Int J Comput Vis (2016) 116:136–160
DOI 10.1007/s11263-015-0834-9

A Joint Gaussian Process Model for Active Visual Recognition with Expertise Estimation in Crowdsourcing

Chengjiang Long[1] · **Gang Hua**[1] · **Ashish Kapoor**[2]

Received: 4 June 2014 / Accepted: 1 June 2015 / Published online: 11 June 2015

Abstract We present a noise resilient probabilistic model for active learning of a Gaussian process classifier from crowds, i.e., a set of noisy labelers. It explicitly models both the overall label noise and the expertise level of each individual labeler with two levels of flip models. Expectation propagation is adopted for efficient approximate Bayesian inference of our probabilistic model for classification, based on which, a generalized EM algorithm is derived to estimate both the global label noise and the expertise of each individual labeler. The probabilistic nature of our model immediately allows the adoption of the prediction entropy for active selection of data samples to be labeled, and active selection of high quality labelers based on their estimated expertise to label the data. We apply the proposed model for four visual recognition tasks, i.e., object category recognition, multi-modal activity recognition, gender recognition, and fine-grained classification, on four datasets with real crowd-sourced labels from the Amazon Mechanical Turk. The experiments clearly demonstrate the efficacy of the proposed model. In addition, we extend the proposed model with the Predictive Active Set Selection Method to speed up the active learning system, whose efficacy is verified by conducting experiments on the first three datasets. The results show our extended model can not only preserve a higher accuracy, but also achieve a higher efficiency.

Keywords Active learning · Crowdsourcing · Gaussian process classifiers

1 Introduction

As research on visual recognition evolves gradually towards an experimental science, partly due to the success of the introduction of the machine learning approach to computer vision (Burl et al. 1995; Burl and Perona 1998; Fergus et al. 2005; Liu et al. 2008; Sanchez and Perronnin 2011; Krizhevsky et al. 2012; Lin et al. 2011), collecting labeled visual datasets at large scale from crowd-sourcing tools such as the Amazon Mechanical Turk has become a common practice (Deng et al. 2009; Vijayanarasimhan and Grauman 2014). Although it is cheap to obtain a large quantity of labels through crowdsourcing, it has been well known that the collected labels could be very noisy. So it is desirable to model the expertise levels of the labelers to ensure the quality of the labels (Deng et al. 2009; Vijayanarasimhan and Grauman 2014; Ambati et al. 2010). The higher the expertise level a labeler is at, the lower the label noises he/she will produce.

Previous works for modeling the labelers' expertise mainly adopted two approaches. The first approach attempts to evaluate the labelers by adopting a pre-labeled gold standard dataset (Ambati et al. 2010). When a labeler is constantly generating contradicting labels on data samples from the gold standard dataset, all labels from that labeler may be discarded as he/she is highly likely to be an irresponsible one. The second approach addresses this issue through evaluating the labels by collecting multiple labels for each data sample (Deng et al. 2009; Vijayanarasimhan and Grau-

Communicated by Jakob Verbeek.

✉ Gang Hua
ganghua@gmail.com; ghua@stevens.edu

Chengjiang Long
clong@stevens.edu

Ashish Kapoor
akapoor@microsoft.com

[1] Stevens Institute of Technology, Hoboken, NJ 07030, USA

[2] Microsoft Research, Redmond, WA 98052, USA

man 2014). Then online or postmortem majority voting, or majority model consistency check is conducted to obtain the more likely ground-truth label of the data sample. The basic assumption is that the majority of the labelers are behaving in good faith.

The first approach is able to evaluate the labelers online, which is desirable. But it needs to pre-label a set of data to serve as the gold standard, which may be an obstacle by itself. The second approach focuses on the label noise. It does not explicitly evaluate the labelers, although it may be extended to do so by online tracking how often a labeler is contradicting with the majority. Notwithstanding their demonstrated success, these two approaches are rather *ad hoc*. There lacks a principled approach to jointly model the global noise level of the labels and the expertise level of each individual labeler, in the absence of gold standard labels, which is what we want to achieve in this paper.

We present a Bayesian model which explicitly models the global noise level of the labels and the expertise level of each individual labeler from crowds (i.e., a group of noisy labelers). These two different statistics are modeled hierarchically with two levels of flip models (Minka 2001). Expectation Propagation (EP) (Minka 2001) is adopted to conduct approximate Bayesian inference of the posterior of the latent classification function. A generalized Expectation Maximization (GEM) algorithm is developed to estimate both quantities. The resulting classifier is more resilient to label noises, adapting to the expertise of labelers.

Another potential improvement that can be made to current crowdsourcing labeling system such as Amazon Mechanical Turk (AMT) is to actively guide the labelers for more efficient labeling. The proposed Bayesian model enables not only active selection of data samples to be labeled, but also active selection of quality labelers. These are enabled by the probabilistic nature of our model and the explicit modeling of both the global label noise and the expertise of each individual labeler, thereby allowing entropy based uncertainty measure to be readily adopted for these purposes.

Therefore, the proposed Bayesian model is able to actively collect multiple copies of crowd-sourced labels with minimal human efforts. As recapped in Fig. 1, the main loop consists of (1) a general Expectation Maximization procedure with an embedded Expectation Propagation algorithm to learn a Gaussian process classifier by a joint treatment of multiple copies of labels, (2) ranking all the candidate unlabeled samples based on the prediction entropy, (3) assigning the actively selected sample to the top K quality online labelers, (4) incorporating their responses to obtain new labeled data, and (5) retraining the classifier.

In order to obtain high classification accuracy with computational cost as low as possible, we extend our proposed model with the Predictive Active Set Selection Method Henao and Winther (2010; 2012). Firstly we randomly select a subset from the whole labeled data and consider it as an initial "active set". And then we iteratively update the active set based on the predictive/cavity distribution inferred by a Gaussian process classifier newly trained upon the previous active set. At each iteration, hyperparameter optimization is carried out on the whole labeled data. We alternate between active set updates and hyperparameter estimation through several passes over the whole labeled data. From this perspective, we still make full use of all the labeled information.

Several aspects distinguish our work from previous active learning based labeling (Vijayanarasimhan and Grauman 2014), Kapoor et al. (2007, 2009), (Loy et al. 2012; Ebert et al. 2012). First of all, our work deals with active learning with multiple labelers, a topic which has not been sufficiently explored before. Secondly, we do not assume that the labels provided by the labelers are absolutely correct. In other words, the labeler may label an example incorrectly. Most previous work on active learning has assumed that the labels provided by the human oracle are noise free. Thirdly, our model allows online evaluation of the expertise of the labelers without relying on any additional pre-labeled gold standard data. Hence we can select more responsible labelers and reduce the noise level of the labels we collected.

The main contributions of this paper are: (1) a Bayesian probabilistic formulation to learn a Gaussian process classifiers from multiple noisy labels, which models both the global label noise and the expertise of each individual labeler; (2) an active learning system which determines which users to label which unlabeled examples; and (3) an extended model with the Predictive Active Set Selection Method to speed up the active learning system. We apply our proposed model on datasets with real noisy labels obtained from the Amazon Mechanical Turk on four visual recognition tasks, i.e., object category recognition, gender recognition, multimodal activity recognition, and fine-grained classification. The results clearly demonstrate the efficacy of our proposed model.

We shall point out that the foundation of the proposed algorithm in this paper is firstly published in our paper in Long et al. (2013). This paper extends our initial work in several ways: (1) we have added more detailed discussion of the technical formulation and algorithm derivation; (2) the computational efficiency of the proposed algorithm is improved by leveraging a predictive active set selection method; (3) the recognition accuracy of the proposed algorithm is improved by exploiting a parameterized kernel in the learning process; (4) the selection of labelers is enabled in the experiments with real crowd-sourced labels on the visual datasets; (5) we add experiments on a fine-grained visual category recognition dataset to further validate the efficacy of the proposed method; (6) we compare the proposed model with the competing models using all the labeled data; and (7) we also

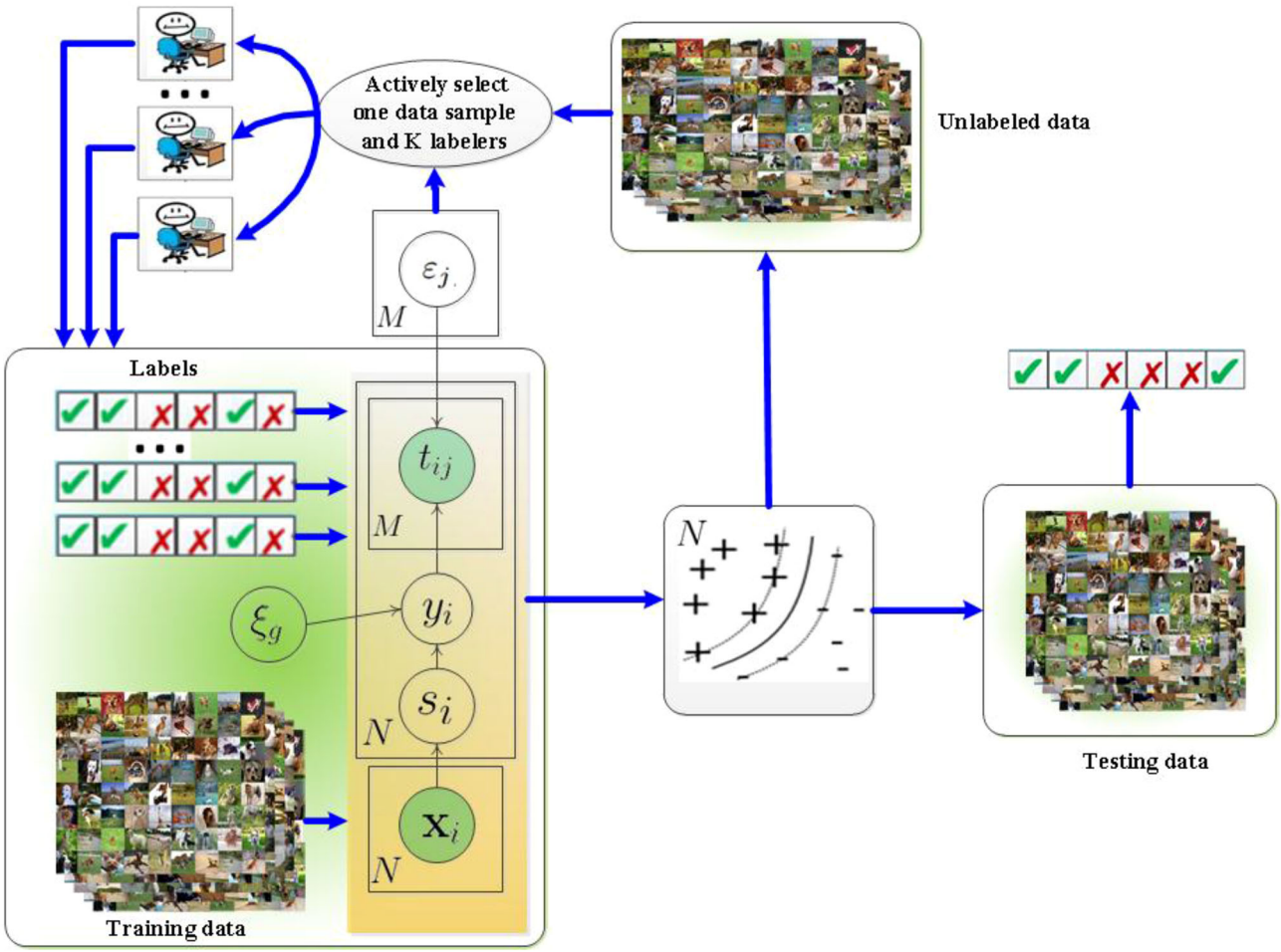

Fig. 1 Illustration of our active visual recognition system with expertise estimation in Crowdsourcing

evaluate the proposed method with different number of initial labeled examples.

The rest of the paper is arranged as follows: Sect. 2 reviews the related work; Sect. 3.1 presents our Bayesian probabilistic model; Sect. 3.2 describes the inference with Expectation Propagation; In Sect. 3.3, we discuss the EM-EP to infer both the soft scores for the data points and the label ratings for the labelers. In Sect. 4, we introduce the active learning strategy. Our model is extended with the Predict Active Set Selection Method to speed up the active learning system as stated in Sect. 5. The experiments are presented in Sect. 6 and further discussions are taken in Sect. 7. Finally we draw the conclusion and present potential future work in Sect. 8.

2 Related Work

Related works can be grouped into five categories including Human in the loop for computer vision, noise resilient Gaussian process classifiers, approximate Bayesian inference methods, sparse approximations in Gaussian process classification, and active learning with crowd-sourced labels.

2.1 Human in the Loop for Computer Vision

Ever since the publication of the ESP games (von Ahn and Dabbish 2004; von Ahn et al. 2006) for generating annotations for images, there have been a lot of active research in the computer vision community to harvest human knowledge from crowds, and in particular, research in visual recognition. For example, Parikh and Zitnick (2010, 2011), Parikh (2011), Zitnick and Parikh (2012), Parikh et al. (2012) have studied various factors in a visual recognition system using crowd-sourced human debugging, including the impacts of features, algorithms, and data (Parikh and Zitnick 2010), the weakest link in a person detector (Parikh and Zitnick 2011), the role of local and global information (Parikh 2011), the role of contour in image understanding (Zitnick and Parikh 2012), and the role of appearance and contextual information (Parikh et al. 2012) for image recognition.

Some other representative works engaging human in the loop for visual recognition include the Visipedia project Branson et al. (2010, 2011), (Wah et al. 2011; Welinder and Perona 2010a), which studies how to build systems and models to engage human (e.g., those from crowds) in various recognition tasks, either in terms of questions and answers, or relabeling. Some more recent work also studies how to bootstrap a fine-grained visual recognition system by actively querying answers from crowds with binary questions (Patterson et al. 2013), and identify discriminative features for more accurate fine-grained visual category recognition using the Bubble game (Deng et al. 2013). There also exists work on interactive object detection and tracking that use active learning (Yao et al. 2012; Vondrick and Ramanan 2011). Most of these works just focus on modeling the output from crowds. They do not attempt to further model the individual expertise of each human ocracle in the modeling and learning process while analyzing the visual content.

2.2 Noise Resilient Gaussian Process Classifiers

Gaussian process classifier (GPC) is a family of nonparametric Bayesian kernel classifiers derived from the Gaussian process priors over hidden functions, with the signs of those continuous outputs to determine the class labels (Williams and Barber 1998; Gibbs and Mackay 2000). A Gaussian process classifier is composed of a Gaussian process prior and a likelihood model which defines the probability of the data label given the hidden function. Given a set of labeled data samples, the output of the hidden functions of these labeled samples would be constrained. To predict the label of a new data, one needs to integrate over all the values of the hidden functions over the labeled data points, which are often intractable when the likelihood model is not Gaussian. A noise resilient likelihood model, namely flip noise model, is introduced in Minka (2001) to better handle label noises in the Gaussian process classifier. More recently, (Kim and Ghahramani 2008) exploited the flip noise model to explicitly handle outlier labels using a Gaussian process classifier.

2.3 Approximate Bayesian Inference Methods

Various approximate inference algorithms have been proposed to solve the inference problem in Gaussian process when the exact inference is analytically intractable.

For example, Williams and Barber adopted Laplace approximation (Williams and Barber 1998) to approximate the posterior as a multi-variate Gaussian. The integral is hence replaced by the mode of the Gaussian. Neal (1997) resorted to Markov chain Monte Carlo to approximate the integral to the mean over a set of samples generated from the posterior distribution. Variational approximation is adopted by Gibbs and Mackay (2000). Opper and Winther (1999) adopted the TAP (Thouless et al. 1977) style mean field approximation to obtain the integral. In the case of Gaussian process classifier, the TAP style mean field approximation (Opper and Winther 1999) is equivalent to the more general Expectation Propagation for approximate inference, which is firstly proposed by Minka (2001). An EM algorithm built on top of EP is proposed to estimate the label noise levels. None of these aforementioned methods ever considered the case where a data sample has multiple copies of noisy labels, which is the focus of our proposed approach.

2.4 Sparse Approximations in Gaussian Process Classification

The computational cost of inference in a Gaussian process classifier scales cubically with the size of the training set, which makes it not ideal for large datasets. A considerable amount of research efforts focus on sparse approximations (Quinonero-candela et al. 2005; Lawrence et al. 2003; Seeger 2002; Naish-Guzman and Holden 2007; Seeger et al. 2003; Snelson and Ghahramani 2006a; Titsias 2009; Yan and Qi 2010) to address this issue. Generally, the existing techniques attempt to reduce the computational cost from $O(N^3)$ to $O(NN_{ws}^2)$, where $N_{ws} < N$, N is the size of the whole training data and N_{ws} is the size of a working set consisting of a subset of the training data or a pseudo-input set (Snelson and Ghahramani 2006a).

The principle of determining a subset from the entire training data is to preserve those more informative data points that contribute more to high classification accuracy and to discard those less informative ones. In contrast, building a pseudo-input set attempts to reduce the difference in distribution between the classifier using all N points and the one using only M points, by estimating the location of an auxiliary set in the input space. However, as the size of the auxiliary set increases, more and more parameters need to be learned, which makes it infeasible for large scale datasets (Snelson and Ghahramani 2006b). Worse still, the latter approach is sensitive to overfitting as a result of the large number of free parameters in the model.

Some recent research work studies from the perspective of a Bayesian framework. Zhang et al. (2011) developed an efficient MCMC algorithm, in which sparsity is enforced in an explicit treatment and a full Bayesian method is carried out. The main computational burden is therefore reduced to be similar to or the same as other sparse kernel methods possibly with a larger pre-factor due to sampling. Henao and Winther (2010, 2012) proposed a framework, namely, the *Predictive Active Set Selection Method for Gaussian Process* (PASS-GP), which uses an active subset of training data to learn a GPC. The subset is "active"

because it is iteratively updated based on the relative importance of each data point, which is measured with the cavity/predictive distribution inferred by the newly learned GPC.

2.5 Active Learning with Crowd-Sourced Labels

Several previous works have explored active learning from noisy crowd-sourced labels (Ambati et al. 2010; Vijayanarasimhan and Grauman 2014) in different domains, where the two aforementioned approaches are exploited to handle label noise.

To better mitigate label noises online in the absence of gold standard labels, Donmez et al. (2009, 2010) have explored confidence interval based estimation and the sequential Bayesian estimation method to evaluate the label quality of the annotators in both stationary and non-stationary cases. Zhao et al. (2011) proposed an incremental relabeling mechanism which employed active learning to not only select the unlabeled data to be labeled by crowds, but also select already labeled data samples to be relabeled until sufficient confidence is built. Raykar et al. (2009), Raykar and Yu (2012) proposed a probabilistic model, which assumes independence of the annotator judgement given the true label, and alternatively conducts model learning and performance evaluation of the multiple annotators.

Dekel and Shamir (2009a) adapted the formulation of support vector machines (SVMs) to identify low quality or malicious annotators, which assumes that each annotator is either good or bad. Later, they (Dekel and Shamir 2009b) proposed a method for pruning low-quality labelers by using the model trained from the entire labeled dataset from all labelers as ground truth. Chen et al. (2010) proposed to identify good annotators by spectral clustering in the worker space. The assumption is that good labelers will behave similarly. Yan et al. (2012, 2011) presented a Bayesian model and adopted a logistic regression function to model the labelers' quality. Simpson et al. (2013) recently tackled the problem of combining multiple noisy annotations and in addition modeling temporal changes in annotators. Hua et al. (2013) presented a collaborative ensemble kernel machine exploring inherent correlations among the labelers through shared data among them. Hence the learned ensemble model is robust to label noises and the proposed method is able to detect irresponsible labelers online. These works build insights on how to deal with label noises and evaluate labeler quality. However, they lack explicit joint modeling of both the label noises and the labelers' quality.

For dealing with noisy annotations, there has been work on classification (Rodrigues et al. 2013), tackling the problem in the context of regression (Groot et al. 2011), sequence labeling (Rodrigues et al. 2013), and ranking (Wu et al. 2011). However, most existing works are centered only on the unobservable ground-truth true labels of the data, whose noisy observations are provided by multiple annotators. Due to the combinatorial explosion of possible outcomes of the latent variables, this choice of latent variables hinders application of such methods to structured prediction problems such as sequence labeling. With the aim of focusing on the annotators, we turn to model the label expertise of the annotators as latent variables, since it helps incorporate the diverse opinions among multiple annotators in a simple and unified way.

For modeling annotators' quality for image labeling from crowds, the most relevant works to our research is Welinder and Perona (2010b) and Welinder et al. (2010). In both works, a parameter vector of each annotator which represents the annotator's expertise, a feature (parameter) vector associated with each image which encodes each annotator's visual response to the image, and a linear classifier operating on the parameter vector associated with each image, are all inferred from existing labels provided by the annotators through a Bayesian model.

We emphasize that in these two pieces of work proposed by Welinder and Perona (2010b) and Welinder et al. (2010), no visual feature is directly extracted from the images and for each image, at least one label is needed to be able to infer the parameter vector associated with it. Hence the classifier induced from their models cannot be applied directly to an unlabeled sample, because there is no feature vector to operate on. In this sense, their models provide a principled way for active data re-labeling. In contrast, our proposed model actively induces a classifier which directly operates on visual features extracted from images, which models the labelers' quality in a principled way to facilitate active selection of annotators for providing better quality labels.

With respect to active learning with Gaussian processes, (Lawrence et al. 2003) proposed a differential entropy score, which favours points whose inclusion leads to a large reduction in predictive (posterior) variance. This approach was then extended by Kapoor et al. (2007), by introducing a heuristic confidence criterion normalized by the variance of the posterior prediction for active learning. The active learning methodology we propose further extends this work to multiple-annotator settings and introduces a new heuristic for selecting the best annotator to label an instance.

Recently, Rodrigues et al. (2014) proposed a general Gaussian process classifier in order to explicitly handle multiple annotators with different levels of expertise. However, their active learning algorithm only selects a single annotator. In contrast, our approach adopts the labels from the top K annotators among all M annotators, which integrates the potential diverse opinions from the subset of annotators with the highest expertise.

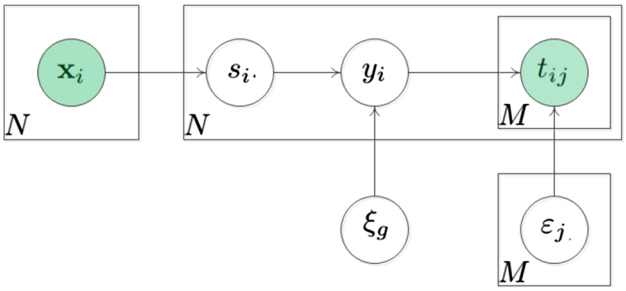

Fig. 2 The graphical model of the proposed Gaussian process classifier, with multiple noisy labels from crowds

3 Formulation, Inference, and Learning

In this section, we present a noise resilient probabilistic model, which is designed to estimate both the global label noise and the expertise of each individual labeler.

Given a set of N data points $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, where $\mathbf{x}_i \in \Re^D$, each of which may be labeled by M labelers. We let s_i denote the latent random variable with a Gaussian process prior. Intuitively, s_i can be interpreted as the soft score for the corresponding data point $\mathbf{x}_i$. The true label of $\mathbf{x}_i$ is denoted as $y_i \in \{-1, 1\}$, which is hidden. The observed label of $\mathbf{x}_i$ from labeler j is denoted as $t_{ij} \in \{-1, 1\}$, which could be noisy, meaning that t_{ij} may not be consistent with the hidden true label y_i. We denote $\mathbf{t}_i = \{t_{ij}\}_{j=1}^M$ as the set of labels from the M labelers for $\mathbf{x}_i$. For notation convenience, we denote $\mathbf{S} = \{s_1, s_2, \ldots, s_N\}$, $\mathbf{Y} = \{y_1, y_2, \ldots, y_N\}$ and $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_N\}$.

3.1 The Probabilistic Model

The proposed probabilistic model is illustrated in the graphical model in Figure 2. The conditional joint probability of this probabilistic model is defined as

$$p(\mathbf{T}, \mathbf{Y}, \mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}, \boldsymbol{\varepsilon}) = \frac{1}{Z} p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}) \prod_{i=1}^{N} \left\{ p(y_i|s_i, \xi_g) \prod_{j=1}^{M} p(t_{ij}|y_i, \varepsilon_j) \right\}, \quad (1)$$

where $\boldsymbol{\vartheta}$ is the hyperparameter of kernel matrix on $\mathbf{X}$; $\boldsymbol{\varepsilon} = \left\{\{\varepsilon_j\}_{j=1}^M, \xi_g\right\}$, where ξ_g is the global label noise measure, and ε_j is the label quality measure for labeler j; and Z is the partition function over $\mathbf{T}$, $\mathbf{Y}$ and $\mathbf{S}$.

In our model, $p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta})$ is a Gaussian process prior (Williams and Barber 1998) to ensure that similar data samples to have similar prediction scores. Formally, it is defined as

$$p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}) \sim \mathcal{N}(\mathbf{S}|\mathbf{0}, \mathbf{K}), \quad (2)$$

where $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{i,j=1}^N$ is a kernel matrix defined over the set of all N data samples. In theory, any valid kernel that measures the similarity among data samples, e.g., Chi-square kernel and linear kernel, can be applied in our model. In this paper, we use the kernel (Kim and Ghahramani 2006)

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_0 \exp\{-\theta_1 d(\mathbf{x}_i, \mathbf{x}_j)^2\} + \theta_2 + \theta_3 \delta(i = j), \quad (3)$$

where $\boldsymbol{\vartheta} = \{\theta_0, \theta_1, \theta_2, \theta_3\}$, the delta function $\delta(i = j)$ is 1 if $i = j$ and 0 otherwise. θ_0 specifies the overall scale of the variation of the latent values, θ_1 is the inverse scale for distance between $\mathbf{x}_i$ and $\mathbf{x}_j$, θ_2 is the overall bias of the latent values from zero mean, and θ_3 is the latent noise variance. We choose this form of kernel because it can be widely applicable and often performs well.

The conditional likelihood probability $p(y_i|s_i, \xi_g)$ is defined as a flip noise model Minka (2001), i.e.,

$$p(y_i|s_i, \xi_g) = \xi_g \Theta(y_i s_i) + (1 - \xi_g)\Theta(-y_i s_i), \quad (4)$$

where $\Theta(\rho) = 1$ if $\rho > 0$, and $\Theta(\rho) = 0$ otherwise. In other words, the *a posteriori* estimation of y_i takes the sign of the predicted soft label s_i with probability ξ_g. Hence, we can use ξ_g to model the global label noise level. This treatment makes the GPC resilient to label noise and outliers (Kim and Ghahramani 2008).

The conditional likelihood probability $p(t_{ij}|y_i, \varepsilon_j)$ is also modeled as a flipping noise model, i.e.,

$$p(t_{ij}|y_i, \varepsilon_j) = \varepsilon_j \Theta(y_i t_{ij}) + (1 - \varepsilon_j)\Theta(-y_i t_{ij}). \quad (5)$$

Intuitively, with probability $1-\varepsilon_j$, t_{ij} will be a flipped version of y_i. Therefore, the larger ε_j is, the higher the probability that t_{ij} will agree with the true label y_i, and vice versa. Hence, ε_j naturally represents the expertise or quality of the labels given by labeler j. We note here that unlike in Minka (2001), we parameterize this model based on label quality, which is one minus the label noise.

3.2 Inference

As a matter of fact, this two-level flip model can be conveniently collapsed by integrating y_i out. It is easy to arrive at

$$p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon}) = p(+1|s_i, \xi_g) \prod_{j=1}^{M} p(t_{ij}| + 1, \epsilon_j) + p(-1|s_i, \xi_g) \prod_{j=1}^{M} p(t_{ij}| - 1, \epsilon_j). \quad (6)$$

Therefore, we can rewrite the joint probability in Eq. 1 as

$$p(\mathbf{T}, \mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}, \boldsymbol{\varepsilon}) = \frac{1}{Z} p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}) \prod_{i=1}^{N} p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon}). \tag{7}$$

This collapsed joint probability will help us to more conveniently derive the EP inference algorithm.

For the proposed Bayesian framework, we assume that we are given a set of labeled data samples $\mathbf{X}_L = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, and the set of labels are denoted as $\mathbf{T}_L = \{t_{ij}|1 \leq i \leq N, 1 \leq j \leq M\}$. We denote $\mathbf{D}_L = \{\mathbf{X}_L, \mathbf{T}_L\}$, $\mathbf{S} = \{\mathbf{S}_L, s_u\}$, and $\mathbf{X} = \{\mathbf{X}_L, \mathbf{x}_u\}$, where $\mathbf{x}_u$ is an unlabeled data sample. To predict the label y_u of an $\mathbf{x}_u$, we need to solve the following Bayesian inference problem, i.e.,

$$\begin{aligned} p(y_u|\mathbf{x}_u, \mathbf{D}_L) &= \int_{\mathbf{S}} p(y_u|s_u) p(\mathbf{S}|\mathbf{D}_L, \mathbf{x}_u) d\mathbf{S} \\ &= \int_{s_u} p(y_u|s_u) \int_{\mathbf{S}_L} p(\mathbf{S}|\mathbf{D}_L, \mathbf{x}_u) d\mathbf{S}_L ds_u, \end{aligned} \tag{8}$$

where

$$p(\mathbf{S}|\mathbf{D}_L, \mathbf{x}_u) \propto p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}) \prod_{s_i \in \mathbf{S}_L} p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon}). \tag{9}$$

The integral in Eq. 8 is intractable as neither $p(\mathbf{S}|\mathbf{D}_L, \mathbf{x}_u)$ nor $p(y_u|s_u)$ can be integrated in closed form. We resort to Expectation Propagation (Minka 2001) to obtain an approximate integral by approximating $p(\mathbf{S}|\mathbf{D}_L, \mathbf{x}_u)$ to be a Gaussian, i.e.,

$$Q(\mathbf{S}) = p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}) \prod_{i=1}^{N} \tilde{F}_i(s_i) \sim \mathcal{N}(\mathbf{S}|\mathbf{m}, \boldsymbol{\Lambda}), \tag{10}$$

where $\mathbf{m} = [m_1, m_2, \ldots, m_N]$ and $\boldsymbol{\Lambda} = diag(v_1, v_2, \ldots, v_N)$ are the mean vector and covariance matrix of the Gaussian distribution $Q(\mathbf{S})$, and each $\tilde{F}_i(s_i)$ is a Gaussian distribution with mean $\tilde{m}_i$, variance v_i, and normalization constant A_i, i.e.,

$$\tilde{F}_i(s_i) = A_i \exp\left(-\frac{1}{2v_i}(s_i - \tilde{m}_i)^2\right), \tag{11}$$

which approximates the joint likelihood of the set of all labels obtained for $\mathbf{x}_i$, i.e.,

$$\tilde{F}_i(s_i) \approx p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon}). \tag{12}$$

Since the prior $p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta})$ is a Gaussian by definition, hence $Q(\mathbf{S})$ will also be a Gaussian distribution. Note this approximation is in contrast to previous work using EP for inference in GPC in the sense that the approximation is performed over the joint likelihood of a set of labels on a single data point.

Most previous work only considered the case of a single label for each datum. Instead of solving for each $\tilde{F}_i(s_i)$ independently, we use EP (Minka 2001) to obtain a better overall approximation. With $\Phi(x) = \int_{-\infty}^{x} \mathcal{N}(\tau; 0, 1) d\tau$, the exact steps of the EP algorithm are summarized in Algorithm 1, where Z_i is defined as

$$Z_i = (C_2 - C_1)\Phi\left(\frac{m_{-i}^{\text{old}}}{\sqrt{v_{-i}^{\text{old}}}}\right) + C_1, \tag{13}$$

$$\begin{aligned} C_1 = (1 - \xi_g) \prod_{t_{ij}=-1} (1 - \epsilon_j) \prod_{t_{ij}=1} \epsilon_j \\ + \xi_g \prod_{t_{ij}=-1} \epsilon_j \prod_{t_{ij}=1} (1 - \epsilon_j), \end{aligned} \tag{14}$$

$$\begin{aligned} C_2 = \xi_g \prod_{t_{ij}=-1} (1 - \epsilon_j) \prod_{t_{ij}=1} \epsilon_j \\ + (1 - \xi_g) \prod_{t_{ij}=-1} \epsilon_j \prod_{t_{ij}=1} (1 - \epsilon_j), \end{aligned} \tag{15}$$

and α is defined as

$$\alpha = \frac{1}{\sqrt{v_{-i}^{\text{old}}}} \cdot \frac{(C_2 - C_1)\mathcal{N}(z_i; 0, 1)}{Z_i}. \tag{16}$$

For more details, please refer to the full derivations in Appendices 1 and 2.

Algorithm 1 The Expectation Propagation Algorithm

Input: $\mathbf{X}$, $\mathbf{T}$, $\boldsymbol{\vartheta}$ and $\boldsymbol{\varepsilon}$.

$A_i = 1$, $v_i = \infty$, $\tilde{m}_i = 0$, $\mathbf{m} = \mathbf{0}$ and $\boldsymbol{\Sigma} = \mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{i,j=1}^{N}$.

repeat

  for all i such that $1 \leq i \leq N$ **do**

  ▷ *Remove* $\tilde{F}_i(s_i)$ *from* $Q(\mathbf{S})$ *to obtain an "old" posterior* $Q_{-i}(s_i) \sim \mathcal{N}(s_i|m_{-i}^{old}, v_{-i}^{old})$*:*

    $v_{-i}^{\text{old}} = (\sigma_{ii}^{-1} - v_i^{-1})^{-1}$, ▷ *where* $\sigma_{ii} = \boldsymbol{\Sigma}_{ii}$

    $m_{-i}^{\text{old}} = v_{-i}^{\text{old}}(\sigma_{ii}^{-1} m_i - v_i^{-1} \tilde{m}_i)$.

  ▷ *Minimize* $KL[Q_{-i}(s_i)p(\mathbf{t}_i|s_i)||Q_{-i}(s_i)\tilde{F}_i(s_i)]$ *:*

    $m_{-i}^{\text{new}} = m_{-i}^{\text{old}} + v_{-i}^{\text{old}}\alpha$,

    $v_i = v_{-i}^{\text{old}}(\frac{1}{m_{-i}^{\text{new}}\alpha} - 1)$,

    $\tilde{m}_i = m_{-i}^{\text{new}} + v_i\alpha$.

  ▷ *Update:*

    $A_i = Z_i\sqrt{1 + v_i{}^{-1} v_{-i}^{\text{old}}} \exp(\frac{v_{-i}^{\text{old}}\alpha}{2m_{-i}^{\text{new}}})$,

    $\boldsymbol{\Sigma} = (\mathbf{K}^{-1} + \boldsymbol{\Lambda}^{-1})^{-1}$, ▷ *where* $\boldsymbol{\Lambda}_{ii} = v_i$

    $m_i = \sum_j \sigma_{ij} \frac{\tilde{m}_j}{v_j}$. ▷ *where* $\sigma_{ij} = \boldsymbol{\Sigma}_{ij}$

  end for

until convergence

Calculate $\log Z_{EP}$.

Output: $\boldsymbol{\Lambda}$, $\tilde{\mathbf{m}}$, $\log Z_{EP}$.

The EP appoximation to the marginal likelihood can be written from the normalization of Eq. 7 as

$$Z_{EP} \approx Z = \int_{\mathbf{S}} p(\mathbf{S}|\mathbf{X}, \boldsymbol{\vartheta}) \prod_{i=1}^{N} p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon}) d\mathbf{S}. \tag{17}$$

According to Rasmussen (2006), we arrive at

$$\begin{aligned}\log(Z_{EP}) = &-\frac{1}{2}\log|\mathbf{K}+\boldsymbol{\Lambda}| - \frac{1}{2}\tilde{\mathbf{m}}^T(\mathbf{K}+\boldsymbol{\Lambda})^{-1}\tilde{\mathbf{m}} \\ &+ \sum_{i=1}^{N} \log \Phi\left(\frac{(C_2 - C_1)m_{-i}^{\text{old}}}{\sqrt{v_{-i}^{\text{old}}}}\right) \\ &+ \frac{1}{2}\sum_{i=1}^{N}\log(v_{-i}^{\text{old}} + v_i) + \sum_{i=1}^{N}\frac{(m_{-i}^{\text{old}} - m_i)^2}{2(v_{-i}^{\text{old}} + v_i)}.\end{aligned} \tag{18}$$

where C_1 and C_2 are defined in Eqs. 39 and 40 in Appendix 1. Equation 18 has a nice intuitive interpretation: the first two terms are the marginal likelihood for a regression model for $\tilde{\mathbf{m}}$, of which each component is subject to an independent Gaussian noise of variance $\boldsymbol{\Lambda}_{ii}$ (as $\boldsymbol{\Lambda}$ is diagonal). The remaining three terms come from the normalization constants for each training example. The first of these penalizes the cavity (or leave-one-out) distributions. Moreover, we can see that the marginal likelihood contains two aspects: (1) the means of the local likelihood approximations should be well predicted by a GP, and (2) the corresponding latent function, when ignoring a particular training example, should be able to predict the corresponding classification label well.

EP obtains a Gaussian approximation $Q(\mathbf{S})$ to the posterior distribution $p(\mathbf{S}|\mathbf{D}_L, \mathbf{x}_u)$. Hence the integral over $\mathbf{S}_L$ in Eq. 8 can also be approximated by a Gaussian distribution over s_u as $\mathcal{N}(s_u|m_u, v_u)$, where m_u and v_u can be obtained in closed form. Denote $\tilde{\mathbf{m}} = [\tilde{m}_1, \tilde{m}_2, \ldots, \tilde{m}_N]^T$ to be the concatenation of the mean value of each $\tilde{F}_i(s_i)$, we have

$$m_u = \mathbf{k}_u^T(\mathbf{K}+\boldsymbol{\Lambda})^{-1}\tilde{\mathbf{m}}, \tag{19}$$

$$v_u = k(\mathbf{x}_u, \mathbf{x}_u) - \mathbf{k}_u^T(\mathbf{K}+\boldsymbol{\Lambda})^{-1}\mathbf{k}_u, \tag{20}$$

where $\mathbf{k}_u = [k(\mathbf{x}_u, \mathbf{x}_1), k(\mathbf{x}_u, \mathbf{x}_2), \ldots, k(\mathbf{x}_u, \mathbf{x}_N)]^T$. We immediately have that the whole integral over all $\mathbf{S}$ in Eq. 8 can be approximated as

$$\begin{aligned}&p(y_u|\mathbf{x}_u, \mathbf{D}_L) \\ &\doteq (2\xi_g - 1)\Phi\left(\frac{y_u\mathbf{k}_u^T(\mathbf{K}+\boldsymbol{\Lambda})^{-1}\tilde{\mathbf{m}}}{\sqrt{k(\mathbf{x}_u, \mathbf{x}_u) - \mathbf{k}_u^T(\mathbf{K}+\boldsymbol{\Lambda})^{-1}\mathbf{k}_u}}\right) \\ &\quad + 1 - \xi_g,\end{aligned} \tag{21}$$

where $\Phi(\cdot)$ is the Probit function. We subsequently predict the label y_u of $\mathbf{x}_u$ based on Eq. 21.

3.3 Learning $\boldsymbol{\vartheta}$ and $\boldsymbol{\varepsilon}$ with Expectation Maximization

To online estimate the quality of both the labels and labelers, we need to online estimate the kernel parameters $\boldsymbol{\vartheta}$ and the parameters $\boldsymbol{\varepsilon} = \left\{\xi_g, \{\varepsilon_j\}_{j=1}^{M}\right\}$, which represent the overall label quality and label quality of each labeler. We further develop a generalized Expectation-Maximization algorithm for estimating it. We start by building the lower bound F of the log likelihood, i.e.,

$$\begin{aligned}&\log p(\mathbf{T}_L, \mathbf{S}_L|\mathbf{X}_L, \boldsymbol{\vartheta}, \boldsymbol{\varepsilon}) \\ &\geq \int_{\mathbf{S}_L} Q(\mathbf{S}_L)\log\frac{p(\mathbf{T}_L, \mathbf{S}_L|\mathbf{X}_L, \boldsymbol{\varepsilon})}{Q(\mathbf{S}_L)} \\ &= \int_{\mathbf{S}_L} Q(\mathbf{S}_L)\log\frac{p(\mathbf{S}|\mathbf{X}_L, \boldsymbol{\vartheta})p(\mathbf{T}_L|\mathbf{s}_L, \boldsymbol{\varepsilon})}{Q(\mathbf{S}_L)} \\ &= C + \int_{\mathbf{S}_L} q(\mathbf{S}_L)\log p(\mathbf{S}_L|\mathbf{X}_L, \boldsymbol{\vartheta})d\mathbf{S}_L \\ &\quad + \sum_{i=1}^{N}\int_{s_i} q(s_i)\log p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon})ds_i = F,\end{aligned} \tag{22}$$

where C is a constant which is independent of $\boldsymbol{\vartheta}$ and $\boldsymbol{\varepsilon}$, and $q(s_i) = \int_{\mathbf{S}\setminus s_i} Q(\mathbf{S})d\mathbf{S}$ is the marginal Gaussian posterior of $q(s_i)$. Its mean m_{s_i} and variance v_{s_i} can be obtained by Eqs. 19 and 20, respectively.

We use $F_{\boldsymbol{\vartheta}}$ and $F_{\boldsymbol{\varepsilon}}$, respectively, to denote the two integrals that make up F in Eq. 22, i.e.,

$$F_{\boldsymbol{\vartheta}} = \int_{\mathbf{S}_L} q(\mathbf{S}_L)\log p(\mathbf{S}_L|\mathbf{X}_L, \boldsymbol{\vartheta})d\mathbf{S}_L, \tag{23}$$

$$F_{\boldsymbol{\varepsilon}} = \sum_{i=1}^{N}\int_{s_i} q(s_i)\log p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon})ds_i. \tag{24}$$

Expanding $F_{\boldsymbol{\vartheta}}$ and $F_{\boldsymbol{\varepsilon}}$, we get:

$$\begin{aligned}F_{\boldsymbol{\vartheta}} &= E_q[\log p(\mathbf{S}_L|\mathbf{X}_L, \boldsymbol{\vartheta})] \\ &= E_q\left[-\frac{1}{2}\log|2\pi\mathbf{K}| - \frac{1}{2}\mathbf{S}_L{}^T\mathbf{K}^{-1}\mathbf{S}_L\right] \\ &= -\frac{1}{2}\log|2\pi\mathbf{K}| - \frac{1}{2}E_q[\mathbf{S}_L{}^T\mathbf{K}^{-1}\mathbf{S}_L] \\ &= -\frac{1}{2}\log|2\pi\mathbf{K}| - \frac{1}{2}E_q[\mathbf{S}_L]^T\mathbf{K}^{-1}E_q[\mathbf{S}_L] \\ &\quad - \frac{1}{2}tr(\mathbf{K}^{-1}Cov[\mathbf{S}_L]),\end{aligned} \tag{25}$$

$$\begin{aligned}F_{\boldsymbol{\varepsilon}} &= \sum_{i=1}^{N} E_q[\log p(\mathbf{t}_i|s_i, \boldsymbol{\varepsilon})] \\ &\approx \sum_{i=1}^{N}\log p(\mathbf{t}_i|m_{s_i}, \boldsymbol{\varepsilon})\end{aligned}$$

$$= \sum_{i=1}^{N} log\{p(+1|m_{s_i}, \xi_g) \prod_j p(t_{ij}| + 1, \epsilon_j) + p(-1|m_{s_i}, \xi_g) \prod_j p(t_{ij}| - 1, \epsilon_j)\}. \quad (26)$$

Since $\boldsymbol{\vartheta}$ is independent of $\boldsymbol{\varepsilon}$, we optimize F by optimizing $F_{\boldsymbol{\vartheta}}$ and $F_{\boldsymbol{\varepsilon}}$ iteratively until convergence.

Initially, we set $\boldsymbol{\varepsilon}_p$ as $\mathbf{1}$, and choose a valid $\boldsymbol{\vartheta}_p$. Then the following iterative steps form the EM algorithm

1. **E-Step** Given the current parameters $\boldsymbol{\vartheta}_p$ and $\boldsymbol{\varepsilon}_p$, conduct the EP inference to obtain an approximate inference of $Q(\mathbf{S}_L) \sim p(\mathbf{S}_L|\mathbf{X}_L, \mathbf{T}_L, \boldsymbol{\vartheta}_p, \boldsymbol{\varepsilon}_p)$.
2. **M-Step** Maximize the lower bound of $\log p(\mathbf{T}_L, \mathbf{S}_L|\mathbf{X}_L, \boldsymbol{\vartheta}, \boldsymbol{\varepsilon})$ in Eq. 22 over $\boldsymbol{\vartheta}$ and $\boldsymbol{\varepsilon}$ to obtain a new parameters $\boldsymbol{\vartheta}$ and $\boldsymbol{\varepsilon}$. $\boldsymbol{\vartheta}_p \leftarrow \boldsymbol{\vartheta}$, $\boldsymbol{\varepsilon}_p \leftarrow \boldsymbol{\varepsilon}$, goto the **E-Step** and iterate until convergence.

In the **M-step** , fixing $F_{\boldsymbol{\vartheta}}$, we optimize $F_{\boldsymbol{\varepsilon}}$ with respect to the parameters $\boldsymbol{\varepsilon}$. And then, fixing $F_{\boldsymbol{\varepsilon}}$, we optimize $F_{\boldsymbol{\vartheta}}$ to get the suitable kernel parameters $\boldsymbol{\vartheta}$. Note that the optimal parameters cannot be computed in a closed form, but can be solved using gradient descent. Therefore, for maximizing the lower bound, we used the gradient based projected L-BFGS-B method (Zhu et al. 1997) using the Armijo rule and a simple line search. We present the exact steps of computing the gradients of the lower bound in Appendix 3. The EM procedure described here converges to and achieves satisfactory results in our experiments.

4 Bayesian Active Learning

For pool based active learning, we assume that we are given a pool of both labeled and unlabeled data samples $\mathbf{X} = \{\mathbf{X}_L, \mathbf{X}_U\}$, and $\mathbf{T}_L$ is the label set for $\mathbf{X}_L$ from M labelers. The proposed model conveniently allows for both active selection of unlabeled data samples to be labeled, and also active selection of higher quality labelers.

For active sample selection, we use an entropy

$$H(y_u) = -\sum_{y_u \in \{1,-1\}} p(y_u|\mathbf{x}_u, \mathbf{D}_L) \log p(y_u|\mathbf{x}_u, \mathbf{D}_L)$$

of the predicted label y_u on unlabeled data $\mathbf{x}_u$, where $p(y_u|\mathbf{x}_u, \mathbf{D}_L)$ can be obtained using the EP algorithm introduced in Sect. 3.2. We select the most uncertain unlabeled example to be labeled, i.e.,

$$\mathbf{x}_u^* = \arg\max_{\mathbf{x}_u \in \mathbf{X}_U} H(y_u). \quad (27)$$

Obviously, we use entropy based active selection strategy. It would be worth mentioning that alternatives such as expected error reduction (Roy and Mccallum 2001; Zhu et al. 2003) also can be adopted for the active selection of samples.

Note ϵ_j in our model directly models the quality of labeler j. It can be regarded as the probability that labeler j would label the data correctly. Therefore, the higher ϵ_j is, the better quality the labeler has. In our active learning process, we can naturally select the top $K < M$ labelers with the top K ϵ_j to label a selected data sample, where ϵ_j is estimated by the EP-GEM algorithm presented in Sect. 3.3. The joint active selection of both labelers and data samples greatly facilitates to obtain higher quality labels.

Another active learning strategy is to only actively select the data sample to be labeled by all M labelers. Our model indeed can benefit from the multiple labels, even though there may be noise. We also compare this strategy with online majority voting in our experiments.

5 Predictive Active Set Selection Method

When the labeled pool becomes bigger and bigger, the learning process will get slower and slower because the time complexity of inference is $O(N^3)$. To apply our proposed model into practice, on one hand, we want to preserve good accuracy. On the other hand, we want to speed up the active learning system with lower computational cost. Here we resort to the Predictive Active Set Selection Method Henao and Winther (2010, 2012). An active subset (here we call it "active set") of the entire labeled data is used to learn a GPC and it is iteratively updated according to the cavity/predictive distribution of each labeled data inferred by the newly learned GPC.

Denoting the active set $\{\mathbf{X}_A, \mathbf{T}_A\}$, and the inactive set $\{\mathbf{X}_I, \mathbf{T}_I\}$, we indicate $|A|$ and $|I|$ to be the size of the active set and the inactive set, respectively. Then the marginal likelihood can be viewed as conditional independence,

$$p(\mathbf{T}|\mathbf{X}) = p(\mathbf{T}_I|\mathbf{T}_A, \mathbf{X}_A, \mathbf{X}_I) p(\mathbf{T}_A|\mathbf{X}_A), \quad (28)$$

where the second term on the right hand can be written as

$$p(\mathbf{T}_A|\mathbf{X}_A) = \int p(\mathbf{T}_A|\mathbf{S}_A) p(\mathbf{S}_A|\mathbf{X}_A, \mathbf{T}_A) d\mathbf{S}_A. \quad (29)$$

where $p(\mathbf{T}_A|\mathbf{S}_A) = \prod_{i=1}^{|A|} p(\mathbf{t}_i|s_i)$ because $\mathbf{t}_i$ is only dependent on s_i. Obviously, we can approximate $p(\mathbf{S}_A|\mathbf{X}_A, \mathbf{T}_A)$ in Eq. 29 by EP as in Sect. 3.2 on active set A. Therefore, we can replace the posterior $p(\mathbf{S}_A|\mathbf{X}_A, \mathbf{T}_A)$ by the multivariate Gaussian $Q(\mathbf{S}_A|\mathbf{X}_A, \mathbf{T}_A) = \mathcal{N}(\mathbf{S}_A|\mathbf{m}_A, \boldsymbol{\Lambda}_{AA})$, where $\mathbf{m}_A = [m_1^A, m_2^A, \ldots, m_{|A|}^A]$ and $\boldsymbol{\Lambda}_{AA} = diag(v_1^A, v_2^A, \ldots, v_{|A|}^A)$ are means and variances obtained by the EP approximation.

The conditional marginal likelihood term for inactive set can be written as

$$p(\mathbf{T}_I|\mathbf{T}_A,\mathbf{X}_A,\mathbf{X}_I)=\int p(\mathbf{T}_I|\mathbf{S}_I)p(\mathbf{S}_I|\mathbf{X}_I,\mathbf{X}_A,\mathbf{S}_A)\times p(\mathbf{S}_A|\mathbf{X}_A,\mathbf{T}_A)d\mathbf{S}_A d\mathbf{S}_I, \quad (30)$$

where $p(\mathbf{S}|\mathbf{X})=p(\mathbf{S}_I|\mathbf{X}_I,\mathbf{X}_A,\mathbf{S}_A)p(\mathbf{S}_A|\mathbf{X}_A,\mathbf{T}_A)$. Marginalizing over $\mathbf{S}_A$ in Eq. 30, it is tractable to obtain

$$Q(\mathbf{T}_I|\mathbf{T}_A,\mathbf{X}_A,\mathbf{X}_I)\approx\int p(\mathbf{T}_I|\mathbf{S}_I)\mathcal{N}\times(\mathbf{S}_I|\mathbf{m}_{I|A},\mathbf{\Lambda}_{II|A})d\mathbf{S}_I, \quad (31)$$

where $\mathbf{m}_{I|A}=[m^I_1,m^I_2,\ldots,m^I_{|I|}]$ and $\mathbf{\Lambda}_{II|A}=diag(v^I_1,v^I_2,\ldots,v^I_{|I|})$ are defined as

$$m^I_i=\mathbf{k}^T_{iA}(\mathbf{K}_{AA}+\mathbf{\Lambda}_{AA})^{-1}\tilde{\mathbf{m}}_A, \quad (32)$$

$$v^I_i=\mathbf{k}_{ii}-\mathbf{k}^T_{iA}(\mathbf{K}_{AA}+\mathbf{\Lambda}_{AA})^{-1}\mathbf{k}_{iA}. \quad (33)$$

Here $\mathbf{k}_{iA}$, $\mathbf{k}_{ii}$, $\mathbf{K}_{AA}$, $\mathbf{\Lambda}_{AA}$ and $\tilde{\mathbf{m}}_A$ are defined over the active set and the inactive set similarly as in Sect. 3.2.

Therefore, we can obtain the approximation to the marginal likelihood in Eq. 28 as

$$\tilde{Z}_{EP}=Z^I_{EP}Z^A_{EP}, \quad (34)$$

where

$$Z^I_{EP}=\int_{\mathbf{S}_I}p(\mathbf{T}_I|\mathbf{S}_I)\mathcal{N}(\mathbf{S}_I|\mathbf{m}_{I|A},\mathbf{\Lambda}_{II|A})d\mathbf{S}_I, \quad (35)$$

$$Z^A_{EP}=\int_{\mathbf{S}_A}p(\mathbf{T}_A|\mathbf{S}_A)\mathcal{N}(\mathbf{S}_A|\mathbf{m}_A,\mathbf{\Lambda}_{AA})d\mathbf{S}_A. \quad (36)$$

This approximate decomposition reduces the complexity of EP from $O(N^3)$ to $O(|A|^3+|I|^3)$. It is clear that we still make full use of the labeled information in both the active set and the inactive set to approximate the marginal likelihood, which is a big difference with those sparse Gaussian process approximation approaches that are only based on the selected subset.

As for the selection strategy for the active set, we firstly initialize the active set with N_{init} labeled data and then update the active set adaptively by two operations, *inclusion* and *deletion*. Once we get the mean m and variance v of each soft score s, we can immediately integrate over all s scores to get the probability $(2\xi_g-1)\Phi(\frac{m}{\sqrt{v}})+1-\xi_g$ following Eq. 21.

For any data point i in the inactive set, we measure the informativeness with its predictive probability, which is calculated based on m^I_i and v^I_i as

$$p^{pred}_i=(2\xi_g-1)\Phi\left(\frac{m^I_i}{\sqrt{v^I_i}}\right)+1-\xi_g.$$

We include the data point into the active set if its predictive probability is less than p_{inc}. This treatment makes sense because data points with small predictive probability are more likely to contribute to improve the classifier's performance.

In the active set, we measure the informativeness of any data point j with its cavity probability, which is calculated based on m^{old}_{-j} and v^{old}_{-j} as

$$p^{cav}_i=(2\xi_g-1)\Phi\left(\frac{m^{\text{old}}_{-j}}{\sqrt{v^{\text{old}}_{-j}}}\right)+1-\xi_g.$$

We remove the data point from the active set if its cavity probability is greater than p_{del}. This is also reasonable, because the cavity probability can be seen as a leave-one-out estimator and those points with cavity probability close to 1 do not contribute to the decision rule so that we can discard them directly.

The setting of the threshold p_{inc} and p_{del} is empirical. We can control the size of the active set by setting reasonable values for p_{inc} and p_{del}. For large-scale applications, we can set the larger p_{inc} and the smaller p_{del} to ensure the size of the active set small so that the computation cost can be significantly reduced. From this perspective, our proposed PASS-JGPC is able to remove the runtime bottleneck. According to Henao and Winther (2010, 2012), we set $p_{inc}=0.6$ and $p_{del}=0.99$ in this paper.

6 Experiments

Our experiments are conducted on four datasets with real crowd-sourced labels. First, we verify the efficacy of our proposed Bayesian model. We conduct experiments with fixed parameters $\boldsymbol{\vartheta}$. This setting saves a lot of computational costs and is feasible because all the competing approaches are running on the same kernels so that the competing condition is fair. In the subsequent set of experiments, we demonstrate the utility of learning optimal parameters $\boldsymbol{\vartheta}$. Finally, we evaluate the validity of the Predictive Active Set Selection Method in our extended Bayesian model with an additional set of experiments.

Starting with a small number of initial labeled data, we measure performance using recognition accuracy curves in both the active learning pool and the hold-out testing set as the process of learning. As it may be difficult for readers to distinguish the different curves as they might overlap closely in some cases, we provide additional tables summarizing the area under the accuracy curves (**AUAC**) with standard deviations based on 30 runs. Note that **AUAC** here is defined as the integration of recognition accuracy over the number of labeled examples. Taking each competing method as the null

hypothesis H_0 and our proposed method as the alternative hypothesis H_1 to support, we present the results of significance test with a t test by reporting the significance levels (i.e., 0.01, 0.05 or 0.10) at which our proposed method performs better significantly.

6.1 Datasets

The first dataset is composed of 3 classes of images from the ImageNet grand challenge (Deng et al. 2009), which includes 2 category of dogs, i.e., "Yorkshire terrier", "English setter" plus the "Meerkat, meerkat" category. These three classes are among the top 10 in the ImageNet grand challenge in terms of number of labeled images, with 3047, 2426 and 2341, respectively. We re-push these images back to Amazon Mechanical Turk and obtained 7 copies of labels for each image. We measure the raw label accuracy by Amazon Mechanical Turk as the percentage of the labels which are correct, i.e., agreeing with the ground-truth labels from the dataset. We also measure a labler's label accuracy as the percentage of his/her labels which are correct. In the ImageNet dataset, the raw label accuracies of these three classes are 97.87, 96.83 and 99.27 %, respectively. The label accuracies of the 7 labelers are 95.52, 95.77, 95.43, 95.81, 95.90, 95.70 and 95.94 %, respectively. The features we used to represent each image is the local coordinate coding (LCC) (Lin et al. 2011) on densely extracted HoG features with 4096 codewords. The LCC features are pooled in 10 spatial cells, resulting a 40960 dimensional feature.

The second dataset we experiment on is a subset of the CMU multi-modal action category dataset (CMU-MMAC) (Spriggs et al. 2009), where crowd-sourced labels have been obtained by Zhao et al. (2011). In total there are 2682 labeled video clips, each has 7 copies of labels from the Amazon Mechanical Turk. The action labels include: 1. close; 2. crack; 3. open; 4. pour; 5. put; 6. read; 7. spray; 8. stir; 9. switch on; 10. take; 11. twist off; 12. twist on; 13. walk; and 14. others. The corresponding number of clips for each action is: 7, 54, 711, 112, 453, 116, 43, 94, 654, 103, 290, 11, 12, and 22, respectively. The raw label accuracies of these 14 actions are 8.16, 62.96, 24.95, 65.94, 48.47, 18.19, 53.48, 45.89, 75.56, 27.20, 40.05, 0.00, 11.90 and 12.99 %, respectively. The label accuracies of 7 labelers are 30.16, 38.35, 35.01, 37.65, 35.46, 34.13 and 37.11 %, respectively. Since it is impractical to average the results over all 13 categories because the results are produced from different number of clips, and it might occupy too much space if we report the results separately for each category. We choose to work on the classification problem of action 9 only, which has sufficient number of labeled clips and its ground-truth label accuracy is 75.56 %. Since the CMU-MMAC dataset incorporates multiple modality, instead of using visual features extracted from video frames, we use the feature extracted from the IMU modality provided by Zhao et al. (2011). The feature dimension is 180. We refer the reader to Zhao et al. (2011) for more details on how the features are extracted.

The third dataset for our experiment is a gender face dataset, where we try to learn a gender classifier from facial features. We collected 5 copies of gender labels for 9441 face images. The raw label accuracy is 95.36 %, and the label accuracies of the 5 labelers are 95.51, 94.90, 95.23, 95.59 and 95.55 %, respectively. The face images are all 64×64. We extract a 5408 dimensional features from each face image. This feature extractor is a convolutional neural network trained for gender recognition with a separate small set of labeled gender face images. The feature is the output of the last layer of the convolutional neural network (Tivive and Bouzerdoum 2006). We will share the features of this data upon publication of our paper.

The last dataset we use is Waterbird dataset (Welinder et al. 2010), which includes 240 images in total with 200 images of water birds and the rest 40 images without any birds at all. There are 4 bird species: Mallard, American Black Duck, Canada Goose and Red-necked Grebe, each of these species has 50 photographs. Each image is labeled by 40 labelers among 53 annotators. The raw label accuracies are 67.56, 48.08, 63.19, 18.53 and 56.49 % (refers to the category without any birds at all), respectively. The mean and variance of the label accuracies of the 53 annotators are are 50.77 and 6.98 %, respectively. We extract a 4096 dimensional features from each image using the code online for exacting the fine-grained feature (Yao et al. 2011).

6.2 Experiments on the ImageNet Dataset

6.2.1 Effectiveness of Labeler Selection

The simulation experiment we conducted is on the ImageNet dataset. To demonstrate the effectiveness of our model to avoid low quality labelers, we use the class "Meerkat, meerkat" as an example. We take 1000 images as positive samples from the class "Meerkat, meerkat" and 1000 images from the other two dog classes to serve as the hold-out testing set. The rest of the images in "Meerkat, meerkat" and an equal number of images from the other two classes are put together to form the active learning pool. We simulate the case that there are 2, 3, 4 irresponsible labelers, who would randomly assign a label to the sample, assuming there is 50 % chance that the label from them will be erroneous. For good labelers, we used the noisy labels obtained from Amazon Mechanical Turk. Therefore, we run our proposed active learning algorithm for both active selection of data samples and labelers. 3 labelers with higher estimated label quality ϵ_j are selected to provide the labels for the actively selected samples.

We name our algorithm as JGPC-ASAL, which stands for joint learning GPC with active selection of both sam-

Table 1 The prefix, infix and suffix, and their descriptions used in the short names of competing algorithms in this paper

Notation	Description
JGPC-/JGPC	Gaussian process classifier with joint treatment of multiple labels for each labeled sample.
Para-JGPC-	JGPC with a parameterized kernel.
PASS-JGPC-	JGPC with predictive active selection method.
GPC-MV	Standard Gaussian process classifier with majority voting the multiple labels for each labeled sample.
GPC-GRD	Standard Gaussian process classifier with the ground-truth label for each labeled sample.
ML-Bernoulli-	Yan et al.'s learning algorithm Yan et al. (2011, 2012) with Bernoulli distributions.
ML-Gaussian-	Yan et al.'s learning algorithm Yan et al. (2011, 2012) with Gaussian distributions.
-F	With the flip noise model.
-K	Without the flip noise model proposed by Kapoor et al. (2007).
-ASAL	Active selection of both samples and labelers.
-ASRL	Active selection of samples with random selection of labelers.
-RSAL	Random selection of samples with active selection of labelers.
-RSRL	Random selection of both samples and labelers.
-AS/AS-	Active selection of samples with the labels from all the labelers.
-RS/RS-	Random selection of samples with the labels from all the labelers.

ples and labelers (we call it joint learning in the sense that the multiple labels of a single example are jointly considered). We compare with a combination of other learning strategies with our model, such as active selection of samples but random selection of labelers, random selection of samples and active selection of labelers, and random selection of both samples and labelers. We call these three algorithms JGPC-ASRL, JGPC-RSAL and JGPC-RSRL, respectively. For all these online learning algorithms based on JGPC, we select 3 labelers to provide the label using the corresponding criterion for labeler selection.

One algorithm we compare against is the active learning GP classifier with the global flip noise observation model similar to the model in Kim and Ghahramani (2008). For this method, at each round, we use the prediction entropy to select the next sample to be labeled and a majority voting is performed to obtain a single label from all 7 copies of labels. We name it as majority vote active learning GPC with flip noise model, or in short GPC-MVAS-F. The corresponding algorithm performing random sample selection using majority voted label, is named as GPC-MVRS-F. We also compare against the algorithm based on the active learning GP classifier proposed by Kapoor et al. (2007), where a Gaussian observation model is adopted and a confidence criterion normalized by the variance of the posterior prediction is adopted for active learning. Again, majority voting is performed at each active learning step to obtain a single label from all 7 copies of noisy labels. We name this algorithm as GPC-MVAS-K, and its random sample selection version is named as GPC-MVRS-K. Since there are no mechanism of labeler selection, we simply gather majority voted labels from all 7 labelers. To facilitate the readers to quickly reference to the meaning of a specific short names, we summarize the descriptions of the prefix, infix and suffix used in the paper as in Table 1.

Figures 3, 4 and Table 2 present the recognition accuracy evolving with increasing number of labeled examples for all the competing methods with 2 (Fig. 3a, b), 3 (Fig. 4a) and 4 (Fig. 4b) irresponsible labelers. We have the following observations: (1) Overall, the recognition accuracy of the proposed JGPC-ASAL is constantly ranked on the top, in both the active learning pool and the hold-out testing set, which is not affected by the number of irresponsible labelers due to the active selection of higher quality labelers; (2) The JGPC-ASRL algorithm achieves roughly the same accuracy as the JGPC-ASAL when there are only 2 irresponsible labelers, but degrades gradually with the increasing number of irresponsible labelers. This phenomenon suggests that active selection of samples to be labeled is more important than the active selection of labelers, when all the labels are relatively low in noise. This confirms our intuition, since the quality of the labelers is all high; (3) The GPC-MVAS-F algorithm outperforms GPC-MVAS-K, which revalidates the advantage of the flip noise model based observation model over a simple Gaussian observation model in a Gaussian process classifier; (4) In all cases for all algorithms, the active sample selection strategy always outperforms its random sample selection counterpart, which suggests that the proposed active learning criterion is robust against label noises; (5) Taking the significance tests based on the results in Table 2, we observe that the JGPC-ASAL algorithm is significantly better than

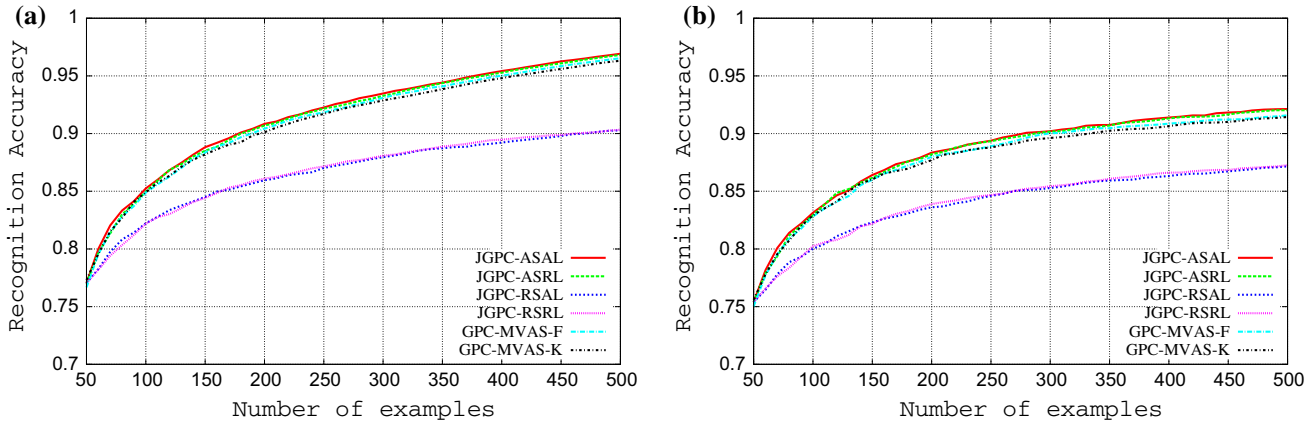

Fig. 3 Recognition performance on the "Meerkat, meerkat" class with 2 irresponsible labelers. **a** Active learning pool - 2 irresponsible labelers. **b** Hold-out testing set - 2 irresponsible labelers

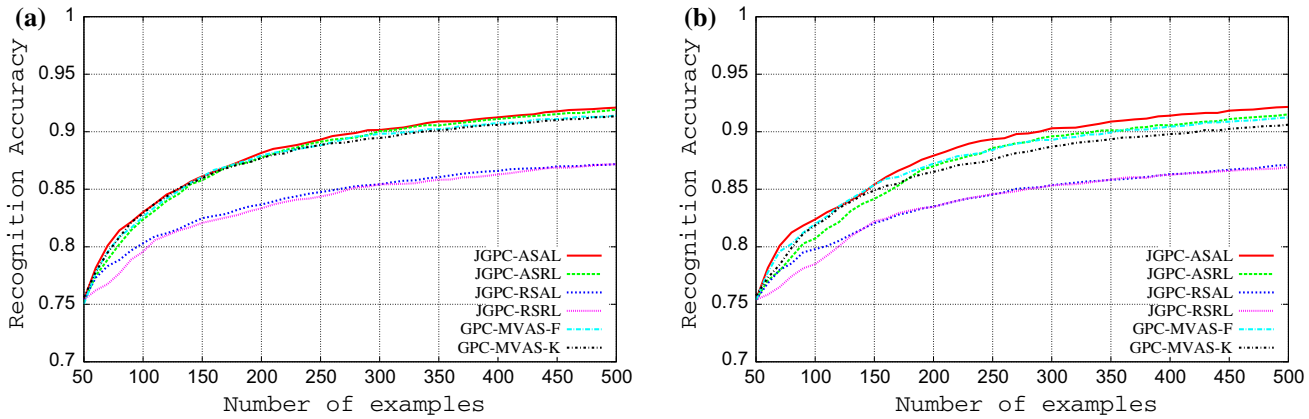

Fig. 4 Recognition performance on the "Meerkat, meerkat" class with 3 and 4 irresponsible labelers. **a** Hold-out testing set - 3 irresponsible labelers. **b** Hold-out testing set - 4 irresponsible labelers

Table 2 The recognition performance measured by AUAC on the "Meerkat, meerkat" class with 2 irresponsible labelers

AUAC*	Active-2	Test-2	Test-3	Test-4
JGPC-ASAL	**412.420** ± 0.028	**398.262** ± 0.035	**397.817** ± 0.030	**397.329** ± 0.033
JGPC-ASRL	412.116 ± 0.031	397.741 ± 0.037	396.386 ± 0.035	392.970 ± 0.055
JGPC-RSAL	389.806 ± 0.076	378.268 ± 0.062	379.154 ± 0.056	377.931 ± 0.075
JGPC-RSRL	390.089 ± 0.075	378.750 ± 0.061	377.641 ± 0.077	377.052 ± 0.057
GPC-MVAS-F	410.658 ± 0.029	396.271 ± 0.028	395.914 ± 0.032	394.020 ± 0.045
GPC-MVAS-K	409.941 ± 0.028	395.710 ± 0.033	395.571 ± 0.034	391.388 ± 0.048

Active-2 stands for the results on the active learning pools. Test-2, Test-3 and Test-4 refer to the results on the hold-out testing set with 2, 3 and 4 irresponsible labelers, respectively
The significance of bold is to emphasize the largest value of the measurement under the same comparison condition
*AUAC here is the area under the accuracy curve. It is defined as the integration of recognition accuracy over the number of labeled examples

any of the competing algorithms at the 0.01 significance level.

Figure 5 visualizes the top three labelers selected at each active learning step when running the proposed JGPC-ASAL algorithm on the "Meerkat, meerkat" class with 4 irresponsible labelers (labeler 4, 5, 6, and 7 are irresponsible labelers). The red, blue, and green color circles represent the top three labelers selected based on the estimated labeler quality measure at each active learning step. As we can clearly observe, at the beginning, the users selected are more or less uniform across the 7 labelers. Then with the progression of the active learning process, the three good labelers (labeler 1, 2 and 3) got constantly selected. This demonstrates the efficacy of the proposed model for online modeling of the labelers' quality.

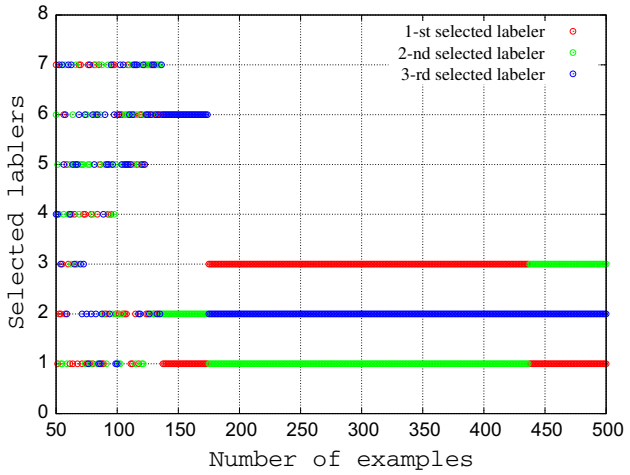

Fig. 5 Selected labelers on "Meerkat, meerkat" class with 4 irresponsible labelers, i.e., labeler 4, labeler 5, labeler 6 and labeler 7

6.2.2 Labelers with Different Expertise

To better understand the behavior of our algorithm, we run two sets of experiments with simulated label noises from the ground-truth labels on the 3 categories of ImageNet dataset. We use the "Meerkat, meerkat" as the positive class and the other two dog classes as the negative class. In the first experiment, we simulated the case that labeler 1, labeler 2, labeler 3, labeler 4, labeler 5, labeler 6 and labeler 7 produce 10, 15, 20, 25, 30, 35 and 40 % erroneous labels, respectively. In the second experiment, we increase the label noise level to have each labeler to produce 15, 20, 25, 30, 35, 40, and 45 % erroneous labels, respectively. We also impose a naive majority voting consensus based labeler selection scheme to the GPC-MVAS-F and GPC-MVRS-F algorithm.

For each labeler, we record online his/her rating of consistent labels with the corresponding majority voted labels. Intuitively, the higher this rate, the better the labeler's quality. We call the GPC-MVAS-F and GPC-MVRS-F algorithm with this simple active labeler selection scheme as GPC-MVASAL-F and GPC-MVRSAL-F, respectively. To validate its efficacy, we also compare against its corresponding random labeler selection version, namely GPC-MVASRL-F and GPC-MVRSRL-F. Again, at each step of the online learning process, we select 3 labelers to provide the labels.

Figure 6 and Table 3 present the results of the two experiments on ImageNet dataset. Our observations are: (1) The proposed JGPC-ASAL algorithm achieves slightly better accuracy than the GPC-MVASAL-F algorithm in the first experiment, and much higher accuracy in the second experiment, which indicates that our proposed labeler selection criterion is more robust to label noises than the naive majority voting consensus based labeler selection criterion; (2) The naive majority voting consensus based labeler selection criterion is also effective, as it achieved better accuracy than its random labeler selection counterpart and; (3) Our proposed JGPC-ASAL algorithm outperforms any of the baselines significantly at the 0.01 significance level.

6.2.3 Experiments with Real Crowd-Sourced Labels

We further run experiments with all real crowd-sourced labels on the 3 categories of images. For each category, we randomly sample 1000 examples from it and another 1000 examples from the other two categories to serve as the hold-out testing set. The rest of the examples in the target category and an equal number of examples from the other two classes are put in the active learning pool. Since the 7 copies of labels we

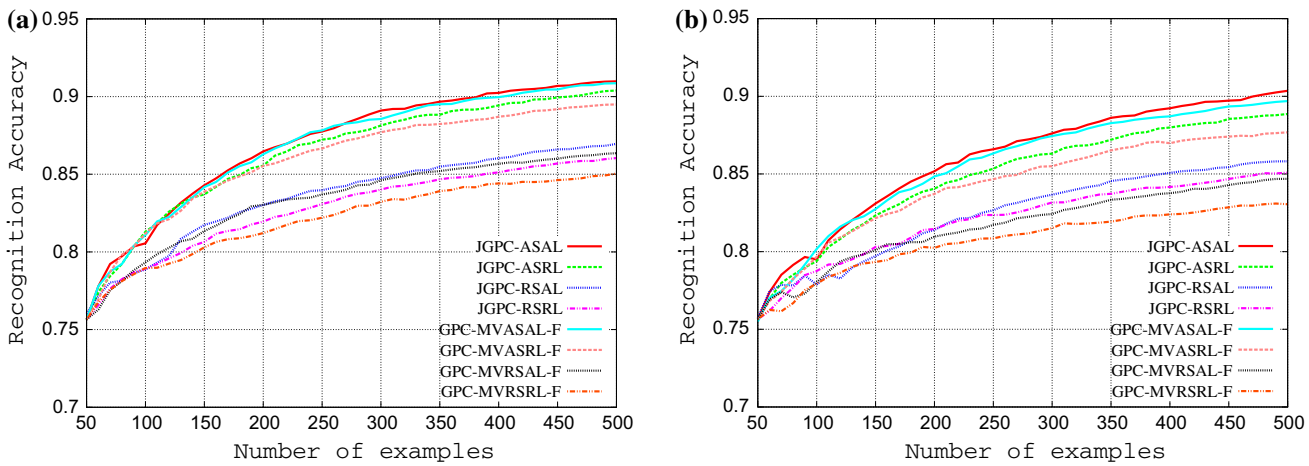

Fig. 6 Experiments with labelers with different level of label noises on the ImageNet dataset: **a** 7 labelers are simulated with 10, 15, 20, 25, 30, 35 and 40 %, respectively; **b** 7 labelers are simulated with 15, 20, 25, 30, 35, 40, and 45 %, respectively. **a** 10–40 % for each labeler. **b** 15–45 % for each labeler

Table 3 The recognition performance measured by AUAC on the "Meerkat, meerkat" class with 7 labelers with different expertise

AUAC	Test-(10–40 %)	Test-(15–45 %)
JGPC-ASAL	**391.916** ± 0.458	**387.183** ± 1.188
JGPC-ASRL	387.216 ± 0.519	379.594 ± 1.272
JGPC-RSAL	374.750 ± 0.502	370.506 ± 1.377
JGPC-RSRL	372.750 ± 0.545	369.506 ± 1.474
GPC-MVASAL-F	390.803 ± 0.432	385.932 ± 1.058
GPC-MVASRL-F	376.094 ± 0.536	372.406 ± 1.313
GPC-MVRSAL-F	373.109 ± 0.679	367.713 ± 1.650
GPC-MVRSRL-F	369.717 ± 0.720	363.370 ± 1.618

Test-(10–40 %) stands for the results on the hold-out testing set where 7 labelers are simulated with 10, 15, 20, 25, 30, 35 and 40 %, respectively; Test-(15–45 %) refers to the results on the hold-out testing set where 7 labelers are simulated with 15, 20, 25, 30, 35, 40 and 45 %, respectively
The significance of bold is to emphasize the largest value of the measurement under the same comparison condition

Table 4 The recognition performance measured by AUAC on the ImageNet dataset with real crowd-sourced labels

AUAC	Active	Test
JGPC-ASAL	**390.927** ± 0.639	**378.221** ± 0.746
JGPC-ASRL	390.343 ± 1.987	378.100 ± 2.491
JGPC-RSAL	372.565 ± 2.035	364.226 ± 2.492
JGPC-RSRL	372.764 ± 2.909	364.242 ± 3.292
GPC-MVAS-F	390.184 ± 2.100	377.914 ± 2.542
GPC-MVAS-K	388.361 ± 2.035	376.533 ± 2.358
ML-Bernoulli-AS	356.100 ± 9.403	344.882 ± 9.659
ML-Gaussian-AS	358.674 ± 7.680	346.670 ± 8.322

Active stands for the results on the active learning pool, while Test refers to the results on the hold-out testing set
The significance of bold is to emphasize the largest value of the measurement under the same comparison condition

collected from the Amazon Mechanical Turk do not really entail labels from irresponsible labelers, we found that active selection of higher quality labelers does not really improve recognition accuracy much.

We argue that our joint treatment of multiple labels in GPC in general is superior to the majority voting strategy (GPC-MVAS-F and GPC-MVAS-K), as manifested by the results shown in Fig. 7 and Table 4. We also compare against two versions of the active learning algorithm proposed by Yan et al. (2011, 2012), namely ML-Bernoulli-AS and ML-Gaussian-AS, respectively.

Figure 7 also presents the results of these competing GPC algorithms with other sample and labeler selection strategy, namely JGPC-ASRL, JGPC-RSAL, and JGPC-RSRL, respectively. The curves plotted in Fig. 7 are averaged over the three classes over multiple runs with different initial labels to counter the statistic differences. As we can clearly observe, the proposed JGPC-ASAL performs best especially at the early stage. JGPC-ASRL is on par with GPC-MVAS-F, and outperforms the GPC-MVAS-K algorithm in this dataset. Again, both JGPC-ASAL and JGPC-ASRL perform better than JGPC-RSAL and JGPC-RSRL, which is consistent with the observation in the synthetic experiments. The ML-Bernoulli-AS and ML-Gaussian-AS performed poorly on this dataset with real crowd-sourced labels, which is not surprising as it induces a linear classifier.

We take significance tests based on the results in Table 4, and find that our proposed JGPC-ASAL algorithm outperforms JGPC-RSAL, JGPC-RSRL, GPC-MVAS-F, GPC-MVAS-K, ML-Bernoulli-AS and ML-Gaussian-AS significantly at the 0.01 significance level on both the active learning pool and the hold-out testing set. When compared to JGPC-ASRL, JGPC-ASAL shows better results significantly at the 0.01 significance level on the active learning pool only and it is even not superior to JGPC-ASRL significantly at the 0.10 significance level on the hold-out testing set.

For the readers' convenience to read and see the differences, we remove the curves of JGPC-RSAL and JGPC-RSRL, and focus on presenting the experimental results at the hold-out testing set in the following subsections.

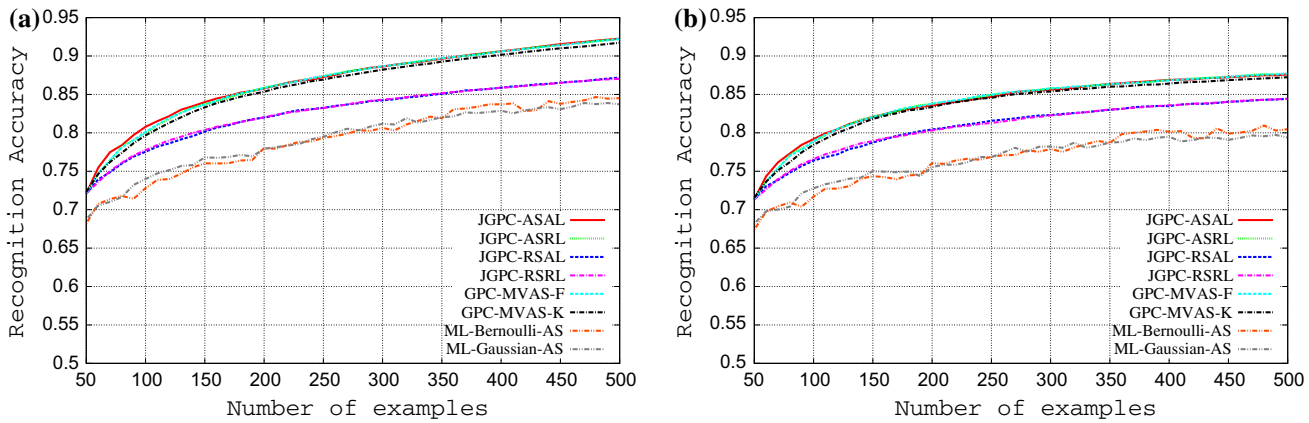

Fig. 7 Recognition performance evaluation on the ImageNet dataset. The results are average over 3 categories. **a** Active learning pool. **b** Hold-out testing set

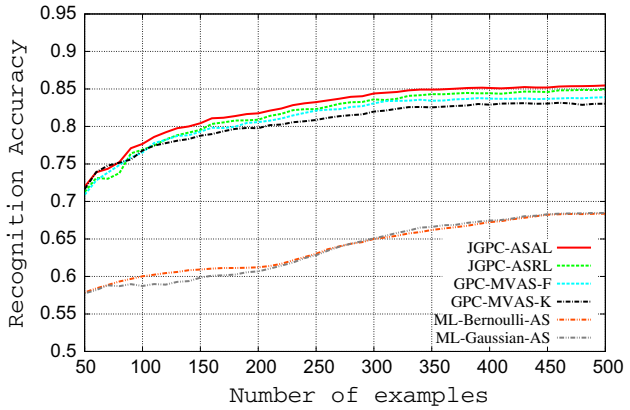

Fig. 8 Recognition performance evaluation on the hold-out testing set of the CMU-MMAC dataset

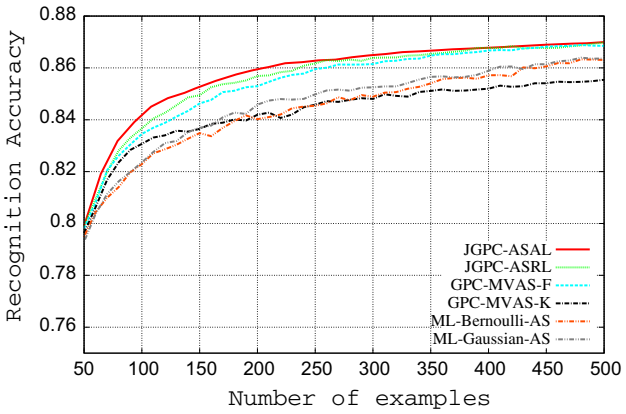

Fig. 9 Recognition performance evaluation on the hold-out testing set of the Face gender dataset

Table 5 The recognition performance measured by AUAC and its significance test on the CMU-MMAC dataset with real crowd-sourced labels

AUAC	Test
JGPC-ASAL	**370.544** ± 1.303
JGPC-ASRL	368.036 ± 1.356
GPC-MVAS-F	365.406 ± 1.291
GPC-MVAS-K	362.340 ± 1.579
ML-Bernoulli-AS	287.542 ± 6.626
ML-Gaussian-AS	286.523 ± 6.322

Test refers to the results on the hold-out testing set
The significance of bold is to emphasize the largest value of the measurement under the same comparison condition

Table 6 The recognition performance measured by AUAC on the Face gender dataset with real crowd-sourced labels

AUAC	Test
JGPC-ASAL	**386.834** ± 0.608
JGPC-ASRL	385.506 ± 0.728
GPC-MVAS-F	384.557 ± 0.648
GPC-MVAS-K	378.758 ± 0.853
ML-Bernoulli-AS	246.216 ± 8.234
ML-Gaussian-AS	246.014 ± 7.190

Test refers to the results on the hold-out testing set
The significance of bold is to emphasize the largest value of the measurement under the same comparison condition

6.3 Experiments on the CMU-MMAC Dataset

We evaluate the proposed framework on the CMU-MMAC dataset. We take 250 clips of action 9 and 250 clips of the other actions to form the hold-out testing set. The remaining rest 404 clips of action 9 and the same number of clips from the other actions are used as the active learning pool. Figure 8 and Table 5 show the classification accuracy of our JGPC-ASAL and the baselines. As apparent in the figure, our proposed JGPC-ASAL algorithm consistently outperforms the GPC-MVAS-F, GPC-MVAS-K, ML-Bernoulli-AS and ML-Gaussian-AS. It is also obvious that our JGPC-ASAL performing active selection of labelers always achieved better performance when compared with its random counterpart JGPC-ASRL. The significance tests show that our proposed JGPC-ASAL algorithm outperforms all the competing methods significantly at the 0.01 significance level. This further validates the efficacy of our model.

6.4 Experiments on the Gender Face Dataset

We also carry out experiments on the gender face dataset. We hold out 2000 face images, for which all 5 copies of labels are in consensus, for testing purpose. The rest of the face images with different percentage of label inconsistency are used in the active learning pool. Figure 9 and Table 6 compare the performance of the proposed JGPC-ASAL algorithm and the competing algorithms. It can be seen that the proposed JGPC-ASAL algorithm again showed superior recognition accuracy when compared with the JGPC-ASRL, GPC-MVAS-F, and GPC-MVAS-K algorithms. The significance tests also show that our JGPC-ASAL outperforms all the baselines significantly at the 0.01 significance level.

6.5 Experiments on the Waterbird Dataset

In order to show the efficacy of our proposed algorithm in the scenario there are more difficult classes where the agreement between labelers is lower, we conduct fine-grained classification on the Waterbird dataset. For each bird specie, we randomly sample 30 images from it and another 30 from the other categories to form the active learning pool. The rest of the examples of the same specie and an equal number of examples from the other classes are put in the hold-out testing pool. Each images is labeled by 40 annotators on the Amazon Mechanical Turk. Here we also assume all the labelers

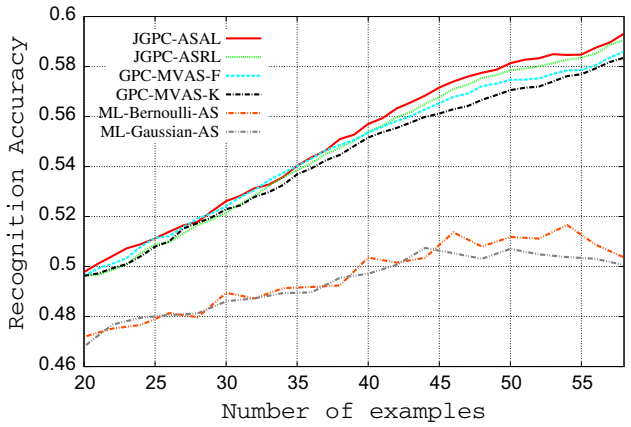

Fig. 10 Recognition performance evaluation on the hold-out testing set of the Waterbird dataset. The results are average over 4 species

Table 7 The recognition performance measured by AUAC on the Waterbird dataset with real crowd-sourced labels

AUAC	Test
JGPC-ASAL	**21.501** $\pm$ 0.701
JGPC-ASRL	21.389 $\pm$ 0.709
GPC-MVAS-F	21.365 $\pm$ 0.729
GPC-MVAS-K	21.20 $\pm$ 0.763
ML-Bernoulli-AS	9.432 $\pm$ 0.984
ML-Gaussian-AS	9.383 $\pm$ 0.813

Test refers to the results on the hold-out testing set
The significance of bold is to emphasize the largest value of the measurement under the same comparison condition

will all label the selected examples. The results are shown in Fig. 10 and Table 7. As we can see, the gaps among the curves are rather small, which indicates the classification task is indeed difficult. In spite of this, our JGPC-ASAL algorithm shows better recognition accuracy when compared with the JGPC-ASRL, GPC-MVAS-F and GPC-MVAS-K.

These observations are consistent with the results of the significance tests that our proposed JGPC-ASAL algorithm: (1) dose not perform better significantly even at the 0.10 significance level; (2) is better than GPC-MVAS-F significantly at the 0.05 significant level; and (3) outperforms all the rest of baselines significantly at the 0.01 significance level. It is also worth mentioning that the size of this dataset is so small that any examples selected can help improve the classification performance, which can explain to the phenomenon why the recognition accuracy of all the Gaussian process classifiers increase rapidly.

6.6 Comparison with Joint Gaussian Process Model

In order to show the efficacy of our joint Gaussian process model and name it as JGPC, we compare it with the majority vote GPC with flip noise model, the majority voting GPC without flip noise model, and the GPC with ground-truth labels on the 4 above mentioned datasets with all the known labeled data. To be brief, we call them GPC-MV-F, GPC-MV-K and GPC-GRD, respectively. The results are summarized in Table 8. It is very clear that our JGPC is slightly better than GPC-MV-F and GPC-MV-K, and comparable to GPC-GRD. This demonstrates the power of our proposed joint Gaussian process model.

Table 8 The comparison of recognition performance among 4 models with all the labeled data on the 4 datasets.(unit: %)

Accuracy(%)	JGPC	GPC-MV-F	GPC-MV-K	GPC-GRD
ImageNet	**90.03**	**90.03**	89.62	**90.03**
CMU-MMAC	85.60	85.20	84.40	**86.00**
Genderface	**86.85**	86.80	86.25	**86.85**
Waterbird	61.25	59.38	56.88	**62.50**

The significance of bold is to emphasize the largest value of the measurement under the same comparison condition

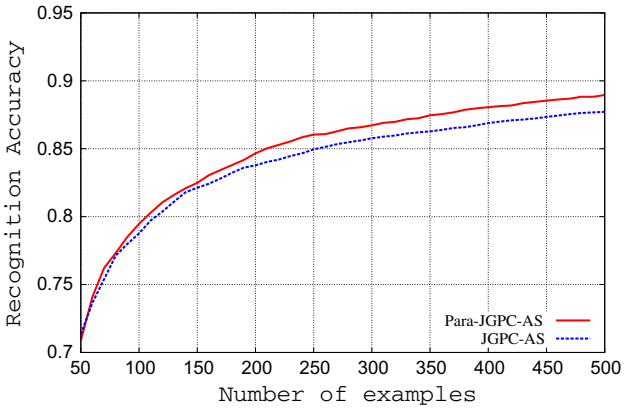

Fig. 11 Efficacy of learning parameters on the hold-out testing set of the ImageNet dataset. The results are average over 3 categories

In the following two subsections, we are going to extend our JGPC model by exploiting a parameterized kernel in the learning process and leveraging a predictive active set selection method. We focus on the active sampling by disabling the selection of labelers in the process.

6.7 Efficacy of Learning Parameters ϑ

Note that in the above experiments, the kernel parameters ϑ are fixed. To evaluate the efficacy of our proposed model with the parameterized kernel, we perform a set of experiments on the ImageNet, CMU-MMAC and Face gender datasets with real crowd-sourced labels. We name our parameterized algorithm performing active learning as Para-JGPC-AS, which enable only the selection of samples without the selection of labelers. We compare their performances with its counterpart JGPC-AS. The results are reported in Figs. 11, 12 and

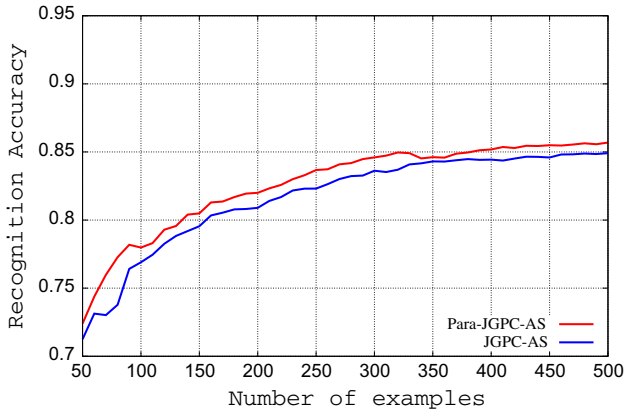

Fig. 12 Efficacy of learning parameters on the hold-out testing set of the CMU-MMAC dataset

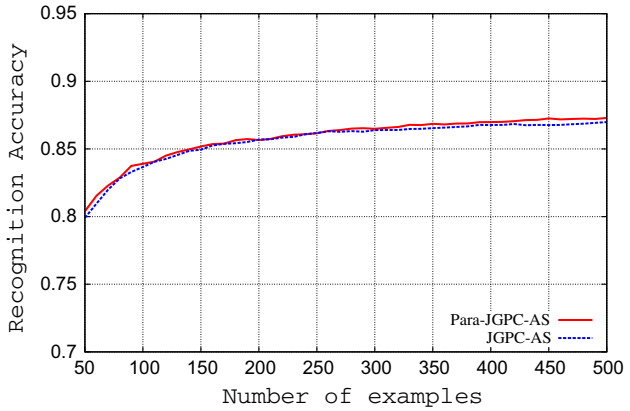

Fig. 13 Efficacy of learning parameters on the hold-out testing set of the Face gender dataset

13, which indicate that Para-JGPC-AS outperforms JGPC-AS. Not surprisingly, this is due to the fact that the learned optimal parameters ϑ is beneficial to improve recognition accuracy for our proposed Bayesian learning framework.

6.8 Efficacy of Predictive Active Set Selection Method

To obtain a quantitative evaluation of the effectiveness of our extended Bayesian model with the Predictive Active Set Selection Method, which we call PASS-JGPC-AS in short. We run another set of experiments on the first three datasets with real crowd-sourced labels except the Waterbird dataset. In this set of experiments, we set $N_{init} = 100$, $p_{inc} = 0.60$, $p_{del} = 0.99$, and initialize the active set as 50 positive data points and 50 negative data points. Considering that it is not necessary to apply PASS-JGPC for sparsity in practice when the number of the labeled examples is too small, we start evaluating PASS-JGPC-AS with 130 labeled data points. The results are provided in Figs. 14, 15 and 16. We find that the recognition accuracy of PASS-JGPC-AS is very close to that of JGPC-AS on all these three datasets.

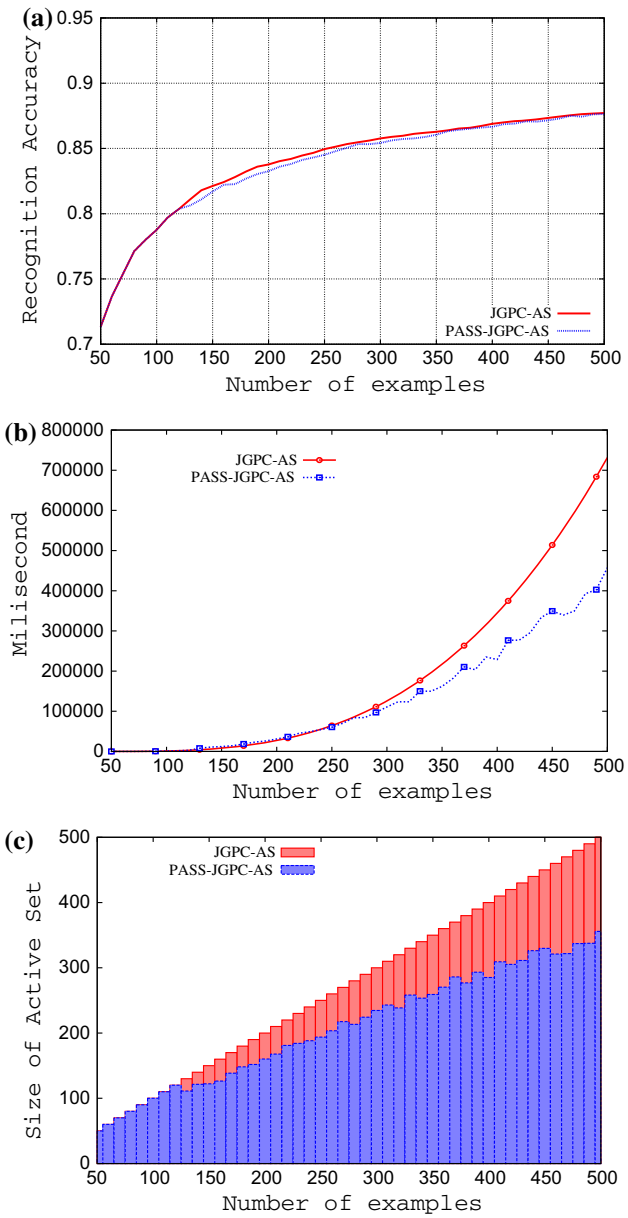

Fig. 14 Efficacy of PASS-JGPC-AS on the ImageNet dataset. The results are average over 3 categories. **a** Hold-out testing set. **b** Speed. **c** Active set

As shown, PASS-JGPC-AS can roughly keep up with the high accuracy of JGPC-AS. From Figs. 14b, 15b, and 16b, we can observe that: (1) when the number of labels is less than 250, PASS-JGPC-AS is a little slower than JGPC-AS. This can be explained by the fact that it usually takes a few iterations for PASS-JGPC-AS to reach the final adaptive active set. (2) As the number of labels increases, PASS-JGPC-AS is faster than JGPC-AS. The corresponding number of data points selected into the active set is plotted in Figs. 14c, 15c, and 16c. This explains why PASS-JGPC-AS obtains the higher speed in the learning process. It is clearly

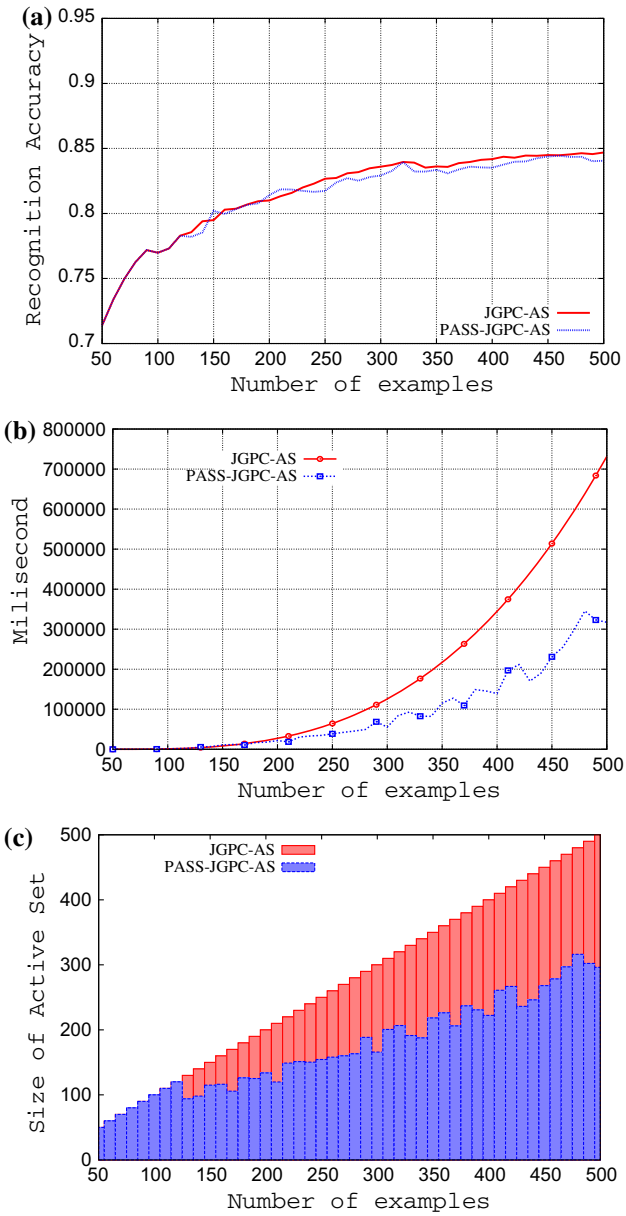

Fig. 15 Efficacy of PASS-JGPC-AS on the CMU-MMAC dataset. **a** Hold-out testing set. **b** Speed. **c** Active set

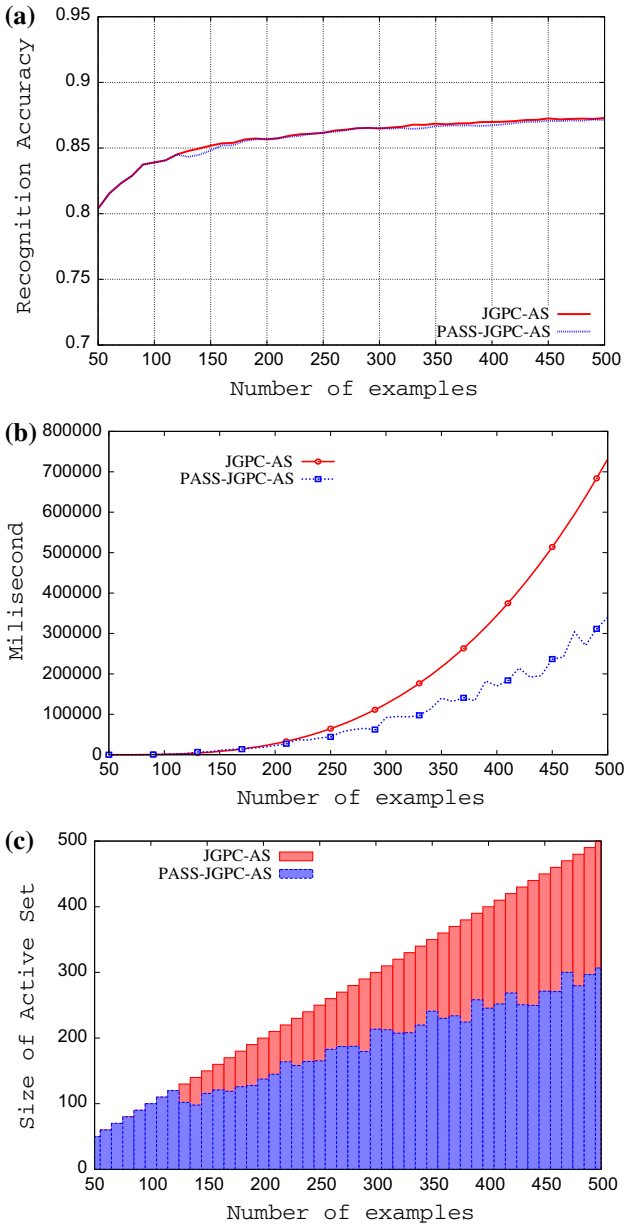

Fig. 16 Efficacy of PASS-JGPC-AS on the Face gender dataset. **a** Hold-out testing set. **b** Speed. **c** Active set

demonstrated that PASS-JGPC-AS shows its superiority in efficiency over JGPC-AS.

6.9 Evaluation with Different Number of Initial Labeled Examples

In order to evaluate the recognition performance with different number of initial labeled examples, we run a group of experiments with our proposed JGPC-ASAL, JGPC-ASRL, and the two baselines, *i.e.*, GPC-MVAS-F and GPC-MVAS-K and start with 2, 4, 8, 16 and 32 labeled examples on the three datasets used as in the previous subsection. On one hand, we compare the recognition performances of our proposed JGPC-ASAL with different number of initial labeled examples, as shown in the Figs. 17, 18 and 19a. As we can observe, with less initial provided labels, the recognition accuracy is lower at the early stages, and the gap of accuracies among the curves is becoming smaller and smaller with the progress of the active learning process. This strongly demonstrates the robustness of our proposed JGPC-ASAL, which obtains relatively stable recognition accuracy and is independent of the number of initial provided labels at the latter stages.

On the other hand, we compare our proposed JGPC-ASAL and JGPC-ASRL, GCP-MVAS-F and GPC-MVAS-F in each

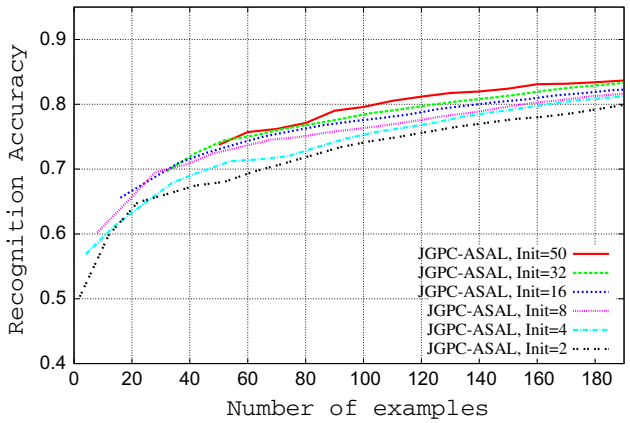

Fig. 17 Recognition performance evaluation with different number of initial labeled examples on the hold-out testing set of the CMU-MMAC dataset

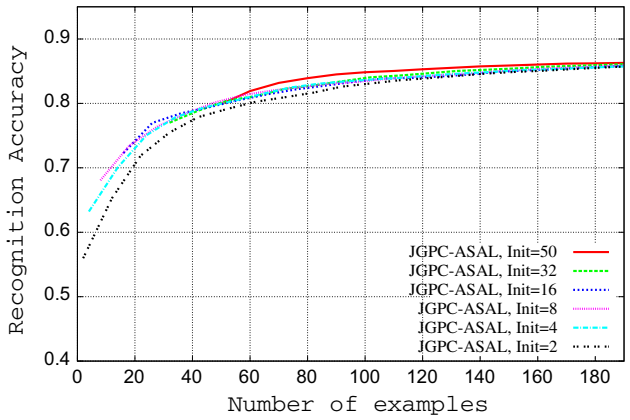

Fig. 18 Recognition performance evaluation with different number of initial labeled examples on the hold-out testing set of the Face gender dataset

case of using the same number of initial labeled examples and report the results on the ImageNet dataset in Fig. 19b–f. Obviously, our proposed JGPC-ASAL always performs best. The outperformance of JGPC-ASAL is also demonstrated on both the CMU-MMAC dataset and the gender face dataset. Here we do not plot the results as in Fig. 19 due to the increased readability.

6.10 Visualization of Active Selection

It is always interesting to see how the samples are selected in the active learning process. Figure 20 presents some examples that are selected actively in the early stages. As we can see, the results are sensible as a lot of examples picked up in the early stage present cluttered background, heavy blurring, and several of them are baby faces. It is well known that it is not easy to recognize the gender of the baby from their facial images.

6.11 Runtime

Our algorithm has been tested with experiments that are performed on the Window server 2008 with 2.40 GHz Intel(R) Xeon(R) CPU E5645, 48 GB RAM, and Microsoft Visual Studio. Net 2013. It usually costs about 45–60 min to collect 500 labeled data by running our proposed JGPC-ASAL/JGPC-AS on each dataset, while it takes about 20–25 min by our proposed PASS-JGPC-AS.

7 Discussions

A set of findings are revealed in our experiments. We summarize them and discuss potential directions to be further explored. The main observations from our experiments are

(a) It is observed that the entropy based active learning criterion works well. It always obtains better recognition performance than the corresponding random learning counterpart.
(b) The proposed joint Gaussian process model outperforms all the competing methods. It is more resilient to label noise, and is valid to model the expertise level for each individual labeler.
(c) It is verified that the proposed model can achieve higher performance with the learned optimal parameterized kernel.
(d) With the Predictive Active Set Selection method, the proposed Bayesian model can achieve higher efficiency and still preserve the high accuracy.
(e) This paper has obtained the good performance for the binary classification tasks on the datasets with the real crowd-sourced labels. Our proposed model is able to deal with multi-class classifications by reducing them to multiple one versus all binary classification problems.

Meanwhile, there are several things to be further explored, i.e.,

(a) Could better active learning criteria be derived for the Bayesian model with multiple copies of labels? E.g, to support batch active learning, and to balance between exploration and exploitation?
(b) Can we have a formulation to better deal with the case that the raw label accuracy is low, even lower than 50 %?
(c) Is there a better way to have a joint treatment of all the labels from multiple labelers?
(d) Is there a better way to extend the current model to deal with multi-class classifications robustly?

To give a deep probing into the above questions, we will need some additional modifications to refine our models and we leave these to be our future work.

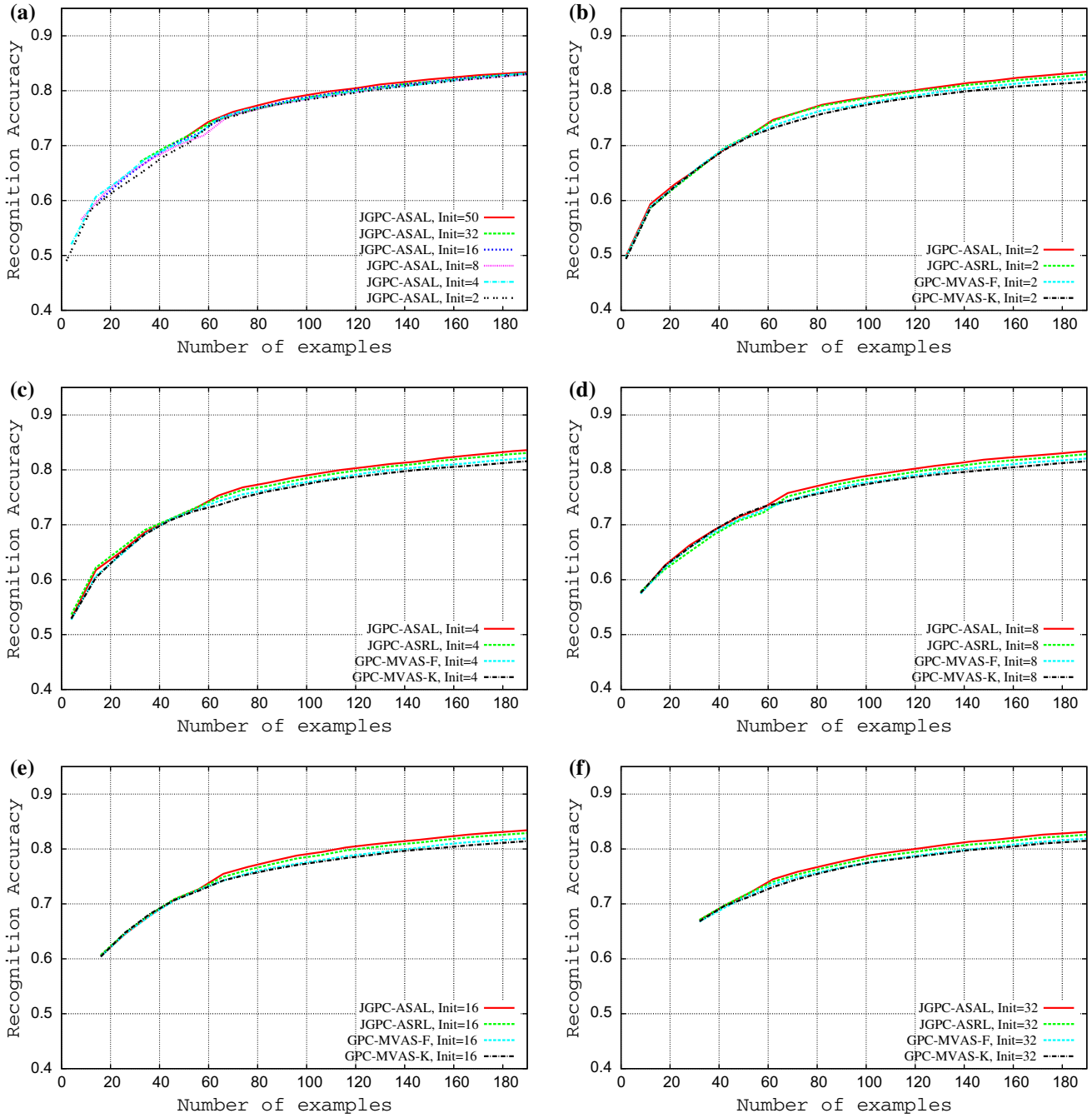

Fig. 19 Recognition performance evaluation with different number of initial labeled examples on the hold-out testing set of the ImageNet dataset. **a** Performance comparison of JGPC-ASAL. **b** With 2 initial labeled examples. **c** With 4 initial labeled examples. **d** With 8 initial labeled examples. **e** With 16 initial labeled examples. **f** With 32 initial labeled examples

8 Conclusion

In this paper, we present a hierarchical Bayesian model to learn a Gaussian process classifier from crowd-sourced labels by jointly considering multiple labels instead of taking the majority voting. Our two-level flip model enables us to design principled active learning strategy to not only select data samples, but also select high quality labelers. Our experiments on four visual recognition datasets with real crowd-sourced labels clearly demonstrated that the active selection of labelers is beneficial when there are a lot of careless labelers. Our joint treatment of multiple labels for each data sample is also proven to be superior to the online majority voting scheme. The Gaussian process classifier

Fig. 20 Some examples selected in the early stages of the active learning process

learned from our model consistently outperforms the one learnt using majority voting strategy. With the learned parameters for the kernel function, the recognition accuracy can be further improved a bit. The extended Bayesian model with the Predictive Active Set Selection Method, not only preserves high recognition accuracy, but also increases the efficiency of our Bayesian active learning system. Our future work will further explore how to design an active learning machine to jointly select both the user and sample in a single criterion.

Acknowledgments Research reported in this publication was partly supported by the National Institute Of Nursing Research of the National Institutes of Health under Award Number R01NR015371. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. This work is also partly supported by US National Science Foundation Grant IIS 1350763, China National Natural Science Foundation Grant 61228303, GH's start-up funds form Stevens Institute of Technology, a Google Research Faculty Award, a gift grant from Microsoft Research, and a gift grant from NEC Labs America.

Appendix 1: Derivation of the Normalization Factor Z_i

$$
\begin{aligned}
Z_i &= \int_{-\infty}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})\{p(+1|s_i,\xi_g) \times p(+1|s_i,\epsilon_j) \\
&\quad + p(-1|s_i,\xi_g)\prod_j p(t_{ij}|-1,\epsilon_j)\}ds_i \\
&= \int_{-\infty}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})\, p(+1|s_i,\xi_g) \\
&\quad \times \prod_j p(t_{ij}|+1,\epsilon_j)ds_i \\
&\quad + \int_{-\infty}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})\, p(-1|s_i,\xi_g) \\
&\quad \times \prod_j p(t_{ij}|-1,\epsilon_j)ds_i \\
&= \int_{-\infty}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})\pounds(\xi_g, s_i) \\
&\quad \times \prod_{t_{ij}\in\{1,-1\}} \pounds(\epsilon_j, t_{ij})ds_i \\
&\quad + \int_{-\infty}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})\pounds(\xi_g, -s_i) \\
&\quad \times \prod_{t_{ij}\in\{1,-1\}} \pounds(\epsilon_j, -t_{ij})ds_i \\
&= \int_{-\infty}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})\pounds(\xi_g, s_i) \\
&\quad \times \prod_{t_{ij}=1} \epsilon_j \prod_{t_{ij}=-1} (1-\epsilon_j)ds_i \\
&\quad + \int_{-\infty}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})\pounds(\xi_g, -s_i) \\
&\quad \times \prod_{t_{ij}=1} (1-\epsilon_j) \prod_{t_{ij}=-1} \epsilon_j ds_i \\
&= \int_{-\infty}^{0} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})C_1 ds_i \\
&\quad + \int_{0}^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})C_2 ds_i, \qquad (37)
\end{aligned}
$$

where

$$
\pounds(\alpha,\beta) = (2\alpha-1)\Theta(\beta) + (1-\alpha), \qquad (38)
$$

$$
\begin{aligned}
C_1 &= (1-\xi_g)\prod_{t_{ij}=-1}(1-\epsilon_j) \\
&\quad \times \prod_{t_{ij}=1}\epsilon_j + \xi_g \prod_{t_{ij}=-1}\epsilon_j \prod_{t_{ij}=1}(1-\epsilon_j), \qquad (39)
\end{aligned}
$$

$$
\begin{aligned}
C_2 &= \xi_g \prod_{t_{ij}=-1}(1-\epsilon_j) \\
&\quad \times \prod_{t_{ij}=1}\epsilon_j + (1-\xi_g)\prod_{t_{ij}=-1}\epsilon_j \prod_{t_{ij}=1}(1-\epsilon_j). \qquad (40)
\end{aligned}
$$

Consider that

$$
\begin{aligned}
&\int_0^{+\infty} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})ds_i \\
&= \int_{-\infty}^{\frac{m_{-i}^{\text{old}}}{\sqrt{v_{-i}^{\text{old}}}}} \mathcal{N}(\tau;0,1)d\tau, = \Phi\left(\frac{m_{-i}^{\text{old}}}{\sqrt{v_{-i}^{\text{old}}}}\right) \qquad (41)
\end{aligned}
$$

$$\int_{-\infty}^{0} \mathcal{N}(s_i|m_{-i}^{\text{old}}, v_{-i}^{\text{old}})ds_i = 1 - \Phi\left(\frac{m_{-i}^{\text{old}}}{\sqrt{v_{-i}^{\text{old}}}}\right), \quad (42)$$

and we can rewrite the eqnarray as:

$$Z_i = (C_2 - C_1)\Phi\left(\frac{m_{-i}^{\text{old}}}{\sqrt{v_{-i}^{\text{old}}}}\right) + C_1. \quad (43)$$

Appendix 2: Moment Matching in the Expectation Propagation Algorithm

Minimizing $KL[Q_{-i}(s_i)p(\mathbf{t}_i|s_i)||Q_{-i}(s_i)\tilde{F}_i(s_i)]$ to obtain an update for the approximation $\tilde{F}_i(s_i)$, we recompute the parameters according to the normalized constant presented in Appendix 1, i.e.,

$$Z_i = (C_2 - C_1)\Phi(z_i) + C_1, \quad (44)$$

where

$$z_i = \frac{m_{-i}^{\text{old}}}{\sqrt{v_{-i}^{\text{old}}}}, \Phi(x) = \int_{-\infty}^{x} \mathcal{N}(\tau; 0, 1)d\tau.$$

By moment matching (Minka 2001), we have

$$m_{-i}^{\text{new}} = m_{-i}^{\text{old}} + v_{-i}^{\text{old}}\alpha, \quad (45)$$

where $\alpha = \frac{1}{\sqrt{v_{-i}^{\text{old}}}} \cdot \frac{(C_2-C_1)\mathcal{N}(z_i;0,1)}{Z_i}$. Hence we can obtain a new $\tilde{F}_i(s_i)$ by recomputing its parameters A_i, $\tilde{m}_i$, and v_i as

$$v_i = v_{-i}^{\text{old}}\left(\frac{1}{m_{-i}^{\text{new}}\alpha} - 1\right), \quad (46)$$

$$\tilde{m}_i = m_{-i}^{\text{new}} + v_i\alpha, \quad (47)$$

$$A_i = Z_i\sqrt{1 + v_i{}^{-1}v_{-i}^{\text{old}}}\exp\left(\frac{v_{-i}^{\text{old}}\alpha}{2m_{-i}^{\text{new}}}\right). \quad (48)$$

Appendix 3: Gradients of the Lower Bound

The gradients of the lower bound with respect to the parameters $\boldsymbol{\varepsilon}$ are as follows

$$\frac{\partial F_{\boldsymbol{\varepsilon}}}{\partial \xi_g} = \sum_{i=1}^{N} q(m_{s_i}) \frac{\aleph(m_{s_i})\prod_j p(t_{ij}|+1, \epsilon_j)}{p(\mathbf{t_i}|m_{s_i})} - \sum_{i=1}^{N} q(m_{s_i}) \frac{\aleph(m_{s_i})\prod_j p(t_{ij}|-1, \epsilon_j)}{p(\mathbf{t_i}|m_{s_i}, \boldsymbol{\varepsilon})}, \quad (49)$$

$$\frac{\partial F_{\boldsymbol{\varepsilon}}}{\partial \varepsilon_j} = \sum_{i=1}^{N} q(m_{s_i}) \frac{\aleph(t_{ij})\Re(+1, i, j)}{p(\mathbf{t_i}|m_{s_i}, \boldsymbol{\varepsilon})} - \sum_{i=1}^{N} q(m_{s_i}) \frac{\aleph(t_{ij})\Re(-1, i, j)}{p(\mathbf{t_i}|m_{s_i}, \boldsymbol{\varepsilon})}, \quad (50)$$

where

$$\aleph(x) = 2\Theta(x) - 1, \quad (51)$$

$$\Re(y_i, i, j) = p(y_i|m_{s_i}, \xi_g)\prod_{j' \neq j} p(t_{ij'}|y_i, \epsilon_{j'}). \quad (52)$$

And the gradient with respect to a kernel parameter $\theta \in \boldsymbol{\vartheta}$ is

$$\frac{\partial F_{\boldsymbol{\vartheta}}}{\partial \theta} = -\frac{1}{2}tr(\mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial \theta} + \frac{1}{2}E_q[\mathbf{S}_L]^T\mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial \theta}\mathbf{K}^{-1}E_q[\mathbf{S}_L] + \frac{1}{2}tr(\mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial \theta}\mathbf{K}^{-1}Cov[\mathbf{S}_L]). \quad (53)$$

References

Ambati, V., Vogel, S., & Carbonell, J. (May 2010). Active learning and crowd-sourcing for machine translation. In: *Proceedings of the Seventh International Conference on Language Resources and Evaluation* (LREC'10).

Branson, S., Perona, P., & Belongie, S. (November 2011). Strong supervision from weak annotation: Interactive training of deformable part models. In: *Proceedings of the IEEE International Conference on Computer Vision*, Barcelona, Spain.

Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., & Belongie, S. (September 2010). Visual recognition with humans in the loop. In: *Proceedings of the European Conference on Computer Vision*, Heraklion, Crete.

Burl, M., & Perona, P. (1998). Using hierarchical shape models to spot keywords in cursive handwriting data. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (CVPR) (pp. 23–28). IEEE.

Burl, M., Leung, T. K., & Perona, P. (1995). Face localization via shape statistics. In: *Proceedings of the First International Workshop on Automatic Face and Gesture Recognition* (pp. 154–159). Zurich.

Chen, S., Zhang, J., Chen, G., & Zhang, C. (2010). What if the irresponsible teachers are dominating? A method of training on samples and clustering on teachers. In: *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence* (AAAI).

Dekel, O., & Shamir, O. (2009). Good learners for evil teachers. In: *Proceedings of the IEEE International Conference on Machine Learning*. IEEE.

Dekel, O., & Shamir, O. (2009). Vox populi: Collecting high-quality labels from a crowd. In: *Proceedings of the 22nd Annual Conference on Learning Theory*.

Deng, J., Krause, J., & Fei-Fei, L. (June 2013). Fine-grained crowdsourcing for fine-grained recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE.

Deng, J., Dong, W., Socher, R., Li, L. -J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In: *Proceedings of the IEEE International Conference on Computer Vision* (pp. 248–255), June 2009. IEEE.

Donmez, P., Carbonell, J., & Schneider, J. (2009). Efficiently learning the accuracy of labeling sources for selective sampling. In: *Special Interest Group on Knowledge Discovery in Data* (SIGKDD).

Donmez, P., Carbonell, J., & Schneider, J. (2010). A probabilistic framework to learn from multiple annotators with time-varying accuracy. In: *Proceedings of the SIAM Conference on Data Mining* (SDM). Philadelphia: SIAM.

Ebert, S., Fritz, M., & Schiele, B. (2012). Ralf: A reinforced active learning formulation for object class recognition. In: *Proceedings of the IEEE International Conference on Computer Vision*. IEEE.

Fergus, R., Fei-Fei, L., Perona, P., & Zisserman, A. (Oct. 2005). Learning object categories from google's image search. In: *Proceedings of the 10th International Conference on Computer Vision*, Beijing.

Gibbs, M., & Mackay, D. (2000). Variational gaussian process classifiers. *IEEE Transactions on Neural Networks 11*(6), 1458–1464.

Groot, P., Birlutiu, A., & Heskes, T. (2011). Learning from multiple annotators with gaussian processes. In: *Proceedings of the Artificial Neural Networks and Machine Learning—ICANN 2011 21st International Conference on Artificial Neural Networks, Part II, Espoo, June 14–17, 2011* (pp. 159–164).

Henao, R., & Winther, O. (2010). Pass-gp: Predictive active set selection for gaussian processes. In: *Proceedings of the Machine Learning for Signal Processing (MLSP), 2010 IEEE International Workshop* (p. 148153).

Henao, R., & Winther, O. (2012). Predictive active set selection methods for gaussian processes. *Neurocomputing*, *80*, 10–18.

Hua, G., Long, C., Yang, M., & Gao, Y. (2013). Collaborative active learning of a kernel machine ensemble for recognition. In: *Proceedings IEEE International Conference on Computer Vision* (pp. 1209–1216). IEEE

Kapoor, A., Grauman, K., Urtasun, R., & Darrell, T. (2007). Active learning with gaussian processes for object categorization. In: *Proceedings IEEE International Conference on Computer Vision*.

Kapoor, A., Hua, G., Akbarzadeh, A., & Baker, S. (2009). Which faces to tag: Adding prior constraints into active learning. In: *Proceedings IEEE International Conference on Computer Vision*. IEEE

Kim, H.-C., & Ghahramani, Z. (2006). Bayesian gaussian process classification with the EM-EP algorithm. *IEEE Transactions Pattern Analysis and Machine Intelligence, 28*(12), 1948–1959.

Kim, H.-C., & Ghahramani, Z. (2008). Outlier robust gaussian process classification. In: *Proceedings of the Structural, Syntactic, and Statistical Pattern Recognition. Joint IAPR International Workshops (SSPR/SPR)* (pp. 896–905).

Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. *Neural Information Processing Systems, 1*, 31–40.

Lawrence, N. D., Seeger, M., & Herbrich, R. (2003). Fast sparse gaussian process methods: The informative vector machine. In: *Advances in Neural Information Processing Systems* (vol. 15, pp. 609–616). Cambridge: MIT Press.

Lin, Y., Lv, F., Zhu, S., Yang, M., Cour, T., Yu, K., Cao, L., & Huang, T. (2011). Large-scale image classification: Fast feature extraction and svm training. In: *Proceedings IEEE International Conference on Computer Vision*. IEEE

Liu, D., Hua, G., Viola, P., & Chen, T. (2008). Integrated feature selection and higher-order spatial feature extraction for object categorization. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* CVPR 2008 (pp. 1–8). IEEE.

Long, C., Hua, G., & Kapoor, A. (December 2013). Active visual recognition with expertise estimation in crowdsourcing. In: *Proceedings IEEE International Conference on Computer Vision*. IEEE

Loy, C., Hospedales, T., Xiang, T., & Gong, S. (2012). Stream-based joint exploration-exploitation active learning. In: *Proceedings IEEE International Conference on Computer Vision*. IEEE

Minka, T. (2001). *A family of algorithms for approximate Bayesian inference*. Ph.D. Thesis. Cambridge: MIT.

Naish-Guzman, A., & Holden, S.B. (2007). The generalized FITC approximation. In: *Neural Information Processing Systems* (NIPS) (pp. 1057–1064).

Neal, R.M. (1997). Monte carlo implementation of gaussian process models for bayesian regression and classification. Technical Report CRGTR972, University of Toronto.

Opper, M., & Winther, O. (1999). Gaussian processes for classification: Mean-field algorithms. *Neural Computation*, *12*, 2000.

Parikh, D. (November 2011). Recognizing jumbled images: The role of local and global information in image classification. In: *Proceedings IEEE International Conference on Computer Vision*. IEEE

Parikh, D., & Zitnick, L. (June 2010). The role of features, algorithms and data in visual recognition. In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*. IEEE

Parikh, D., & Zitnick, L. (June 2011). Finding the weakest link in person detectors. In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*. IEEE

Parikh, D., Zitnick, C. L., & Chen, T. (2012). Exploring tiny images: The roles of appearance and contextual information for machine and human object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(10), 1978–1991.

Patterson, G., Horn, G. V., Belongie, S., Perona, P., & Hays, J. (2013). Bootstrapping fine-grained classifiers: Active learning with a crowd in the loop. In: *Proceedings Neural Information Processing Systems (NIPS) 2013 Crowd Workshops*.

Quinonero-candela, J., Rasmussen, C. E., & Herbrich, R. (2005). A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, *6*, 2005.

Rasmussen, C. E. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.

Raykar, V. C., & Yu, S. (2012). Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research, 13*, 491–518 .

Raykar, V. C., Yu, S., Zhao, L. H., Jerebko, A., Florin, C., Valadez, G. H., Bogoni, L., & Moy, L. (2009). Supervised learning from multiple experts: whom to trust when everyone lies a bit. In: *Proceedings IEEE International Conference on Machine Learning*. IEEE

Rodrigues, F., Pereira, F. C., & Ribeiro, B. (2013). Learning from multiple annotators: Distinguishing good from random labelers. *Pattern Recognition Letters*, *34*(12), 1428–1436.

Rodrigues, F., Pereira, F., & Ribeiro, B. (2014). Gaussian process classification and active learning with multiple annotators. In: *Proceedings IEEE International Conference on Machine Learning*. IEEE

Rodrigues, F., Pereira, F.C., & Ribeiro, B. (2013). Sequence labeling with multiple annotators. *Machine Learning, 95*(2), 165–181.

Roy, N., & Mccallum, A. (2001). Toward optimal active learning through sampling estimation of error reduction. In: *Proceedings IEEE International Conference on Machine Learning*, pp. 441–448. Burlington, MA: Morgan Kaufmann.

Sanchez, J., & Perronnin, F. (2011). High-dimensional signature compression for large-scale image classification. In: *Proceedings IEEE International Conference on Computer Vision*. IEEE

Seeger, M. (2002). Pac-Bayesian generalisation error bounds for gaussian process classification. *Journal of Machine Learning Research*, *3*, 233–269.

Seeger, M., Williams, C. K. I., & Lawrence, N. D. (2003). Fast forward selection to speed up sparse gaussian process regression. In: *Proceedings of the Workshop on Artificial Intelligence and Statistics*, (vol. 9).

Simpson, E., Roberts, S. J., Psorakis, I., & Smith, A. (2013). Dynamic bayesian combination of multiple imperfect classifiers. In: *Decision Making and Imperfection* (pp. 1–35). Berlin: Springer

Snelson, E., & Ghahramani Z. (2006). Sparse gaussian processes using pseudo-inputs. In: *Advances in Neural information Processing Systems* (pp. 1257–1264). Cambridge: MIT Press.

Snelson, E., & Ghahramani, Z. (2006). Variable noise and dimensionality reduction for sparse gaussian processes. In: *Proceedings of the Conference on Uncertainty in Artificial Intelligence* (UAI). Edinburgh: AUAI Press.

Spriggs, E. H., Torre, F. D. L., & Hebert, M. (2009). Temporal segmentation and activity classification from first-person sensing. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*. IEEE

Thouless, D. J., Anderson, P. W., & Palmer, R. G. (1977). Solution of a "solvable model of a spin glass". *Philosophical Magazine*, *35*, 593.

Titsias, M. K. (2009). Variational learning of inducing variables in sparse gaussian processes. *Artificial Intelligence and Statistics*, *12*, 567–574.

Tivive, F. H. C., & Bouzerdoum, A. (2006). A gender recognition system using shunting inhibitory convolutional neural networks. In: *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2006, Part of the IEEE World Congress on Computational Intelligence, WCCI* 2006, Vancouver, BC, 16–21 July 2006, (pp. 5336–5341). IEEE

Vijayanarasimhan, S., & Grauman, K. (2014). Large-scale live active learning: Training object detectors with crawled data and crowds. *International Journal of Computer Vision (IJCV)*, *108*(1–2), 97–114.

von Ahn, L., & Dabbish, L. (2004). Labeling images with a computer game. In: *Proceedings ACM Conference on Human Factors in Computing Systems* (pp. 319–326). New York, NY: ACM

von Ahn, L., Liu, R., & Blum, M. (2006). Peekaboom: A game for locating objects in images. In: *Proceedings ACM Conference on Human Factors in Computing Systems* (pp. 55–64). New York, NY: ACM

Vondrick, C., & Ramanan, D. (2011). Video annotation and tracking with active learning. In: *Neural Information Processing Systems* (NIPS) (pp. 28–36). Cambridge, MA: MIT Press

Wah, C., Branson, S., Perona, P., & Belongie, S. (November 2011). Multiclass recognition and part localization with humans in the loop. In: *Proceedings IEEE International Conference on Computer Vision, Barcelona, Spain*. IEEE

Welinder, P., & Perona, P. (June 2010). Online crowdsourcing: rating annotators and obtaining cost-effective labels. San Francisco. In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*. IEEE

Welinder, P., & Perona., P. (2010). Online crowdsourcing: Rating annotators and obtaining cost-effective labels. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*. IEEE

Welinder, P., Branson, S., Belongie, S., & Perona, P. (2010). The multidimensional wisdom of crowds. In: *Neural Information Processing Systems* (NIPS).

Williams, C., & Barber, D. (1998). Bayesian classification with gaussian processes. *IEEE Trans Pattern Analysis and Machine Intelligence*, *20*(12), 1342–1351.

Wu, O., Hu, W., & Gao, J. (2011). Learning to rank under multiple annotators. In: *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Two* (pp. 1571-1576). Menlo Park, CA: AAAI Press

Yan, F., & Qi, Y. A. (2010). Sparse gaussian process regression via l1 penalization. In: *Proceedings IEEE International Conference on Machine Learning* (pp. 1183–1190). IEEE

Yan, Y., Rosales, R., Fung, G., & Dy, J. G. (2011). Active learning from crowds. In: *Proceedings IEEE International Conference on Machine Learning* (pp. 1161–1168). IEEE

Yan, Y., Rosales, R., Fung, G., & Dy, J. (2012). Active learning from multiple knowledge sources. In: *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics* (AISTATS).

Yao, A., Gall, J., Leistner, C., & Van Gool, L. (2012). Interactive object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE

Yao, B., Khosla, A., & Fei-Fei, L. (2011). Combining randomization and discrimination for fine-grained image categorization. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Springs, Colorado, June 2011. IEEE

Zhang, Z., Dai, G., & Jordan, M. I. (2011). Bayesian generalized kernel mixed models. *Journal of Machine Learning Research*, *12*, 111–139.

Zhao, L., Sukthankar, G., & Sukthankar, R. (2011). Incremental relabeling for active learning with noisy crowdsourced annotations. In: *Proceedings of the IEEE Third International Conference on and 2011 IEEE Third International Conference on Social Computing* (SocialCom). IEEE

Zhu, X., Lafferty, J., & Ghahramani, Z. (2003). Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In: *Proceedings of the ICML 2003 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining* (pp. 58–65).

Zhu, C., Byrd, R. H., Lu, P., & Nocedal, J. (1997). Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, *23*, 550–560.

Zitnick, L., & Parikh, D. (2012). The role of image understanding in contour detection. In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition* (pp. 622–629). IEEE