	MCMC
Has full posterior	no	partial	yes	yes
Has point estimates	yes	yes	yes	yes
Has uncertainty info.	no	partial	yes	yes
Allows hidden data	no	yes	yes	yes
Complexity	low	low	medium	high

In addition, we impose the constraint that $\mathrm{q}(\boldsymbol{\Omega})$ is a degenerate distribution at $\boldsymbol{\Omega}^{\mathrm{d}}$ and denote by $\mathcal{C}$ the set of such distributions. Note, again, that this constraint will prevent us from obtaining uncertainty information on the posterior distribution approximation of $\boldsymbol{\Omega}$.

With the above assumptions we have that

$$\begin{aligned} &\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta}) \| \mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})) \\ &= \int \mathrm{q}(\mathbf{z}) \ln\left(\frac{\mathrm{q}(\mathbf{z})}{\mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}^{\mathrm{d}}, \mathbf{y})}\right) d\mathbf{z} - \ln \mathrm{p}(\boldsymbol{\Omega}^{\mathrm{d}}, \mathbf{y}) + \ln \mathrm{p}(\mathbf{y}) \end{aligned} \tag{34}$$

and therefore, given $\boldsymbol{\Omega}^{\mathrm{d}}$, the distribution $\mathrm{q}(\mathbf{z})$ minimizing the KL divergence in (34) is given by

$$\mathrm{q}(\mathbf{z}) = \mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}^{\mathrm{d}}, \mathbf{y}). \tag{35}$$

If a point estimate of $\mathbf{z}$ is required, representative statistics of $\mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}^{\mathrm{d}}, \mathbf{y})$ such as the mean can be used. In addition, we have other information made available by the approximate posterior distribution.

Now, the new estimate of $\boldsymbol{\Omega}$, denoted by $\boldsymbol{\Omega}^{\mathrm{d,new}}$, where the distribution $\mathrm{q}(\boldsymbol{\Omega})$ is degenerate, is determined by

$$\begin{aligned} \boldsymbol{\Omega}^{\mathrm{d,\ new}} &= \arg\min_{\mathrm{q}(\boldsymbol{\Omega})\in\mathcal{C}} \left\{\mathrm{KL}(\mathrm{q}(\mathbf{z})\mathrm{q}(\boldsymbol{\Omega}) \| \mathrm{p}(\boldsymbol{\Omega}, \mathbf{z}|\mathbf{y}))\right\} \\ &= \arg\min_{\boldsymbol{\Omega}} \left\{\mathrm{KL}(\mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}^{\mathrm{d}}, \mathbf{y}) \| \mathrm{p}(\boldsymbol{\Omega}, \mathbf{z}|\mathbf{y}))\right\} \\ &= \arg\min_{\boldsymbol{\Omega}} \left\{-\int \mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}^{\mathrm{d}}, \mathbf{y}) \ln \mathrm{p}(\boldsymbol{\Omega}, \mathbf{z}, \mathbf{y}) d\mathbf{z}\right\} \\ &= \arg\max_{\boldsymbol{\Omega}} \left\{\int \mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}^{\mathrm{d}}, \mathbf{y}) \ln \mathrm{p}(\boldsymbol{\Omega}, \mathbf{z}, \mathbf{y}) d\mathbf{z}\right\}. \end{aligned} \tag{36}$$

Notice that the expectation (the E-step in EM) in (36) depends on our ability to calculate $\mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}, \mathbf{y})$. If $\mathrm{p}(\mathbf{z}|\boldsymbol{\Omega}, \mathbf{y})$ is intractable, the EM algorithm can not be applied, but VB methods can still be used. Moreover, as already pointed out, the EM algorithm does not provide uncertainty information on $\mathrm{q}(\boldsymbol{\Omega})$ since this distribution is forced to be degenerate.

From the presentation above we see that variational methods provide general solutions to inference problems and provide approximate distributions (rather than merely point estimates) of the unknown variables. The comparison among the various inference algorithms described above is summarized in Table I. Examples of the application of VB algorithms in multimedia related problems are presented in the appendices.

IV. Local Variational Bounds for Joint and Conditional Probabilities in Variational Inference

The VB distribution approximation discussed so far can be considered as a global method since it approximates the posterior distribution by bounding $\ln \mathrm{p}(\mathbf{y})$ below with $\mathcal{L}(\mathrm{q}(\boldsymbol{\Theta}))$ (see (24) and (25)). An alternative local approach involves finding bounds on individual or groups of distributions in the joint probability model in (6). The general principle in local variational methods is to convert a complex quantity to a simpler one by expanding it with additional variational parameters. This simpler quantity is either a lower or an upper bound of the original quantity, and is utilized as its surrogate. Using this expansion results in optimization problems that are in turn tractable as opposed to the original ones.

The application of local variational methods to Bayesian inference relies on the approximations of the priors, the conditional distributions or the joint distribution. In this discussion we focus on the general formulation presented in Section III for consistency. Let us assume that the KL divergence in (27) and therefore the general solution in (29) can not be computed analytically for all unknowns $\{\boldsymbol{\Theta}_i\}_i$, due, for instance, to the form of the prior model $\mathrm{p}(\boldsymbol{\Theta})$ or the conditional distribution $\mathrm{p}(\mathbf{y}|\boldsymbol{\Theta})$. The goal therefore is to replace these complex distributions by their simplified bounds, which makes the computation in (27) and (29) tractable.

As an example illustrating the concept introduced above, consider again the classification problem given in Section II-A. In this example, both the l_p-quasinorm prior $\mathrm{p}(\mathbf{z}|\alpha)$ in (5) and the logistic sigmoid data likelihood $\mathrm{p}(\mathbf{y}|\mathbf{z})$ in (4) make the computation of the KL divergence analytically intractable. In the following we give lower bounds for $\mathrm{p}(\mathbf{z}|\alpha)$ and $\mathrm{p}(\mathbf{y}|\mathbf{z})$, respectively, and show how they can be used to resolve the intractability issue.

To find a lower bound on $\mathrm{p}(\mathbf{z}|\alpha)$, consider the following general relationship between weighted arithmetic and geometric means of two non-negative numbers a and b

$$a^p b^{1-p} \leq pa + (1-p)b, \tag{37}$$

with $0 \leq p \leq 1$. Applying this inequality to the exponent in (5), we have

$$|z_i|^p = \frac{\left(z_i^2\right)^{p/2} v_i^{1-p/2}}{v_i^{1-p/2}} \leq \frac{(p/2) z_i^2 + (1-p/2) v_i}{v_i^{1-p/2}}, \tag{38}$$

where v_i is a non-negative number.

Using this inequality, $\mathrm{p}(\mathbf{z}|\alpha)$ can be bounded variationally as follows

$$\begin{aligned}\mathrm{p}(\mathbf{z}|\alpha) \geq \mathrm{const} \cdot \alpha^{\frac{D}{p}} \exp\left[-\frac{\alpha p}{2} \sum_{i=1}^{D} \left(\frac{z_i^2 + \frac{2-p}{p} v_i}{v_i^{1-p/2}}\right)\right] \\ = \mathrm{M}(\mathbf{z}, \alpha, \mathbf{v}),\end{aligned} \tag{39}$$

where we have introduced the additional positive variational parameters $\mathbf{v} = \{v_i\}_{i=1}^D$. Note that with the lower bound approximation, the exponent in (39) is quadratic in $\mathbf{z}$ and the prior becomes a Gaussian.

To obtain a lower bound approximation to $\mathrm{p}(\mathbf{y}|\mathbf{z})$ we make use of the following inequality [24]

$$\sigma(a) \geq \sigma(b) \exp\left((a-b)/2 - \lambda(b)(a^2 - b^2)\right), \tag{40}$$

where a, b are arbitrary real numbers, and $\lambda(b) = \frac{1}{2b}(\sigma(b) - \frac{1}{2})$.

Using (40), a lower bound of the observation model in (4) is found as

$$\begin{aligned}\mathrm{p}(\mathbf{y}|\mathbf{z}) \geq \prod_{i=1}^{N} \exp\left(\mathbf{z}^{\mathrm{T}} \mathbf{x}_i y_i\right) \sigma\left(\xi_i\right) \exp\left\{-\left(\mathbf{z}^{\mathrm{T}} \mathbf{x}_i + \xi_i\right)/2\right. \\ \left. -\lambda(\xi_i)((\mathbf{z}^{\mathrm{T}} \mathbf{x}_i)^2 - \xi_i^2)\right\} = \mathrm{H}(\mathbf{z}, \boldsymbol{\xi}, \mathbf{y})\end{aligned} \tag{41}$$

where $\boldsymbol{\xi} = \{\xi_i\}_{i=1}^N$ are real-valued variational parameters.

Using the lower bounds in (39) and (41) we finally have

$$\mathrm{p}(\mathbf{z}, \alpha, \mathbf{y}) \geq \mathrm{p}(\alpha) \mathrm{M}(\mathbf{z}, \alpha, \mathbf{v}) \mathrm{H}(\mathbf{z}, \mathbf{y}, \boldsymbol{\xi}) = \mathrm{F}(\mathbf{y}, \mathbf{z}, \alpha, \mathbf{v}, \boldsymbol{\xi}), \tag{42}$$

which can be used to find an upper bound of the KL divergence in (26), that is

$$\begin{aligned}&\mathrm{KL}(\mathrm{q}(\Theta) \| \mathrm{p}(\Theta|\mathbf{y})) \\ &\leq \int \mathrm{q}(\Theta) \log\left(\frac{\mathrm{q}(\boldsymbol{\Theta})}{\mathrm{F}(\mathbf{y}, \mathbf{z}, \alpha, \mathbf{v}, \boldsymbol{\xi})}\right) d\Theta + \ln \mathrm{p}(\mathbf{y}).\end{aligned} \tag{43}$$

Instead of minimizing the KL divergence itself, we can minimize this upper bound to obtain the following general result instead of (29)

$$\mathrm{q}_i(\Theta_i) = Z_i \exp\left\{\mathrm{E}_{j \neq i}\left[\ln \mathrm{F}(\mathbf{y}, \mathbf{z}, \alpha, \mathbf{v}, \boldsymbol{\xi})\right]\right\}. \tag{44}$$

Note that since the bound $\mathrm{F}(\mathbf{y}, \mathbf{z}, \alpha, \mathbf{v}, \boldsymbol{\xi})$ is quadratic in $\mathbf{z}$, (44) can be calculated analytically as opposed to (29). An important question within this approach is the tightness of the bound in (39). Clearly, the tightness of this bound is determined by the selection of the variational parameters $\mathbf{v}$ and $\boldsymbol{\xi}$. By minimizing the bound in (43) with respect to the variational parameters $\mathbf{v}$ and $\boldsymbol{\xi}$ along with the unknowns $\mathbf{z}$ and α, one can obtain estimates of the unknowns sequentially approaching the ones that minimize the original KL divergence. In theory, exact values of the unknowns that minimize the original KL divergence can be obtained through the solution of this expanded optimization problem.

In general, it is hard to find bounds like the ones described above for arbitrary functions. However, a systematic method exists for finding variational transformations of certain distributions through the principle of convex duality [41], [42]. Due to limited space we skip the details and refer the interested readers to [43] for more information on this topic.

V. Variational Loopy Belief Propagation

In the previous sections we have discussed VB methods for approximating the posterior distributions. In this section we present an alternative approach for approximate inference, known as Loopy Belief Propagation. We start by introducing the sum-product algorithm, based on which LBP was developed. Then we show that for pairwise MRFs the LBP algorithm can be derived from a VB inference procedure.

Let us consider the joint probability distribution in (6) and assume that the model parameters Ω are fixed to known values or to values calculated in an iterative procedure. To simplify notation, define $\Gamma = \{\Omega, \mathbf{y}\}$ as the set of fixed (estimated or observed) variables. We are interested in estimating the posterior $\mathrm{p}(\mathbf{z}|\Gamma) \propto \mathrm{p}(\mathbf{z}, \Gamma)$. We define

$$\mathrm{P}(\mathbf{z}) = \mathrm{p}(\mathbf{z}|\Gamma), \tag{45}$$

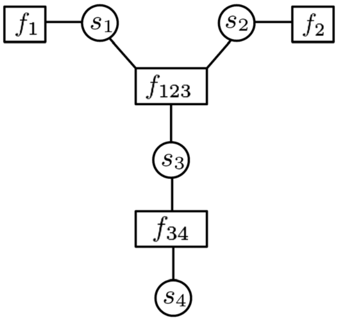


Fig. 4. Factor graph of the distribution in (8).

where we use capital letter to denote that the dependence on fixed variables has been suppressed.

Our goal is to calculate $\mathrm{P}(\mathbf{z})$. However, since in many real world problems this task is intractable we concentrate here on the calculation of the marginals (with respect to the elements in $\mathbf{z}$) and mode of $\mathrm{P}(\mathbf{z})$.

A. Factor Graphs and Message-Passing Algorithms

Let us follow [44] in the process to find the marginals and the mode. Using the directed or undirected graphical representation of $\mathrm{P}(\mathbf{z})$ we can write

$$\mathrm{P}(\mathbf{z}) = \frac{1}{Z}\prod_m f_m(\mathbf{z}_m) \tag{46}$$

where the factor $f_m(\mathbf{z}_m)$ is a function of a subset $\mathbf{z}_m \subseteq \mathbf{z}$ and Z is the normalization constant.

For functions factorized in the form of (46) and their graphical representations, we can create the corresponding factor graphs. In a factor graph each variable $z_n \in \mathbf{z}$ is represented as a circle, and each factor f_m is represented as a square. An edge between the variable z_n and the factor f_m exists if and only if $z_n \in \mathbf{z}_m$.

As one example of converting directed graphs into factor graphs, Fig. 4 depicts the factor graph corresponding to the distribution on $\mathbf{s} = \{s_1, s_2, s_3, s_4\}$ in (8), where the following factors are defined

$$\begin{cases} f_1(s_1) = \mathrm{p}(s_1), \\ f_2(s_2) = \mathrm{p}(s_2), \\ f_{123}(s_1, s_2, s_3) = \mathrm{p}(s_3|s_1, s_2), \\ f_{34}(s_3, s_4) = \mathrm{p}(s_4|s_3). \end{cases} \tag{47}$$

Note that, for clarity, again we have replaced m in (46) by the indices of the variables in the corresponding factor.

Similarly, Fig. 5 depicts the factor graph corresponding to the HMM in (9), where the following factors are defined

$$\begin{cases} f_0(z_1) = \mathrm{p}(z_1), & \\ f_m(z_m) = \mathrm{p}(y_m|z_m), & m = 1, 2, 3, 4, \\ f_{m-1m}(z_{m-1}, z_m) = \mathrm{p}(z_m|z_{m-1}), & m = 2, 3, 4 \end{cases} \tag{48}$$

Note that we have suppressed the dependence on $\{y_m\}_{m=1}^4$ in (48) since they are observed and fixed.

For undirected graphs, Figs. 6 and 7 depict the factor graphs corresponding to the distributions in (11) and (12), respectively.

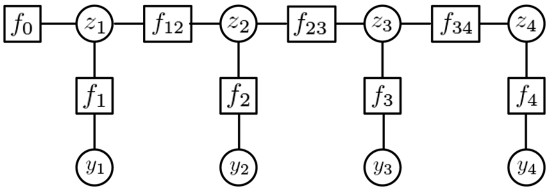


Fig. 5. Factor graph of the distribution in (9).

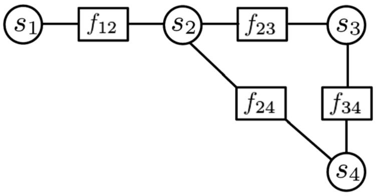


Fig. 6. Factor graph of the distribution in (11).

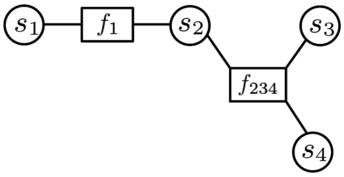


Fig. 7. Factor graph of the distribution in (12).

Notice that the factor graphs in Figs. 4, 5, and 7 are all trees (i.e., connected graphs without cycles), while the one in Fig. 6 is not.

The introduction of factor graphs as well as the conversion from directed/undirected graphs to factor graphs allows us to calculate the marginal distributions $\mathrm{P}(z_n)$ from (46). The method we present to calculate these marginals is called the sum-product algorithm and it is a generalization of the message-passing or belief propagation (BP) method proposed in [45]. It is important to note that it will exactly recover the marginal distributions if the factor graph is a tree [2]. When the factor graph is not a tree there is no guarantee that the algorithm will recover the marginals. Note that in order to simplify the presentation, we assume that the variables in $\mathbf{z}$ are discrete such that marginalization is performed by a summation.

We denote by $\mathcal{N}(m)$ the index set of variables that factor f_m depends on, i.e., $\mathcal{N}(m)$ is the set of indices for variables in $\mathbf{z}_m$. Analogously $\mathcal{M}(n)$ denotes the index set of factors in which variable z_n is present, i.e., $z_n \in \mathbf{z}_m$ for $m \in \mathcal{M}(n)$. We will use $\backslash$ to remove a variable index or a factor index from a set, whose meaning will be clear from the context.

The sum-product algorithm defines two types of messages, namely those from variables to factors and those from factors to variables. Specifically, $q_{n\to m}$ denotes the message sent by variable z_n and received by factor f_m, and $r_{m\to n}$ denotes the message sent by factor f_m and received by variable z_n. Both types of messages are functions of the value of z_n.

There are two rules to update the messages

- From variable to factor

$$q_{n\to m}(z_n) = \prod_{m'\in\mathcal{M}(n)\backslash m} r_{m'\to n}(z_n) \tag{49}$$

- From factor to variable

$$r_{m\rightarrow n}(z_n)=\sum_{\mathcal{N}(m)\backslash n}\left(f_m(\mathbf{z}_m)\prod_{n'\in\mathcal{N}(m)\backslash n}q_{n'\rightarrow m}(z'_n)\right) \tag{50}$$

  A factor node which is connected to only one variable node will always broadcast $r_{m\rightarrow n}(z_n)=f_m(z_n)$. Similarly a variable node that is connected to only one factor node will always broadcast $q_{n\rightarrow m}(z_n)=1$.

Intuitively, a message can be considered as the sender's belief of the involved variable taking the specified value. For example, $q_{n\rightarrow m}(z_n)$ represents "how much" variable z_n believes in itself taking the specified value, and $r_{m\rightarrow n}(z_n)$ represents the factor f_m's belief on variable z_n taking the specified value. Note that from (49) and (50) it is clear that message updates are performed in an interlocked fashion. A variable z_n summarizes (by taking products of) the incoming messages from neighboring factors other than the destination factor f_m and sends the message to f_m. A factor f_m summarizes the incoming messages from neighboring variables other than the destination variable z_n, multiplies them with f_m itself, marginalizes variables other than z_n, and sends the message to z_n. Note that a factor can send a message to a variable once it has received incoming messages from all other neighboring variables, and the same applies when a variable has to send a message to a neighboring factor.

To find the marginal for each variable we proceed as follows [24]: pick any node and designate it as the root, propagate messages from all the leaves up to the root using (49) and (50), then propagate messages back from the root to all the leaves. After this two-pass message passing every variable has received messages from all its neighboring factors. The marginals are then found as

$$\mathrm{P}(z_n)\propto\prod_{m\in\mathcal{M}(n)}r_{m\rightarrow n}(z_n) \tag{51}$$

To find the mode of $\mathrm{P}(\mathbf{z})$ in a tree-structured factor graph the max-product method can be employed, which replaces all the summations in (50) with maximization operations in the process of propagating messages from the leaves to the root. During this "bottom-up" message passing, a record of the variables in $\mathbf{z}_m\backslash z_n$ that have given rise to the maximum is kept at each factor. After the root receives all the incoming messages, the algorithm back tracks down to the leaves to find the maximizing values of the variables, following the records kept at the factors [24].

B. LBP on Pairwise MRFs

We have just described a method to find the marginals from joint distributions representable as tree-structured factor graphs. The initial broadcasting has been defined from variable nodes to factor nodes as well as from factor nodes to variable nodes. As an alternative, we can also start by initializing all variable-to-factor messages to one, that is,

$$q_{n\rightarrow m}(z_n)=1\quad\text{for all } n,m, \tag{52}$$

and then proceed with both types of message updates in parallel. The main difference between this initialization and the one described above is that this can be applied to factor graphs which are not trees. See [2] for the convergence of this method and the different message scheduling.

In the rest of this section (for notational simplicity) we will assume that $\mathrm{P}(\mathbf{z})$ in (46) is a pairwise MRF, that is,

$$\mathrm{P}(\mathbf{z})\propto\prod_i f_i(z_i)\prod_{(i,j)\in\mathcal{E}}f_{i,j}(z_i,z_j), \tag{53}$$

where $f_i(z_i)$ is the local evidence (notice that it will very likely contain information on a set of observed variables, which have been considered fixed and hence suppressed in notation), $f_{i,j}(z_i,z_j)$ is the potential for edge $i-j$ and $\mathcal{E}$ is the set of undirected edges. In the prior and conditional distribution terminology $\prod_i f_i(z_i)$ models the observation process and $\prod_{(i,j)\in\mathcal{E}} f_{i,j}(z_i,z_j)$ corresponds to the prior distribution.

For the pairwise MRF described in (53) there are two types of factors: unary factors associated with $f_i(z_i)$ and binary factors associated with $f_{i,j}(z_i,z_j)$. With the initialization in (52), the original two types of message updates reduce to one: the message passing between variables. Denote the message passed from z_i to z_j as $m_{i\rightarrow j}(z_j)$ and the belief at z_i as $b(z_i)$. We denote the index set of the neighboring variables of z_i as $\mathrm{nbrs}(i)$. The pseudocode for LBP on a pairwise MRF is given in Algorithm 1.

Algorithm 1 Loopy Belief Propagation on a pairwise MRF [2]

```
  Initialize messages $m_{i\rightarrow j}(z_j)=1$ for all edges $i-j$;
  Initialize beliefs $b(z_i)\propto 1$ for all nodes $z_i$;
  **repeat**
     Send message on each node
     $m_{i\rightarrow j}(z_j)=\sum_{z_i}\left(f_i(z_i)f_{i,j}(z_i,z_j)\prod_{k\in\mathrm{nbrs}(i)\backslash j}m_{k\rightarrow i}(z_i)\right)$
Update belief of each node
     $b(z_i)\propto f_i(z_i)\prod_{j\in\mathrm{nbrs}(i)}m_{j\rightarrow i}(z_i)$
  **until** beliefs don't change significantly
  Return marginal probabilities $\mathrm{P}(z_i)\propto b(z_i)$
```

C. Connection Between LBP and VB Inference

Let us now interpret the LBP algorithm from a variational perspective. First we note that our goal in Bayesian analysis is to find an approximate posterior distribution via

$$\min_{\mathrm{q}(\mathbf{z})}\mathbf{E}_{\mathrm{q}(\mathbf{z})}\left[\log\frac{\mathrm{q}(\mathbf{z})}{\prod_i f_i(z_i)\prod_{(i,j)\in\mathcal{E}}f_{i,j}(z_i,z_j)}\right]. \tag{54}$$

Observe that the unnormalized distribution of $\mathbf{z}$ in (53) can be written as

$$\prod_i f_i(z_i)\prod_{(i,j)\in\mathcal{E}}f_{i,j}(z_i,z_j)=\frac{\prod_{(i,j)\in\mathcal{E}}f_i(z_i)f_{i,j}(z_i,z_j)f(z_j)}{\prod_i f_i^{c_i}(z_i)}, \tag{55}$$

where c_i is the number of neighbors of variable z_i minus 1. To reveal the connection between LBP and VB, consider the following form of approximation to $\mathrm{P}(\mathbf{z})$

$$\mathrm{q}(\mathbf{z}) = \prod_{(i,j)\in\mathcal{E}} \mathrm{q}_{i,j}(z_i, z_j) / \prod_{i} q_i^{c_i}(z_i), \tag{56}$$

where $\mathrm{q}_{i,j}(z_i, z_j)$ and $\mathrm{q}_i(z_i)$ are the approximate marginal binary and unary distributions, respectively.

It is shown in [1] and [2] that utilizing (56) in the minimization of the KL divergence in (26) leads to

$$\begin{aligned} \mathrm{q}(z_i) &\propto f_i(z_i) \prod_{j\in\mathcal{M}(i)} \prod_{k\in\mathcal{M}(i)\backslash j} (m_{k\to i}(z_i))^{1/c_i} \\ &= f_i(z_i) \prod_{j\in\mathcal{M}(i)} m_{j\to i}(z_i), \end{aligned} \tag{57}$$

which is the belief update rule in Algorithm 1. Finally, the pairwise distribution in (56) can be written as

$$\begin{aligned} \mathrm{q}_{i,j}(z_i, z_j) \propto f_i(z_i) f_j(z_j) f_{i,j}(z_i, z_j) \\ \prod_{k\in\mathcal{M}(i)\backslash j} m_{k\to i}(z_i) \prod_{k'\in\mathcal{M}(j)\backslash i} m_{k'\to j}(z_j) \end{aligned} \tag{58}$$

From the discussion above, we see that by adopting the form of approximation in (56), the message update rules in belief propagation can be derived within the variational Bayesian framework.

VI. Expectation Propagation

In our presentation of the VB analysis in Section III, the inference procedure was based on the minimization of $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y}))$ in (26). As is pointed out, the KL divergence is not symmetric, that is, $\mathrm{KL}(\mathrm{q}(\boldsymbol{\Theta})\|\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})) \neq \mathrm{KL}(\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})\|\mathrm{q}(\boldsymbol{\Theta}))$. Therefore, another set of approximation methods can be obtained by minimizing the forward KL divergence $\mathrm{KL}(\mathrm{p}(\boldsymbol{\Theta}|\mathbf{y})\|\mathrm{q}(\boldsymbol{\Theta}))$, giving rise to the Expectation Propagation algorithm [46], [47]. EP methods are less mature than the VB methods presented above, but they also have the potential of accurately approximating the posterior distributions in certain inverse problems [24], [48].

EP methods result from the modification of another method for approximating the posterior distribution, referred to as Assumed Density Filtering (ADF) (see [46] and references therein). We therefore present ADF first, followed by the EP presentation. ADF was independently proposed in the statistics and control literature. Its name was coined in control; other names include online Bayesian learning, moment matching, and weak marginalization.

As an introductory case, let us assume that the model parameters have been fixed at their known or estimated values, similarly to the assumption made in Section V. Also assume that the observations $\mathbf{y} = \{\mathbf{y}_i\}_{i=1}^N$ have been taken independently of each other, such that the joint distribution factorizes as follows

$$\mathrm{p}(\mathbf{z}, \mathbf{y}) = \mathrm{p}(\mathbf{z}) \prod_{i=1}^{N} \mathrm{p}(\mathbf{y}_i|\mathbf{z}). \tag{59}$$

Note that this assumption is valid in many practical scenarios. A more general distribution represented by a factor graph will be considered later.

To motivate the EP algorithm, let us consider a cluttering problem [24]. In this example, each observation $\mathbf{y}_i$ is a Gaussian mixture consisting of a signal component with mean $\mathbf{z}$ and a background clutter with mean $\mathbf{0}$. Moreover, $\mathbf{z}$ is modeled as a zero-mean Gaussian distribution. Given these definitions, the joint distribution involving N observations and the unknown mean $\mathbf{z}$ is expressed as

$$\mathrm{p}(\mathbf{z}, \mathbf{y}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, a\mathbf{I}) \prod_{i=1}^{N} \left[(1-w)\mathcal{N}(\mathbf{y}_i|\mathbf{z}, \mathbf{I}) + w\mathcal{N}(\mathbf{y}_i|\mathbf{0}, b\mathbf{I})\right], \tag{60}$$

where the weights w and the variance levels a and b are all fixed and assumed known. The posterior distribution of $\mathbf{z}$ is proportional to $\mathrm{p}(\mathbf{z}, \mathbf{y})$ and hence involves 2^N terms, which makes exact inference infeasible for large values of N. Therefore, we need to approximate the posterior distribution of $\mathbf{z}$.

By defining $\mathrm{q}_0(\mathbf{z}) = \mathrm{p}(\mathbf{z})$ we have an (exact) approximation of the posterior distribution of $\mathbf{z}$ (which is also the joint distribution) when no observations are available. We now take into account the first observation $\mathbf{y}_1$. If $\mathrm{q}_1(\mathbf{z})$ approximates the distribution $\mathrm{p}(\mathbf{y}_1|\mathbf{z})\mathrm{q}_0(\mathbf{z})/Z_1$, where $Z_1 = \int \mathrm{p}(\mathbf{y}_1|\mathbf{z})\mathrm{q}_0(\mathbf{z})\mathrm{d}\mathbf{z}$, then

$$\mathrm{p}(\mathbf{z}, \mathbf{y}_1) \approx Z_1 \mathrm{q}_1(\mathbf{z}). \tag{61}$$

As more observations are added, we have

$$\mathrm{p}(\mathbf{z}, \mathbf{y}_1, \ldots, \mathbf{y}_k) \approx \mathrm{q}_k(\mathbf{z}) \prod_{l=1}^{k} Z_l \tag{62}$$

where

$$\mathrm{q}_k(\mathbf{z}) \approx \frac{\mathrm{p}(\mathbf{y}_k|\mathbf{z})\mathrm{q}_{k-1}(\mathbf{z})}{Z_k} \tag{63}$$

and

$$Z_k = \int \mathrm{p}(\mathbf{y}_k|\mathbf{z})\mathrm{q}_{k-1}(\mathbf{z})\mathrm{d}\mathbf{z}. \tag{64}$$

The left hand side of (62) is the joint distribution, and $\mathrm{q}_k(\mathbf{z})$ on the right hand side is an approximation to the posterior distribution given k observations. Therefore, $\prod_{l=1}^{k} Z_l$ is an approximation to the model evidence $\mathrm{p}(\mathbf{y}_1, \ldots, \mathbf{y}_k)$. Finally, after all N observations are taken, $\mathrm{q}_N(\mathbf{z})$ will provide an approximation of $\mathrm{p}(\mathbf{z}|\mathbf{y})$ and $\mathrm{p}(\mathbf{y}) \approx \prod_{l=0}^{N} Z_l$.

To complete the ADF method description we need to define how the proximity between two distributions is measured and the type of distribution $\mathrm{q}_k(\mathbf{z})$ used. The distribution $\mathrm{q}_k(\mathbf{z})$ is selected as

$$\mathrm{q}_k(\mathbf{z}) = \underset{\mathrm{q}(\mathbf{z})\in\mathcal{C}}{\arg\min}\, \mathrm{KL}\left(\frac{\mathrm{p}(\mathbf{y}_k|\mathbf{z})\mathrm{q}_{k-1}(\mathbf{z})}{Z_k} \| \mathrm{q}(\mathbf{z})\right), \tag{65}$$

where $\mathcal{C}$ is a given class of probability distributions.

For the mixture example described above it is clear that a good choice for $\mathrm{q}_k(\mathbf{z})$ is a multivariate Gaussian distribution. In general, members of the exponential family are used. Note that if $\mathrm{q}_k(\mathbf{z})$ is from the exponential family, the minimization of the KL divergence (65) reduces to the estimation of a set of sufficient statistics. In the particular case when $\mathrm{q}_k(\mathbf{z})$ is a Gaussian distribution, this is simply to match the mean and covariance of $\mathrm{q}_k(\mathbf{z})$ with those of $\mathrm{p}(\mathbf{y}_k|\mathbf{z})\mathrm{q}_{k-1}(\mathbf{z})/Z_k$, giving rise to the name of moment matching.

A modification of ADF is obtained by approximating each $\mathrm{p}(\mathbf{y}_i|\mathbf{z})$ (as a function of $\mathbf{z}$) by an unnormalized density function and then use the product of the unnormalized densities (and the prior) to approximate $\mathrm{p}(\mathbf{z},\mathbf{y})$. As already mentioned, this refinement of ADF leads to EP.

To provide an algorithmic description of EP, we first note that for a given set of observations $\mathbf{y}$ we can write

$$\mathrm{p}(\mathbf{z},\mathbf{y}) = \mathrm{p}(\mathbf{z})\prod_{i=1}^{N}\mathrm{p}(\mathbf{y}_i|\mathbf{z}) = \prod_{i=0}^{N} f_i(\mathbf{z}) \tag{66}$$

where $f_0(\mathbf{z}) = \mathrm{p}(\mathbf{z})$ and $f_i(\mathbf{z}) = \mathrm{p}(\mathbf{y}_i|\mathbf{z})$ for $i = 1,\ldots,N$.

The factor graph notation $\mathrm{p}(\mathbf{z},\mathbf{y}) = \prod_i f_i(\mathbf{z})$ in (66) is very general and can be used to represent both directed and undirected graphs. Since $\mathrm{p}(\mathbf{z}|\mathbf{y}) \propto \mathrm{p}(\mathbf{z},\mathbf{y})$, our goal is to find an approximate posterior distribution in the form of

$$\mathrm{q}(\mathbf{z}) = \frac{1}{Z}\prod_i \tilde{f}_i(\mathbf{z}), \tag{67}$$

where $\tilde{f}_i(\mathbf{z})$ is an approximate of $f_i(\mathbf{z})$, and provide an estimate of the model evidence $Z \approx \mathrm{p}(\mathbf{y})$. The model evidence can be used to estimate the model parameters if needed.

The EP algorithm for the approximation of the posterior distribution of $\mathbf{z}$ for the joint representation in (66) is summarized in Algorithm 2.

Algorithm 2 Expectation Propagation-1 From $\mathrm{p}(\mathbf{z},\mathbf{y})$ in (66), approximate $\mathrm{p}(\mathbf{z}|\mathbf{y})$ by $\mathrm{q}(\mathbf{z})$ in (67) and estimate $\mathrm{p}(\mathbf{y})$

```
Let $\mathcal{C}$ be a given class of probability distributions.
Initialize the approximating factors $\tilde{f}_i(\mathbf{z}) \in \mathcal{C}$
Initialize the approximate posterior $\mathrm{q}(\mathbf{z}) \propto \prod_i \tilde{f}_i(\mathbf{z})$
repeat
    Select a factor $\tilde{f}_i(\mathbf{z})$ to refine
    Remove $\tilde{f}_i(\mathbf{z})$ from the approximate posterior by division
        $\mathrm{q}_{-i}(\mathbf{z}) = \mathrm{q}(\mathbf{z})/\tilde{f}_i(\mathbf{z})$
    Compute $f_i(\mathbf{z})\mathrm{q}_{-i}(\mathbf{z})$ and denote the normalization constant by $Z_i = \int f_i(\mathbf{z})\mathrm{q}_{-i}(\mathbf{z})\mathrm{d}\mathbf{z}$
    Find
        $\mathrm{q}^{\mathrm{new}}(\mathbf{z}) = \arg\min_{\mathrm{q}(\mathbf{z})\in\mathcal{C}} \mathrm{KL}(f_i(\mathbf{z})\mathrm{q}_{-i}(\mathbf{z})/Z_i\|\mathrm{q}(\mathbf{z}))$
    Update the factor $\tilde{f}_i(\mathbf{z}) = Z_i\mathrm{q}^{\mathrm{new}}(\mathbf{z})/\mathrm{q}_{-i}(\mathbf{z})$.
until convergence
Calculate the approximate model evidence
    $\mathrm{p}(\mathbf{y}) \approx \int \prod_i \tilde{f}_i(\mathbf{z})\mathrm{d}\mathbf{z}$    (68)
```

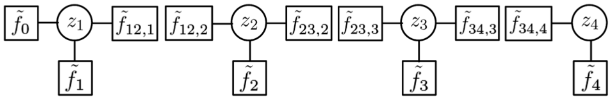


Fig. 8. Factor graph of the EP approximation of the distribution in (9) by the distribution in (72).

In the above EP description we have assumed that all the factors $f_i(\mathbf{z})$ and approximations $\tilde{f}_i(\mathbf{z})$ are functions of the entire $\mathbf{z}$. We consider now the case where the factors depend only on a subset of the variables in $\mathbf{z}$. Similarly to the discussion above, the objective is, given the factorized joint distribution

$$\mathrm{p}(\mathbf{z},\mathbf{y}) = \prod_i f_i(\mathbf{z}_i), \tag{69}$$

to obtain an approximate posterior

$$\mathrm{q}(\mathbf{z}) = \frac{1}{Z}\prod_i \tilde{f}_i(\mathbf{z}_i) \tag{70}$$

In the description that follows we assume that

$$\tilde{f}_i(\mathbf{z}_i) = \prod_j \tilde{f}_{ij}(z_j) \tag{71}$$

where $\mathbf{z}_i = \{z_j\}$, that is, z_j denotes the variables in $\mathbf{z}_i$. Note that it would have been more appropriate to use $\{z_j^i\}$ to denote the set of variables in $\mathbf{z}_i$, however it will always be clear from the context the variables and factors we refer to.

Let us illustrate this factorized approximation with the HMM example in (9) and (48). For the factor graph in Fig. 5, we seek an approximate posterior distribution of the form

$$\mathrm{q}(\mathbf{z}) \propto \prod_{i=0}^{4}\tilde{f}_i(z_i)\prod_{i=2}^{4}\tilde{f}_{i-1i,i-1}(z_{i-1})\tilde{f}_{i-1i,i}(z_i). \tag{72}$$

Note that, for clarity, in the subscripts of factors coming from two variables, we have made explicit the variables involved. Graphically we are converting the graphical model in Fig. 5 to the one in Fig. 8.

The EP algorithm to approximate the posterior distribution for the model in (69) is presented in Algorithm 3. An example of the application of this model in Gaussian Process Classification (GPC) is presented in Appendix D.

In Algorithm 3 the approximate factor components are expressed as

$$\tilde{f}_{ij}(z_j) \propto \sum_{z_{m\neq j}} f_i(\mathbf{z}_i)\prod_{l\neq i}\prod_{m\neq j}\tilde{f}_{lm}(z_m). \tag{73}$$

This is the sum-product rule in which messages from variable nodes to factor have been eliminated and all the j-terms $\tilde{f}_{ij}(z_j)$ are updated simultaneously. This suggests, as explained in [46], that more flexible approximating distributions, which could involve the grouping of several factors or different partitions of the variables in $\mathbf{z}_i$, could be used to achieve higher accuracy.

Algorithm 3 Expectation propagation-2 From $\mathrm{p}(\mathbf{z}, \mathbf{y})$ in (69), approximate $\mathrm{p}(\mathbf{z}|\mathbf{y})$ by $\mathrm{q}(\mathbf{z})$ in (70) and estimate $\mathrm{p}(\mathbf{y})$

```
Initialize the approximating factors $\tilde{f}_{ij}(z_j)$
Initialize the approximate posterior $\mathrm{q}(\mathbf{z}) \propto \prod_i \prod_j \tilde{f}_{ij}(z_j)$
repeat
    Select a product of factors $\prod_j \tilde{f}_{ij}(z_j)$ to refine
    Remove $\prod_j \tilde{f}_{ij}(z_j)$ from the approximate posterior by division
        $\mathrm{q}_{-i}(\mathbf{z}_i) = \mathrm{q}(\mathbf{z}) / \prod_j \tilde{f}_{ij}(z_j)$
    Compute $f_i(\mathbf{z}_i)\mathrm{q}_{-i}(\mathbf{z}_i)$ and denote the normalization constant by $Z_i = \int f_i(\mathbf{z}_i)\mathrm{q}_{-i}(\mathbf{z}_i)\mathrm{d}\mathbf{z}_i$.
    Find
        $\mathrm{q}^{\mathrm{new}}(\mathbf{z}_i) = \arg\min_{\mathrm{q}(\mathbf{z}_i) \in \mathcal{C}_i} \mathrm{KL}(f_i(\mathbf{z}_i)\mathrm{q}_{-i}(\mathbf{z}_i)/Z_i \| \mathrm{q}(\mathbf{z}_i))$
    where $\mathcal{C}_i$ is the set of probability distributions for which $\mathrm{q}(\mathbf{z}_i) = \prod_j \mathrm{q}(z_j)$
    Update the product of factors
        $\prod_j \tilde{f}_{ij}(z_j) = Z_i \mathrm{q}^{\mathrm{new}}(\mathbf{z}_i)/\mathrm{q}_{-i}(\mathbf{z}_i)$
until convergence
Calculate the approximation of the model evidence
    $\mathrm{p}(\mathbf{y}) \approx \int \prod_i \prod_j \tilde{f}_{ij}(z_j)\mathrm{d}\mathbf{z}$    (74)
```

VII. Conclusions

In this paper we have provided an overview of variational Bayesian modeling and inference methods for multimedia and related areas based on the use of probabilistic graphical models. We have elaborated on the principles of VB inference as well as discussed its relative merits and limitations compared with various other inference algorithms including MAP, EM and MCMC. VB provides (approximate) full posterior distribution to a wider range of problems at medium cost of computation. The use of local variational bounds and several distribution representations have been discussed and shown to lead to tractable variational inference. We have also shown the connection between VB and LBP, as well as the connection with other posterior approximation methods such as the global and local EP algorithms. Examples of VB algorithms applied to representative multimedia problems have been included as appendices.

Appendix A
Image Blind Deconvolution using VB Analysis

Consider the application of VB analysis to the image blind deconvolution problem. We describe here a simple yet powerful VB model, and refer the reader to [36], [49] for the omitted algorithmic details.

Consider the following model utilized for the generation of observations $\mathbf{y}$ from unknown variables $\mathbf{x}$,

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \tag{A.1}$$

where $\mathbf{n} \in \mathbb{R}^N$ is the additive noise and $\mathbf{H} \in \mathbb{R}^{N \times N}$ represents the system matrix. In image blind deconvolution the system matrix $\mathbf{H}$ is constructed using the unknown blur Point Spread Function (PSF) $\mathbf{h} \in \mathbb{R}^N$. The goal of blind deconvolution is to provide an estimate of the original image $\mathbf{x}$ and the blur PSF $\mathbf{h}$, given the observations $\mathbf{y}$ and the prior knowledge.

One crucial design component in blind deconvolution is accurate modeling of natural image characteristics. It is well known that when high-pass filters are applied to natural images, the resulting coefficients are sparse. This property is expressed via the use of sparse image priors. Specifically, we consider the following general form of super-Gaussian image priors on x_γ

$$\mathrm{p}(x_\gamma) = \prod_{i=1}^{N} \mathrm{p}\left(x_\gamma(i)\right) = Z \exp\left(-\sum_i \rho\left(x_\gamma(i)\right)\right), \tag{A.2}$$

where Z is the normalization constant, $\rho(\cdot)$ is a penalty function, and $x_\gamma(i)$ denotes the output of a high-pass filter output at pixel i. A variety of functions that can be represented using A.2 is discussed in [36].

Sparsity is achieved with sub-quadratic forms of $\rho(\cdot)$, which do not lead to straightforward application of Bayesian inference. However, with some acceptable restrictions on its form, the function $\rho(\cdot)$ can bounded as follows [36]

$$\rho\left(x_\gamma(i)\right) \leq \frac{1}{2}\xi_\gamma(i)x_\gamma^2(i) - \rho^*\left(\frac{1}{2}\xi_\gamma(i)\right), \tag{A.3}$$

where $\xi_\gamma(i)$ is a variational parameter and $\rho^*(\cdot)$ denotes the concave conjugate of $\rho(\cdot)$. Notice that only the first, quadratic term depends on $\mathbf{x}_\gamma$ and hence the image prior in A.2 can be lower bounded by a Gaussian distribution, which is more amenable for Bayesian inference. As an alternative, the prior in (A.2) can be represented as

$$\mathrm{p}(x_\gamma) = \int \mathrm{p}(x_\gamma|\xi_\gamma)\mathrm{p}(\xi_\gamma)\mathrm{d}\xi_\gamma, \tag{A.4}$$

where $\mathrm{p}(x_\gamma|\xi_\gamma)$ is a Gaussian distribution with precision ξ_γ and therefore (A.4) is known as the Scale Mixture of Gaussians (SMG) model. The SMG is a bit more restrictive than variational lower bounding in terms of the class of priors that can be represented.

Using the above representations, the super-Gaussia priors are transformed to Gaussian forms. VB inference can be applied then for the inference of the unknowns [36], [43]. Fig. 9 presents an example of applying the prior in (A.2) with $\rho(\cdot)$ being a logarithm function. Fig. 9(b) shows the restored image along with the estimated blur kernel. As can be seen, most part of the restored image is sharp, corroborating the effectiveness of the model and the VB algorithm.

Appendix B
Video Foreground/Background Separation and Network Anomaly Detection using VB Analysis

As two additional examples of application of VB algorithms in multimedia problems, we consider foreground/background

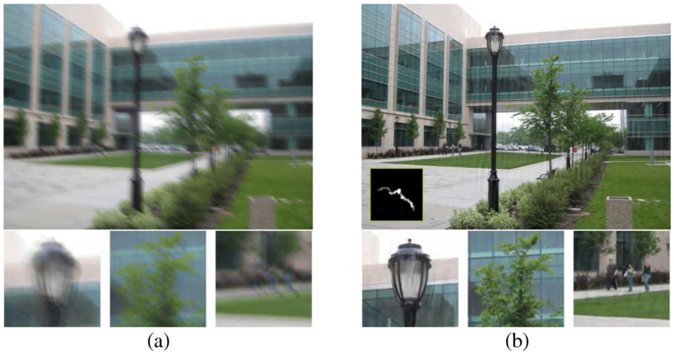


Fig. 9. Blind image deconvolution using VB analysis.

separation in video analysis [50], [51] and anomaly detection in the field of network security [52], [53], which share a common data model as we will see. The algorithmic details are based on [13], [21].

Consider the measurement system expressed as

$$\mathbf{Y} = \mathbf{R}\mathbf{E} + \mathbf{X} + \mathbf{N}, \tag{B.1}$$

where the signal of interest $\mathbf{E} \in \mathbb{R}^{M \times N}$ undergoes a transformation $\mathbf{R} \in \mathbb{R}^{L \times M}$ and is corrupted by both noise $\mathbf{N}$ and a smooth background $\mathbf{X}$. The signal $\mathbf{E}$ is assumed to be column-wise sparse, i.e., $\|\mathbf{e}_{\cdot i}\|_0 \ll M$ for $i = 1, \ldots, N$, where $\| \cdot \|_0$ is the ℓ_0-(pseudo)norm. The smooth background $\mathbf{X}$ is a low-rank matrix consisting of linearly dependent columns. The transformation $\mathbf{R}$ in general has the effect of compression, i.e., $L \leq M$.

In video analysis, $\mathbf{E}$ denotes the moving objects in the foreground, $\mathbf{R}$ is a known matrix representing measurement distortion such as blurring and resolution scaling, and $\mathbf{X}$ is the measured background. In network anomaly detection, $\mathbf{E}$ consists of the temporal snapshots of flow anomalies, $\mathbf{R}$ represents the network routing operation, and $\mathbf{X}$ contains the smooth link measurements resulting from the normal traffic flows.

The model in (B.1) subsumes Compressive Sensing (CS) and Robust Principal Component Analysis as its two special cases. For CS, the low-rank component $\mathbf{X}$ is not present, $\mathbf{R}$ is random, while for RPCA, the measurement matrix $\mathbf{R}$ reduces to an identity matrix.

Hierarchical Bayesian model has been employed to capture the low-rank and sparse properties of the corresponding terms as well as the data observation process. Specifically, $\mathbf{X}$ is modeled as the sum of outer products $\mathbf{X} = \mathbf{A}\mathbf{B}^{\mathrm{T}} = \sum_i^k \mathbf{a}_{\cdot i}\mathbf{b}_{\cdot i}^{\mathrm{T}}$, where Gaussian priors with Gamma precisions $\boldsymbol{\gamma} = \{\gamma_i\}_{i=1}^k$ are used such that most of the terms in the summation are annihilated during the inference procedure and $\mathbf{X}$ is rendered low rank. The sparse term $\mathbf{E}$ consists of i.i.d. Gaussian entries, whose precisions $\boldsymbol{\alpha} = \{\alpha_{ij}\}$ are assumed to have Jeffreys prior. The observation noise is assumed to be white Gaussian with precision β. The joint distribution involving all variables is expressed as

$$\begin{aligned} &\mathrm{p}(\mathbf{Y}, \mathbf{A}, \mathbf{B}, \mathbf{E}, \boldsymbol{\gamma}, \boldsymbol{\alpha}, \beta) \\ &= \mathrm{p}(\mathbf{Y}|\mathbf{A}, \mathbf{B}, \mathbf{E}, \beta)\mathrm{p}(\mathbf{A}|\boldsymbol{\gamma})\mathrm{p}(\mathbf{B}|\boldsymbol{\gamma})\mathrm{p}(\boldsymbol{\gamma})\mathrm{p}(\mathbf{E}|\boldsymbol{\alpha})\mathrm{p}(\boldsymbol{\alpha})\mathrm{p}(\beta). \end{aligned} \tag{B.2}$$

Using the notation in (2), we have that $\mathbf{z} = \{\mathbf{A}, \mathbf{B}, \mathbf{E}\}$, $\Omega = \{\boldsymbol{\gamma}, \boldsymbol{\alpha}, \beta\}$, and $\Theta = \{\mathbf{z}, \Omega\}$. Mean field approximation in (28) is employed to the unknowns Θ and (29) is applied to find the estimates. Since the prior model is conjugate to the observation model, VB analysis can be carried out directly without invoking local variational bounds. The omitted technical details can be found in [21].

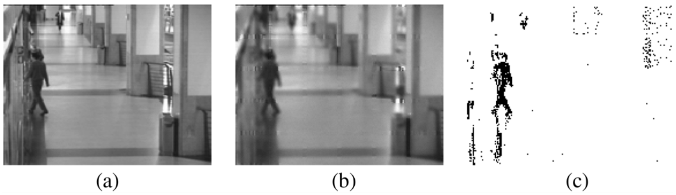


Fig. 10. Foreground detection from blurred and noisy video.

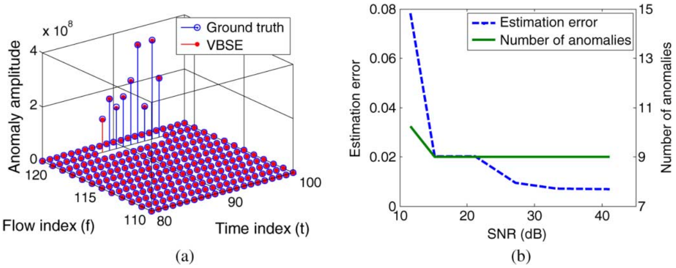


Fig. 11. Detection of network anomalies from Internet2 data.

The algorithm described above is termed VB Sparse Estimator (VBSE), whose performance is tested on real life datasets.

A. Video Foreground/Background Separation

In this experiment, the *CAVIAR* test video sequence was used. A sample video frame is presented in Fig. 10(a), showing the hallway in a shopping mall with people moving in the foreground. The original video frames were blurred using a radius-2 out-of-focus kernel. Dense Gaussian noise was added to the blurred video resulting in the observation with a Signal-to-Noise-Ratio (SNR) value at 23.5 dB. A blurred and noisy sample frame is shown in Fig. 10(b).

The presented VBSE algorithm was then applied on the blurred and noisy observation, and a result is shown in Figs. 10(c). From the figure we see that VBSE produces a clean foreground map that highlights the moving shoppers and their reflections on the ground. Also note that the VBSE approach is free of input parameters and is hence amenable to automated deployment.

B. Network Anomaly Detection

In this example, we use the dataset of Internet2 backbone network, which consists of $M = 121$ origin-destination flows and $L = 41$ links. The routing matrix $\mathbf{R}$ is provided along with the data set.

Fig. 11(a) illustrates the zoom-in view of the anomalies detected across the flows and time snapshots by VBSE. The regions not shown in the plots are all estimated to be zeros, in agreement with the ground truth. As can be seen in the figures, VBSE successfully detects the OD flow anomalies given the link measurements and accurately estimates their amplitudes.

To further investigate the algorithmic performance, we artificially added dense Gaussian noise to the link measurements. The performance of VBSE at various SNR levels is shown in

Fig. 11(b), where the estimation error and the estimated number of anomalies are plotted against the SNR levels. As can be seen, VBSE is able to precisely identify the number of anomalies as well as yields low estimation errors, even when significant noise is present.

APPENDIX C
IMAGE CLASSIFICATION AND ANNOTATION WITH HIERARCHICAL BAYESIAN MODEL AND VARIATIONAL INFERENCE

In this section we discuss an image classification and annotation problem, where the variational inference techniques presented in Section III and IV are applied. The data model and the algorithm presented in the following have been originally proposed in [54]. Here we paraphrase the problem and focus on the part where variational approximation is essential. Readers interested in more technical details are referred to [54] and the references therein.

A. Latent Dirichlet Allocation

In this subsection we present a hierarchical Bayesian model known as Latent Dirichlet Allocation (LDA) [55], [56], which is widely used for modeling the relationships among the observed "codewords" and the latent "topics" in "articles". Besides the classification problem presented herein, LDA modeling is widely utilized in multimedia problems (see, for instance, [57] for an application in text mining, and [58] for an application in video abnormal event detection).

In this example we focus on the supervised variant of LDA, known as supervised LDA. For image-based applications, an article is an image represented as a bag of N codewords $\mathbf{r} = \{r_n\}_{n=1}^N$, where each r_n is assumed to be drawn from a fixed codeword vocabulary. Similarly, each image is associated with M annotation terms $\mathbf{w} = \{w_m\}_{m=1}^M$ drawn from a fixed annotation vocabulary. In supervised LDA adopted for image classification, each image is assigned a class label $c \in \{1, \ldots, C\}$. For notational convenience, we denote $\mathbf{y} = \{\mathbf{r}, \mathbf{w}, c\}$ as the set of observations.

Besides the observation $\mathbf{y}$, consider K latent topics that govern the distributions of codewords and annotation terms. Specifically, $\{\boldsymbol{\pi}_k\}_{k=1}^K$ and $\{\boldsymbol{\beta}_k\}_{k=1}^K$ parameterize the multinomial codeword and annotation distributions, respectively. Moreover, each image has a K-dimensional topic prior $\boldsymbol{\theta}$ drawn from the Dirichlet distribution $\mathcal{D}(\boldsymbol{\alpha})$. Within an image, each region $n \in \{1, \ldots, N\}$ is independently associated with a topic $s_n \in \{1, \ldots, K\}$, which determines the multinomial codeword likelihood $\mathrm{p}(r_n|\boldsymbol{\beta}_{s_n})$. For each annotation term $m \in \{1, \ldots, M\}$ of an image, it is associated with a randomly chosen image region $f_m \in \{1, \ldots, N\}$, whose topic assignment s_{f_m} determines the multinomial annotation distribution $\mathrm{p}(w_m|\boldsymbol{\pi}_{s_{f_m}})$. Again for notational convenience, we denote the set of all latent variables as $\mathbf{z} = \{\boldsymbol{\theta}, \mathbf{s}, \mathbf{f}\}$, where $\mathbf{s} = \{s_n\}_{n=1}^N$ and $\mathbf{f} = \{f_m\}_{m=1}^M$.

In addition, define a topic weight vector $\boldsymbol{\eta}_c \in \mathbb{R}^K$ for each class c. As in [54], these weight vectors and the empirical topic frequency jointly determine the class distribution for an image. In the supervised LDA model, we consider $\{\boldsymbol{\eta}_c\}_{c=1}^C$, $\{\boldsymbol{\pi}_k\}_{k=1}^K$, $\{\boldsymbol{\beta}_k\}_{k=1}^K$ and $\boldsymbol{\alpha}$ as deterministically unknown, and denote them collectively as Ω.

B. Variational Approximate Inference

The approach employed to infer the latent $\mathbf{z}$ and estimate the parameters $\boldsymbol{\Omega}$ is known as variational EM algorithm. We present the high-level overview of the algorithm, while referring the interested readers to [54]–[56] for the omitted technical details.

The posterior distribution of the hidden $\mathbf{z}$ given an annotated image is

$$\mathrm{p}(\mathbf{z}|\mathbf{y}, \boldsymbol{\Omega}) = \frac{\mathrm{p}(\mathbf{z}, \mathbf{y}|\boldsymbol{\Omega})}{\mathrm{p}(\mathbf{y}|\boldsymbol{\Omega})}, \tag{C.1}$$

where the computation of the marginal likelihood or evidence $\mathrm{p}(\mathbf{y}|\boldsymbol{\Omega})$ in the denominator is intractable. To resolve this issue, we consider a convexity-based variational inference procedure. Define fully factorized variational distribution using mean field approximation as

$$\mathrm{q}(\boldsymbol{\theta}, \mathbf{s}, \mathbf{f}) = \mathrm{q}(\boldsymbol{\theta}|\gamma) \prod_{n=1}^{N} \mathrm{q}(s_n|\boldsymbol{\psi}_n) \prod_{m=1}^{M} \mathrm{q}(f_m|\boldsymbol{\phi}_m), \tag{C.2}$$

where $\gamma \in \mathbb{R}^K$ is a variational Dirichlet, $\boldsymbol{\psi}_n \in \mathbb{R}^K$ is a variational multinomial over K topics, and $\boldsymbol{\phi}_m \in \mathbb{R}^N$ is a variational multinomial over N image regions, respectively. Note that $\mathrm{q}(\boldsymbol{\theta}, \mathbf{s}, \mathbf{f})$ is a family of distributions indexed by the variational parameters, which are to be chosen via an optimization procedure to minimize the KL divergence between $\mathrm{q}(\boldsymbol{\theta}, \mathbf{s}, \mathbf{f})$ and the true posterior distribution $\mathrm{p}(\boldsymbol{\theta}, \mathbf{s}, \mathbf{f}|\mathbf{y}, \boldsymbol{\Omega})$. As is shown in Section III, such a minimization is equivalent to maximizing a lower bound of $\mathrm{p}(\mathbf{y}|\boldsymbol{\Omega})$.

The details of the iterative optimization procedure can be found in [53] and its references (in particular [55] and [56]). After the iterations converge, the optimal lower bound of $\mathrm{p}(\mathbf{y}|\boldsymbol{\Omega})$ is used to determine the approximate ML estimates of $\boldsymbol{\Omega}$. As a summary, the algorithm consists of an iterative optimization procedure for determining the optimal lower bound of the marginal likelihood $\mathrm{p}(\mathbf{y}|\boldsymbol{\Omega})$ (or equivalently finding the optimal approximation to the posterior distribution $\mathrm{p}(\mathbf{z}|\mathbf{y}, \boldsymbol{\Omega})$) and an ML estimation to determine the optimal values of the model parameters $\boldsymbol{\Omega}$.

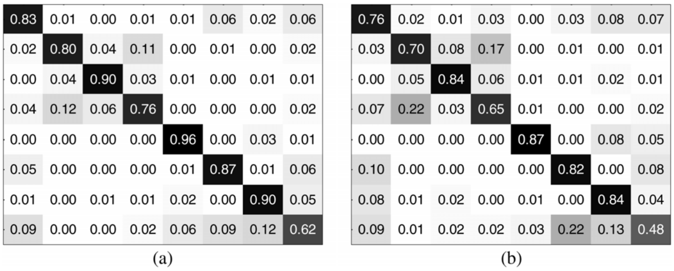
Note that for the image classification problem considered here, the training phase described above yields a set of model parameters $\boldsymbol{\Omega}$. For testing a new data point, the above approximate inference procedure is applied first to obtain the per image region topic distributions $\{\boldsymbol{\psi}_n\}_{n=1}^N$, which are averaged to yield a topic distribution per image, i.e., $\bar{\boldsymbol{\psi}} = \sum_n \boldsymbol{\psi}_n$. With the estimated parameter $\{\boldsymbol{\eta}_c\}_{c=1}^C$, the predicted class is given by

$$c^\star = \underset{c \in \{1, \ldots, C\}}{\arg\max}\ \boldsymbol{\eta}_c^{\mathrm{T}} \bar{\boldsymbol{\psi}}. \tag{C.3}$$

Finally, a distribution over the annotation terms is approximated as the averaged contribution from all image regions, given by

$$\mathrm{p}(w|\mathbf{r}, c) = \sum_{n=1}^{N} \mathrm{E}_{\mathrm{q}(s_n)} \left[\mathrm{p}(w|\boldsymbol{\beta}_{s_n})\right], \tag{C.4}$$

according to which the most probable annotation terms can be assigned to the image.



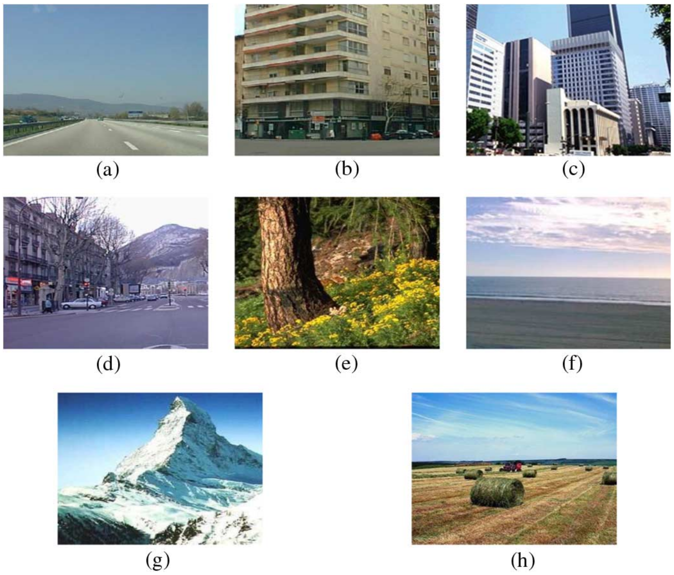
0.83	0.01	0.00	0.01	0.01	0.06	0.02	0.06
0.02	0.80	0.04	0.11	0.00	0.01	0.00	0.02
0.00	0.04	0.90	0.03	0.01	0.00	0.01	0.01
0.04	0.12	0.06	0.76	0.00	0.00	0.00	0.02
0.00	0.00	0.00	0.00	0.96	0.00	0.03	0.01
0.05	0.00	0.00	0.00	0.01	0.87	0.01	0.06
0.01	0.00	0.01	0.01	0.02	0.00	0.90	0.05
0.09	0.00	0.00	0.02	0.06	0.09	0.12	0.62

(a)

0.76	0.02	0.01	0.03	0.00	0.03	0.08	0.07
0.03	0.70	0.08	0.17	0.00	0.01	0.00	0.01
0.00	0.05	0.84	0.06	0.01	0.01	0.02	0.01
0.07	0.22	0.03	0.65	0.01	0.00	0.00	0.02
0.00	0.00	0.00	0.00	0.87	0.00	0.08	0.05
0.10	0.00	0.00	0.00	0.00	0.82	0.00	0.08
0.08	0.01	0.02	0.00	0.01	0.00	0.84	0.04
0.09	0.01	0.02	0.02	0.03	0.22	0.13	0.48

(b)

Fig. 12. Confusion matrices of image classification experiments.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 13. Examples of correctly classified images with annotations in the braces (adapted from Fig. 4 of [54]).

C. Examples

In this section we briefly demonstrate the application of the above variational inference and LDA model for image classification. The experiments were performed using the SLDA software made available by the authors of [53]. The LabelMe [59] data set consisting of images from $C = 8$ classes with annotations were used.

For this experiment, 1600 images were equally divided into a training set and a testing set. As an example, we set the number of topics to be $K = 50$ and set $\boldsymbol{\alpha}$ to be a vector with values 0.02.

The overall training and testing classification accuracies are 0.83 and 0.745, respectively. The confusion matrices are visualized in Fig. 12, where the (i, j)th element denotes the empirical frequency that images from class i are predicted to be from class j. Examples of correctly classified images and annotation terms are shown in Fig. 13. The labels assigned to the images correlate well with the objects contained in the images. For more details on the performance evaluation and comparison with other algorithms, the readers are referred to [54].

Due to space constraints, we were able to present only a few representative applications of VB. However, VB is widely applied to multimedia problems, such as speech recognition [60], [61], medical imaging [62], video processing [63], [64], etc.

Appendix D
Expectation Propagation For Gaussian Process Classification And Its Applications

A. Gaussian Process Classification

In this section we consider the application of the EP algorithm to a supervised classification problem. Given the training data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i$ is the input and y_i is the associated binary class label, the goal of Gaussian Process Classification is to learn a mechanism to assign class labels to unseen inputs.

A Gaussian Process (GP) [48] is an ensemble of functions with probabilities assigned to them. Every realization of a GP is a function $f(\mathbf{x})$ that maps the input $\mathbf{x}$ to a real number. The likelihood of the class label y associated with an input $\mathbf{x}$ is determined by

$$\mathrm{p}(y|f(\mathbf{x})) = \Phi(yf(\mathbf{x})) = \begin{cases} \Phi(f(\mathbf{x})), & y = +1 \\ \Phi(-f(\mathbf{x})), & y = -1 \end{cases}, \tag{D.1}$$

where $\Phi(\cdot)$ is a so-called squashing (or sigmoid) function.

Assuming independent data samples, it follows that the joint likelihood of $\mathbf{y} = \{y_i\}_{i=1}^N$ is given by

$$\mathrm{p}(\mathbf{y}|\mathbf{z}) = \prod_{i=1}^{N} \mathrm{p}(y_i|f_i), \tag{D.2}$$

where

$$\mathbf{z} = [f_1, f_2, \cdots, f_N]^{\mathrm{T}} = [f(\mathbf{x}_1), f(\mathbf{x}_2), \cdots, f(\mathbf{x}_N)]^{\mathrm{T}} \tag{D.3}$$

consists of latent functions drawn from a GP.

The GP definition implies that $\mathbf{z}$ in (D.3) follows a multivariate Gaussian distribution governed by its mean and covariance matrix. Assigning a prior distribution to the latent $\mathbf{z}$, we have

$$\mathrm{p}(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{K}), \tag{D.4}$$

where the covariance matrix $\mathbf{K}$ is calculated from $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$. Examples of $\mathbf{K}$ include those based on Radial Basis Functions (RBF) and Neural Networks (NN). The interested readers are referred to [48] for more details.

Note in the discussion here, prior and posterior are defined in relation to the observed class labels $\mathbf{y}$, not the inputs $\mathbf{X}$, which are assumed fixed. Therefore, to simplify notation we have suppressed the dependence on $\mathbf{X}$ in the equations.

As is the case for all Bayesian based algorithms, the classification is performed with the posterior distribution

$$\mathrm{p}(\mathbf{z}|\mathbf{y}) = \frac{1}{Z}\mathrm{p}(\mathbf{y}|\mathbf{z})\mathrm{p}(\mathbf{z}) = \frac{1}{Z}\mathrm{p}(\mathbf{z}) \prod_{i=1}^{N} \mathrm{p}(y_i|f_i), \tag{D.5}$$

where Z is the evidence of the observed class labels. However, due to the presence of non-Gaussian $\mathrm{p}(\mathbf{y}|\mathbf{z})$, the computation of Z is in general intractable. Therefore, we have to resort to either sampling based approaches or deterministic approximate inference methods. In the following we describe the application of the EP algorithm for approximating the posterior distribution.

B. EP for Gaussian Process Classification

The objective here is to approximate the posterior distribution $\mathrm{p}(\mathbf{z}|\mathbf{y})$. From the factorization in (D.5) we see one possible approach is to approximate each $\mathrm{p}(y_i|f_i)$ with an un-normalized Gaussian distribution, i.e.,

$$\mathrm{p}(y_i|f_i) \approx t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) = \tilde{Z}_i \mathcal{N}(f_i|\tilde{\mu}_i, \tilde{\sigma}_i^2) \tag{D.6}$$

The approximate joint posterior distribution of $\mathbf{z}$ given $\mathbf{y}$ is therefore given by

$$\mathrm{q}(\mathbf{z}|\mathbf{y}) = \frac{1}{Z_{\mathrm{EP}}} \mathrm{p}(\mathbf{z}) \prod_{i=1}^{N} t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) = \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}, \Sigma), \tag{D.7}$$

where Z_{EP} is the EP approximation to the normalization constant Z.

Given the approximation in (D.7), the objective then is to determine the parameters $\{\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2\}_{i=1}^{N}$ as well as Z_{EP}. The EP algorithm takes a "one-at-a-time" approach, where one set of parameters are updated while all the other parameters are kept at their most recent estimates.

Starting, for instance, with

$$t_i(f_i|\tilde{Z}_i, 0, v_i^2) = (2\pi v_i^2)^{1/2} \mathcal{N}(f_i|0, v_i^2), \tag{D.8}$$

where $v_i \to \infty$, the EP algorithm works by iterating through the factors indexed by $i \in \{1, \ldots, N\}$, and for each iteration the following procedure is carried out:

1) Obtain the approximate marginal posterior

$$\mathrm{q}(f_i|\mathbf{y}) = \mathcal{N}(f_i|\mu_i, \sigma_{ii}^2), \tag{D.9}$$

where μ_i is the ith element in $\boldsymbol{\mu}$ and σ_{ii}^2 is the ith element on the diagonal of $\boldsymbol{\Sigma}$, respectively.

2) Obtain the cavity probability distribution

$$\mathrm{q}_{-i}(f_i) = \mathcal{N}(f_i|\mu_{-i}, \sigma_{-i}^2) \tag{D.10}$$

by removing from $\mathrm{q}(f_i|\mathbf{y})$ the observation $t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2)$.

3) Incorporate to $\mathrm{q}_{-i}(f_i)$ the true likelihood $\mathrm{p}(y_i|f_i)$ to obtain the distribution

$$\mathrm{r}(f_i) = \mathrm{q}_{-i}(f_i)\mathrm{p}(y_i|f_i)/Z_i \tag{D.11}$$

where

$$Z_i = \int \mathrm{q}_{-i}(f_i)\mathrm{p}(y_i|f_i)\mathrm{d}f_i. \tag{D.12}$$

4) Find a Gaussian approximation $\hat{\mathrm{q}}(f_i)$ to $\mathrm{r}(f_i)$ in (D.11) by minimizing the KL divergence over the set $\mathcal{G}$ of Gaussian distributions

$$\hat{\mathrm{q}}(f_i) = \arg\min_{\mathrm{q}(f_i)\in\mathcal{G}} \mathrm{KL}(\mathrm{r}(f_i)\|\mathrm{q}(f_i)), \tag{D.13}$$

which is in turn solved via the moment matching procedure.

5) From

$$Z_i\hat{\mathrm{q}}(f_i) \approx \mathrm{q}_{-i}(f_i)p(y_i|f_i) = \mathcal{N}(f_i|\mu_{-i}, \sigma_{-i}^2)\mathrm{p}(y_i|f_i) \tag{D.14}$$

obtain

$$t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) = \frac{Z_i\hat{\mathrm{q}}(f_i)}{\mathcal{N}(f_i|\mu_{-i}, \sigma_{-i}^2)} = \tilde{Z}_i\mathcal{N}(f_i|\tilde{\mu}_i, \tilde{\sigma}_i^2) \tag{D.15}$$

TABLE II
GPC WITH EP ON HYPERSPECTRAL IMAGE CLASSIFICATION

Data set	No. of training & test samples	No. of classes	GPC (RBF)	GPC (NN)	SVM
El Tarf	100/100	5	95.2	96.0	93.8
Po	100/120	5	92.7	93.8	92.8
AVIRIS	1800/4588	9	87.8	87.5	87.7

6) Update $\mathrm{q}(\mathbf{z}|\mathbf{y})$ in (D.7) using the newly obtained $t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2)$, where Z_{EP} is computed as the normalization constant. Since all terms in the product are Gaussians, this determination is tractable. This concludes one iterate in the EP algorithm, and in the next iteration, another factor is updated.

When the EP iterations converge, the approximate joint posterior $\mathrm{q}(\mathbf{z}|\mathbf{y})$ is available for use in predicting the class label y_* for an unseen input $\mathbf{x}_*$. Specifically, this is done via the following two-step procedure:

1) Obtain the approximate posterior distribution

$$\mathrm{q}(f_*|\mathbf{y}, \mathbf{x}_*) = \int \mathrm{p}(f_*|\mathbf{z}, \mathbf{x}_*)\mathrm{q}(\mathbf{z}|\mathbf{y})\mathrm{d}\mathbf{z}. \tag{D.16}$$

2) Compute the probability of class label y_* associated with $\mathbf{x}_*$ by marginalizing the latent f_*

$$\begin{aligned}\mathrm{p}(y_*|\mathbf{y}, \mathbf{x}_*) &= \int \mathrm{p}(y_*|f_*)\mathrm{q}(f_*|\mathbf{y}, \mathbf{x}_*)\mathrm{d}f_* \\ &= \int \Phi(y_* f_*)\mathrm{q}(f_*|\mathbf{y}, \mathbf{x}_*)\mathrm{d}f_*. \end{aligned} \tag{D.17}$$

C. Hyperspectral image classification

In this section we consider a remote sensing image classification problem in which the GPC with EP approximations is applied. This problem is discussed in [65], where GPC is compared with the state-of-the-art support vector machine (SVM) algorithm.

Three data sets were used in the experiments in [65]. The information of the data sets is summarized in Table II.

As is reported in [65] and [66], GPC with EP approximation yields similar or even higher classification accuracies compared with SVM. Table II shows the overall accuracies obtained from GPC with EP and SVM, respectively. In the table RBF and NN denote two the types of prior covariance matrix used in the definition of the Gaussian process. As can be seen, the GPC with EP approximation performs similarly as or even better than the state-of-the-art classification algorithm.

REFERENCES

[1] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[2] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

[3] S. J. D. Prince, *Computer Vision: Models, Learning, and Inference*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[4] P. Chen and D. Suter, "Recovering the missing components in a large noisy low-rank matrix: Application to SFM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1051–1063, 2004.

[5] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010, pp. 1791–1798.

[6] J. P. Haldar and Z. Liang, "Spatiotemporal imaging with partially separable functions: A matrix recovery approach," in *Proc. IEEE Int. Symp. Biomedical Imaging: From Nano to Macro*, 2010, pp. 716–719.

[7] N. Srebro, "Learning with matrix factorization," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, USA, 2004.

[8] Z. Liu and L. Vandenberghe, "Interior-point method for nuclear norm approximation with application to system identification," *SIAM J. Matrix Anal. Applicat.*, vol. 31, no. 3, pp. 1235–1256, 2009.

[9] S. Oh, A. Montanari, and L. Karbasi, "Sensor network localization from local connectivity: Performance analysis for the MDS-MAP algorithm," in *Proc. IEEE Information Theory Workshop*, 2010, pp. 1–5.

[10] J. Wang, Y. Shi, W. Ding, and B. Yin, "A low-rank matrix completion based intra prediction for H.264/AVC," in *Proc. IEEE 13th Int. Workshop Multimedia Signal Processing*, 2011, pp. 1–6.

[11] N. Barzigar, A. Roozgard, S. Cheng, and P. Verma, "An efficient video denoising method using decomposition approach for low-rank matrix completion," in *Conf. Record 46th Asilomar Conf. Signals, Systems and Computers*, 2012, pp. 1684–1687.

[12] M. D. Dao, D. T. Nguyen, Y. Cao, and T. D. Tran, "Video concealment via matrix completion at high missing rates," in *Proc. Asilomar Conf. Signals, Systems and Computers*, 2010, pp. 758–762.

[13] D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos, "Sparse Bayesian methods for low-rank matrix estimation," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 3964–3977, Aug. 2012.

[14] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, and Y. Ma, "Toward a practical face recognition system: Robust alignment and illumination by sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 2, pp. 372–386, 2012.

[15] T. Goodall, S. Gibson, and M. C. Smith, "Parallelizing principal component analysis for robust facial recognition using CUDA," *Proc. Symp. App. Accelerators in High Perf. Comput.*, pp. 121–124, 2012.

[16] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala, "Latent semantic indexing: A probabilistic analysis," *Elsevier J. Comput. Syst. Sci.*, vol. 61, no. 2, pp. 217–235, 2000.

[17] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, 2010, pp. 763–770.

[18] P. Huang, S. D. Chen, P. Smaragdis, and M. Hasegawa-Johnson, "Singing-voice separation from monaural recordings using robust principal component analysis," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2012, pp. 57–60.

[19] W. Tan, G. Cheung, and Y. Ma, "Face recovery in conference video streaming using robust principal component analysis," in *Proc. 18th IEEE Int. Conf. Image Processing*, 2011, pp. 3225–3228.

[20] X. Wang, W. Wan, and G. Liu, "Multi-task low-rank and sparse matrix recovery for human motion segmentation," in *Proc. 19th IEEE Int. Conf. Image Processing*, 2012, pp. 897–900.

[21] Z. Chen, R. Molina, and A. K. Katsaggelos, "A variational approach for sparse component estimation and low-rank matrix recovery," *J. Commun.*, vol. 8, no. 9, pp. 600–611, 2013.

[22] M. Beal, "Variational algorithms for approximate Bayesian inference," Ph.D. dissertation, Univ. College London, London, U.K., 2003.

[23] J. Miskin, "Ensemble learning for independent component analysis," Ph.D. dissertation, Astrophysics Group, Univ. Cambridge, Cambridge, U.K., 2000.

[24] C. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[25] S. T. Jaakkola and I. M. Jordan, "Bayesian parameter estimation via variational methods," *Statist. Comput.*, vol. 10, no. 1, pp. 25–37, 2000.

[26] D. G. Tzikas, C. L. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Process. Mag.*, vol. 25, no. 6, pp. 131–146, 2008.

[27] C. W. Fox and S. J. Roberts, "A tutorial on variational Bayesian inference," *Artif. Intell. Rev.*, vol. 38, no. 2, pp. 85–95, 2012.

[28] S. Dupont and J. Luettin, "Audio-visual speech modeling for continuous speech recognition," *IEEE Trans. Multimedia*, vol. 2, no. 3, pp. 141–151, 2000.

[29] A. V. Nefian, L. Liang, X. Pi, X. Liu, and K. P. Murphy, "Dynamic Bayesian networks for audio-visual speech recognition," *EURASIP J. Adv. Signal Process.*, vol. 2002, no. 11, pp. 1274–1288, 2002.

[30] Z. Liu, F. Jiang, G. Tian, S. Wang, F. Sato, S. Meltzer, and M. Tan, "Sparse logistic regression with l_p penalty for biomarker identification," *Statist. App. Genet. Molec. Biol.*, vol. 6, no. 1, 2007.

[31] A. Kabán and R. Durrant, "Learning with $L_{q<1}$ vs L_1-norm regularisation with exponentially many irrelevant features," *Mach. Learn. and Knowl. Discov. Databases, Lecture Notes in Comput. Sci.*, vol. 5211, pp. 580–596, 2008.

[32] D. Koller and N. Friedman, *Probabilistic Graphical Models - Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.

[33] M. I. Jordan, Z. Ghahramani, T. S. Jaakola, and L. K. Saul, "An introduction to variational methods for graphical models," in *Learning in Graphical Models*. Cambridge, MA, USA: MIT Press, 1998, pp. 105–162.

[34] S. S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, pp. 721–741, 1984.

[35] J. Grimmer, "An introduction to Bayesian inference via variational approximations," *Polit. Anal.*, vol. 19, no. 1, pp. 32–47, 2010.

[36] S. D. Babacan, R. Molina, M. N. Do, and A. K. Katsaggelos, "Bayesian blind deconvolution with general sparse image priors," in *Proc. Eur. Conf. Computer Vision*, 2012, pp. 341–355.

[37] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.

[38] T. Buchgraber, "Variational sparse Bayesian learning: Centralized and distributed processing," Ph.D. dissertation, Graz Univ. Technol., Graz, Austria, 2013.

[39] A. D. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the E-M algorithm," *J. Roy. Statist. Soc., Series B*, vol. 39, pp. 1–37, 1977.

[40] G. Parisi, *Statistical Field Theory*. Reading, MA, USA: Addison-Wesley, 1988.

[41] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[42] R. T. Rockafellar, *Convex analysis*. Princeton, NJ, USA: Princeton Univ. Press, 1970.

[43] J. A. Palmer, K. Kreutz-Delgado, D. P. Wipf, and B. D. Rao, "Variational EM algorithms for non-Gaussian latent variable models," in *Advances in Neural Information Processing Systems 18*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA, USA: MIT Press, 2006, pp. 1059–1066.

[44] D. J. C. MacKay, *Information Theory, Inference & Learning Algorithms*. New York, NY, USA: Cambridge Univ. Press, 2007.

[45] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA, USA: Morgan Kaufmann, 1988.

[46] T. P. Minka, "A family of algorithms for approximate Bayesian inference," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, USA, 2001.

[47] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proc. Conf. Uncertainty in Artificial Intelligence*, 2001, pp. 362–369.

[48] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[49] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational Bayesian blind deconvolution using a total variation prior," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 12–26, Jan. 2009.

[50] A. M. McIvor, "Background subtraction techniques," in *Proc. Image and Vision Computing New Zealand 2000, Reveal Limited*, Auckland, New Zealand.

[51] M. Piccardi, "Background subtraction techniques: A review," in *Proc. IEEE Int. Conf. Syst., Man and Cybernetics*, 2004, vol. 4, pp. 3099–3104.

[52] A. Patcha and J. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Elsevier Comput. Netw.*, vol. 51, no. 12, pp. 3448–3470, 2007.

[53] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surveys*, vol. 41, no. 3, pp. 15:1–15:58, 2009.

[54] C. Wang, D. Blei, and F. Li, "Simultaneous image classification and annotation," in *Proc. IEEE Comput. Vision & Pattern Recognit.*, 2009, pp. 1903–1910.

[55] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[56] D. M. Blei and J. D. McAuliffe, "Supervised topic models," in *Advances in Neural Information Processing Systems 20*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA, USA: MIT Press, 2008, pp. 121–128.

[57] J. Zeng, W. K. Cheung, and J. Liu, "Learning topic models by belief propagation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 5, pp. 1121–1134, 2013.

[58] X. Wang, X. Ma, and W. E. L. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 3, pp. 539–555, 2009.

[59] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image annotation," *Int. J. Comput. Vision*, vol. 77, no. 1-3, pp. 157–173, 2008.

[60] S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda, "Variational Bayesian estimation and clustering for speech recognition," *IEEE Trans. Speech Audio Process.*, vol. 12, no. 4, pp. 365–381, 2004.

[61] Y. Tam and T. Schultz, "Dynamic language model adaptation using variational Bayes inference," in *Proc. INTERSPEECH*, 2005, pp. 5–8.

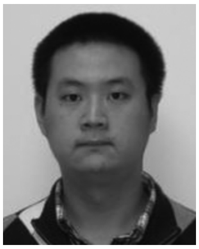
[62] W. Penny, S. Kiebel, and K. Friston, "Variational Bayesian inference for fMRI time series," *NeuroImage*, vol. 19, no. 3, pp. 727–741, 2003.

[63] X. Li and Y. Zheng, "Patch-based video processing: A variational Bayesian approach," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 1, pp. 27–40, 2009.

[64] I. Pruteanu-Malinici and L. Carin, "Infinite hidden Markov models for unusual-event detection in video," *IEEE Trans. Image Process.*, vol. 17, no. 5, pp. 811–822, May 2008.

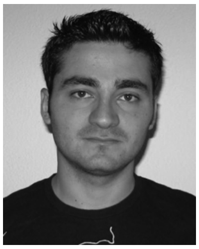
[65] Y. Bazi and F. Melgani, "Gaussian process approach to remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 1, pp. 196–197, 2010.

[66] Y. Bazi and F. Melgani, "Classification of hyperspectral remote sensing images using Gaussian processes," in *Proc. IEEE Int. Geoscience and Remote Sensing Symp.*, 2008, pp. 1013–1016.



Zhaofu Chen received the B.Sc. degree in information science and engineering from Zhejiang University, Hangzhou, China, in 2008, and the M.Sc. degree in electrical and computer engineering from University of Florida, Gainesville, FL, in 2010. He is currently a Ph.D. candidate in electrical engineering and computer science at Northwestern University, Evanston, IL.

In 2010, he joined the Image and Video Processing Lab at Northwestern University, where he is currently a graduate research assistant. His research interests include statistical analysis of sparse signals with applications in networking and multimedia.



S. Derin Babacan (M'10) received the B.Sc. degree from the Electrical and Electronics Department, Bogazici University, Istanbul, Turkey, in 2004, and the M.Sc. and Ph.D. degrees from the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, in 2006 and 2009, respectively.

Between 2010–2012, he was a Beckman Postdoctoral Fellow at the Beckman Institute for Advanced Science and Technology at the University of Illinois at Urbana-Champaign, and he is currently at Google, Inc. His primary research interests are inverse problems in image processing, compressive sensing and computational photography. He is the recipient of an IEEE International Conference on Image Processing Paper Award in 2007.

Rafael Molina (M'88) was born in 1957. He received the Degree in mathematics (statistics) in 1979 and the Ph.D. degree in optimal design in linear models in 1983. He became a Professor of Computer Science and Artificial Intelligence with University of Granada, Spain in 2000. He was a Former Dean of the Computer Engineering School, University of Granada (1992–2002), and the Head of the Computer Science and Artificial Intelligence Department, University of Granada (2005–2007).

His current research interests include using Bayesian modeling and inference in problems like image restoration (applications to astronomy and medicine), super resolution of images and video, blind deconvolution, computational photography, source recovery in medicine, compressive sensing, low rank matrix decomposition, active learning, and classification.

Dr. Molina serves the IEEE and other Professional Societies, including Applied Signal Processing. He was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING (2005–2007), an Associate Editor of Progress in Artificial Intelligence (2010), an Associate Editor of Digital Signal Processing (2011), and an Area Editor (2011). He is a recipient of the IEEE ICIP Paper Award in 2007, the ISPA Best Paper Award in 2009, and the EUSIPCO Best Student Paper Award in 2013. He co-authored a paper that received the Runner-Up Prize at Reception for early-stage researchers at the House of Commons.

Aggelos K. Katsaggelos (F'98) received the Diploma degree in electrical and mechanical engineering from the Aristotelian University of Thessaloniki, Greece, in 1979, and the M.S. and Ph.D. degrees in Electrical Engineering from the Georgia Institute of Technology, in 1981 and 1985, respectively.

In 1985, he joined the Department of Electrical Engineering and Computer Science at Northwestern University, where he is currently a Professor holder of the AT&T chair. He was previously the holder of the Ameritech Chair of Information Technology (1997–2003). He is also the Director of the Motorola Center for Seamless Communications, a member of the Academic Staff, NorthShore University Health System, an affiliated faculty at the Department of Linguistics and he has an appointment with the Argonne National Laboratory.

He has published extensively in the areas of multimedia signal processing and communications (over 200 journal papers, 500 conference papers and 40 book chapters) and he is the holder of 21 international patents.

Among his many professional activities Prof. Katsaggelos was Editor-in-Chief of the IEEE Signal Processing Magazine (1997–2002), a BOG Member of the IEEE Signal Processing Society (1999–2001), and a member of the Publication Board of the IEEE Proceedings (2003–2007). He is a Fellow of the IEEE (1998) and SPIE (2009) and the recipient of the IEEE Third Millennium Medal (2000), the IEEE Signal Processing Society Meritorious Service Award (2001), the IEEE Signal Processing Society Technical Achievement Award (2010), an IEEE Signal Processing Society Best Paper Award (2001), an IEEE ICME Paper Award (2006), an IEEE ICIP Paper Award (2007), an ISPA Paper Award (2009), and a EUSIPCO Paper Award (2013). He was a Distinguished Lecturer of the IEEE Signal Processing Society (2007–2008).