International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:9, No:8, 2015

Variational EM Inference Algorithm for Gaussian Process Classification Model with Multiclass and Its Application to Human Action Classification

Wanhyun Cho, Soonja Kang, Sangkyoon Kim, Soonyoung Park

***Abstract*—**In this paper, we propose the variational EM inference algorithm for the multi-class Gaussian process classification model that can be used in the field of human behavior recognition. This algorithm can drive simultaneously both a posterior distribution of a latent function and estimators of hyper-parameters in a Gaussian process classification model with multiclass. Our algorithm is based on the Laplace approximation (LA) technique and variational EM framework. This is performed in two steps: called expectation and maximization steps. First, in the expectation step, using the Bayesian formula and LA technique, we derive approximately the posterior distribution of the latent function indicating the possibility that each observation belongs to a certain class in the Gaussian process classification model. Second, in the maximization step, using a derived posterior distribution of latent function, we compute the maximum likelihood estimator for hyper-parameters of a covariance matrix necessary to define prior distribution for latent function. These two steps iteratively repeat until a convergence condition satisfies. Moreover, we apply the proposed algorithm with human action classification problem using a public database, namely, the KTH human action data set. Experimental results reveal that the proposed algorithm shows good performance on this data set.

***Keywords*—**Bayesian rule, Gaussian process classification model with multiclass, Gaussian process prior, human action classification, laplace approximation, variational EM algorithm.

I. Introduction

MANY studies on the classification method using the Gaussian process model have recently been conducted. The most representative research has been performed by Rasmussen and Williams [1]. They investigated the general theory on the problem of using a Gaussian stochastic process in machine learning. Nickisch and Rasmussen [2] provide a comprehensive overview of many recent algorithms used for approximate inference in Gaussian process models for the purpose of probabilistic binary classification. The relationships between these several approaches are elucidated theoretically, and the properties of the different algorithms are corroborated through experimental results. Chan and Dong [3] propose a generalized Gaussian process model (GGPM), which is a unifying framework that encompasses many existing Gaussian process models, such as Gaussian process regression, classification, and counting. They derive a close-form efficient Taylor approximation for inference on the model, and draw interesting connections to the other model-specific closed-form approximations. Chan [4] proposes a family of multivariate Gaussian process models for correlated output. This is based on the assumption that the likelihood function takes the generic form of the multivariate exponential family distribution. In [4], this model is defined as a multivariate GGPM, and Taylor and Laplace algorithms are derived for approximate inference on the generic model. Kim and Ghahramani [5] present an approximate EM algorithm, the EM-EP algorithm, to learn both the latent function and hyper-parameters in a Gaussian process classification model. Rasmussen and Nickisch [6] have recently proposed the Gaussian Process for Machine Learning Toolbox version 3.4 for implementing inference and prediction in Gaussian process models.

Recently, several papers were published that apply Gaussian stochastic process models of human behavior recognition. Raskin et al. [7] presented an approach for tracking human body parts and classification of human actions. They introduce Gaussian processing Annealed Particle Filter Tracker, which is an extension of the annealed particle filter tracker and uses Gaussian process Dynamical Model in order to reduce the dimensionality of the problem, increase the tracker's stability, and learn the motion models. Zhou et al. [8] presents a spectral analysis-based feature-reduced Gaussian Processes classification approach to recognition of articulated and deformable human actions from image sequences. Using Tensor Subspace Analysis, space-time human silhouettes extracted from action sequences are transformed to a low dimensional multivariate time series, from which structure-based statistical features are extracted to summarize the action properties. Gaussian processes classification based on spectrally reduced features is then applied to learn and predict action categories. Zhao et al. [9] considered probabilistic multinomial probit classification for tensor-variety inputs with Gaussian processes priors placed over the latent function. In order to take into account the underlying multi-modes structure information within the model, they propose a framework of probabilistic product kernels for tonsorial data based on a generative model assumption.

Wanhyun Cho is with the Department of Statistics, Chonnam National University, Gwangju, 500-757 South Korea (phone: 82-62-530-3443; fax: 82-62-530-3449; e-mail: whcho@ chonnam.ac.kr).

Soonja Kang is with the Department of Mathematical Education, Chonnam National University, Gwangju, 500-757 South Korea (e-mail: sjkang@ chonnam.ac.kr).

Sangkyoon Kim and Soonyoung Park are with the Department of Information & Electronics Engineering, Mokpo National University, Muan, Jeonnam, South Korea (e-mail: narciss76@mokpo.ac.kr, sypark@mokpo.ac.kr).

This work was jointly supported by the National Research Foundation of Korea Government (2014R1A1A4A0109398) and the Research Foundation of Chonnam National University (2014-2256).

The major contribution of our research is an inference algorithm that can drive simultaneously both a posterior distribution of a latent function and estimators of hyper-parameters in a Gaussian process classification model. The proposed algorithm is performed in two steps: each of which is the expectation step and maximization step. First, in the expectation step, using the Bayesian formula and LA, we derive approximately the posterior distribution of the latent function based on learning data. Furthermore, we calculate a mean vector and covariance matrix of the latent function. Second, in the classification step, using a derived posterior distribution of the latent function, we derive the maximum likelihood estimator for hyper-parameters necessary to define a covariance matrix. Finally, we apply the proposed algorithm with human action classification problem using a public database, namely, the KTH human action data set.

II. Multiclass Gaussian Process Classification Model

A. Gaussian Process Classification Model

In general, the Bayesian Gaussian process classification model with multiclass consists of three components: a latent function with Gaussian process (GP) prior, a response function with multiclass, and a link function that relates the latent function and response mean. First, we define the latent function $\mathbf{f}(\mathbf{x})$ for multiclass classification at each function having C classes as

$$\begin{aligned}&\mathbf{f}(\mathbf{x}) = (\mathbf{f}^1, \cdots, \mathbf{f}^c, \cdots, \mathbf{f}^C)^T, \\ &\mathbf{f}^c = (\mathrm{f}_1^c, \cdots, \mathrm{f}_i^c, \cdots \mathrm{f}_n^c)^T, c = 1, \cdots, C\end{aligned} \tag{1}$$

where the superscript c denotes a particular class and the subscript i denotes the observation number. Then, we assume that a GP prior for the latent function $\mathbf{f}(\mathbf{x})$ is defined as the Gaussian distribution having zero mean vector and covariance matrix **K.** In other words

$$\mathbf{f}(\mathbf{x}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}). \tag{2}$$

Here, the GP prior for multiclass classification usually has only intra-class correlations. The covariance matrix $\mathbf{K}$ in the GP prior of latent functions is defined as

$$\mathbf{K} = \begin{pmatrix} \mathbf{K^1} & \cdots & \mathbf{0} \\ \vdots & \mathbf{K^c} & \vdots \\ \mathbf{0} & \cdots & \mathbf{K^C} \end{pmatrix}, \tag{3}$$

where $\mathbf{K^c}$ is a covariance matrix of latent function $\mathbf{f}^c(\mathbf{x})$ related to class c. The covariance function $\mathrm{k}_{\mathrm{ij}}^{\mathrm{c}}$ of each covariance matrix $\mathbf{K^c}$ is defined by

$$\begin{aligned}&\mathbf{K}^c = \left(k_{ij}^c\right), c = 1, \cdots, C \\ &k_{ij}^c = k^c(\mathbf{x}_i, \mathbf{x}_j) = \theta_0^c \exp(-\frac{1}{2\theta_1^c} \| \mathbf{x}_i - \mathbf{x}_j \|), i, j = 1, \cdots, n\end{aligned} \tag{4}$$

The hyper-parameter θ_0^{c} specifies the latent function scale related to class c, whereas the hyper-parameter θ_1^{c} specifies a function of the length scale. Here, we represent $\Theta = \{\theta_0^1, \theta_1^1, \cdots, \theta_0^C, \theta_1^C\}$ as the set of all hyper-parameters of the model.

Second, the response vector $\mathbf{Y} = \left(\mathbf{y}^1, \cdots, \mathbf{y}^c, \cdots, \mathbf{y}^C\right)^T$ consists of independent and identical multinomial random variables in which each component vector $\mathbf{y}^c$ represents the c-th class. In other words, let us define $\mathbf{Y} = (\mathbf{y}^1, \cdots, \mathbf{y}^c, \cdots, \mathbf{y}^C)^T$ as a vector of the same length $\mathbf{f}(\mathbf{x})$ in which each component of the c-th random vector $\mathbf{y}^c = (\mathrm{y}_1^c, \cdots, \mathrm{y}_i^c, \cdots y_n^c)^T$ for $c = 1, \cdots, C$ contains all entries of 1 for the c class which is the label for the i-th observation and 0 is for the other $C-1$ classes. Here, we can assume that the density function $p(\mathbf{Y}|\boldsymbol{\pi})$ of the response vector $\mathbf{Y}$ is given as the following form:

$$p(\mathbf{Y} \mid \boldsymbol{\pi}) = \prod_{i=1}^{n} \prod_{c=1}^{C} (\pi_i^c)^{y_i^c} \tag{5}$$

where the indicator variable y_i^c is 1 or 0 with probability π_i^c and $1\text{-}\pi_i^c$ and π_i^c denotes the probability that an i-th observation is a type of the particular class c.

Third, we consider the link function that specifies the relation between the latent function $\mathbf{f}(\mathbf{x})$ and the response mean vector $E(\mathbf{Y}|\mathbf{f})$. Here, the link function can be defined as

$$E(\mathbf{Y} \mid \mathbf{f}) = (E(\mathbf{y}^1 \mid \mathbf{f}), \cdots, E(\mathbf{y}^C \mid \mathbf{f})) \tag{6}$$

where

$$\begin{aligned}E(\mathbf{y}^c \mid \mathbf{f}) = (E(y_1^c \mid \mathbf{f}), \cdots, E(y_i^c \mid \mathbf{f}), \cdots, E(y_n^c \mid \mathbf{f})), \\ \mathrm{c} = 1, \cdots, \mathbf{C}\end{aligned} \tag{7}$$

and

$$E(y_i^c \mid \mathbf{f}) = \pi_i^c = \frac{\exp(\mathrm{f}_i^c)}{\sum_{c^*=1}^{C} \exp(\mathrm{f}_i^{c^*})}, \; i = 1, \cdots, n, \; c = 1, \cdots C. \tag{8}$$

B. Laplace Approximation Algorithm

Inference in the classification problem is processed naturally by two steps. First, by using Bayes' rule, we compute the posterior distribution of the latent variable $\mathbf{f}(\mathbf{x})$ based on n training data as:

$$p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}, \Theta) = p(\mathbf{y} \mid \mathbf{f}, \Theta) p(\mathbf{f} \mid \mathbf{X}, \Theta) / \, p(\mathbf{y} \mid \mathbf{X}, \Theta) \tag{9}$$

Second, using posterior distribution $p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}, \Theta)$ derived at the first step, we compute the predictive distribution of the latent variable $\mathrm{f}_*(\mathbf{x}_*)$ corresponding to a test case $\mathbf{x}_*$:

$$p(\mathrm{f}_* \mid \mathbf{X}, \mathbf{y}, \Theta, \mathbf{x}_*) = \int p(\mathrm{f}_* \mid \mathbf{X}, \mathbf{x}_*, \mathbf{f}) \, p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}, \Theta) d\mathbf{f}. \tag{10}$$

In addition, we produce a predictive probability which a test point $\mathbf{x}_*$ is belonging to some class c:

$$\begin{aligned}\tilde{\pi}_*^c &= p(\mathrm{y}_*^c \mid \mathbf{X}, \mathbf{y}, \mathbf{x}_*, \Theta) \\ &= \int p(\mathrm{y}_*^c \mid \mathrm{f}_*, \Theta)\, p(\mathrm{f}_* \mid \mathbf{X}, \mathbf{y}, \mathbf{x}_*, \Theta) d\mathrm{f}_*,\ c = 1, \cdots, C\end{aligned} \tag{11}$$

However, the likelihood function, $p(\mathbf{y} \mid \mathbf{f}, \Theta)$ considered at the first step is given as the non-Gaussian, and it makes the integral analytically intractable. Thus, we need to use either analytic approximations of integrals or solutions based on Monte Carlo sampling.

Here, we will consider the Laplace approximation method that can approximate the non-Gaussian posterior $p(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)$ with a Gaussian posterior $q(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)$. This approximation means that performing a second-order Taylor expansion for the log-posterior $\Psi(\mathbf{f}) = \ln p(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)$ at the mode $\mathbf{m}_\mathbf{F}$ of the posterior, i.e.

$$\mathbf{m}_\mathbf{f} = \arg\max_\mathbf{f} \Psi(\mathbf{f}) = \arg\max_\mathbf{f} \ln p(\mathbf{Y} \mid \mathbf{f})\, p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta}) \tag{12}$$

It gives us:

$$\begin{aligned}\Psi(\mathbf{f}) \cong\ & \Psi(\mathbf{m}_\mathbf{f}) + \nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}} (\mathbf{f} - \mathbf{m}_\mathbf{f}) \\ & + \frac{1}{2}(\mathbf{f} - \mathbf{m}_\mathbf{f})^T (\nabla\nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}})(\mathbf{f} - \mathbf{m}_\mathbf{f})\end{aligned} \tag{13}$$

Here, taking the logarithm of the un-normalized posterior of the latent function $\mathbf{f}$, it can be given as

$$\begin{aligned}\Psi(\mathbf{f}) &= \ln p(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta) = \ln p(\mathbf{f} \mid \mathbf{X}, \Theta) + \ln p(\mathbf{Y} \mid \mathbf{f}) \\ &= \ln p(\mathbf{Y} \mid \mathbf{f}) - \frac{1}{2}\mathbf{f}^T \mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\ln|\mathbf{K}| - \frac{n}{2}\ln 2\pi\end{aligned} \tag{14}$$

Moreover, taking the first and second derivatives of log posterior with respect to $\mathbf{f}$, we obtain

$$\begin{aligned}\nabla\Psi(\mathbf{f}) &= \nabla \ln p(\mathbf{Y} \mid \mathbf{f}) - \mathbf{K}^{-1}\mathbf{f} \\ \nabla\nabla\Psi(\mathbf{f}) &= \nabla\nabla \ln p(\mathbf{Y} \mid \mathbf{f}) - \mathbf{K}^{-1} = -\mathbf{W} - \mathbf{K}^{-1}\end{aligned} \tag{15}$$

where $\mathbf{W} \equiv -\nabla\nabla \ln p(\mathbf{Y} \mid \mathbf{F})$ is diagonal matrix. Hence, we have obtained:

$$\begin{aligned}\Psi(\mathbf{f}) &\cong \Psi(\mathbf{m}_\mathbf{f}) - \frac{1}{2}(\mathbf{f} - \mathbf{m}_\mathbf{f})^T (\mathbf{W} + \mathbf{K}^{-1})(\mathbf{f} - \mathbf{m}_\mathbf{f}) \\ &\cong \ln N(\mathbf{f} \mid \mathbf{m}_\mathbf{f}, (\mathbf{K}^{-1} + \mathbf{W})^{-1})\end{aligned} \tag{16}$$

Finally, we have obtained a Gaussian approximate posterior $q(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)$ to the true posterior $p(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)$ with mean vector $\mathbf{m}_\mathbf{f}$ and covariance matrix $\mathbf{V} = (\mathbf{K}^{-1} + \mathbf{W})^{-1}$. Here, the mean vector $\mathbf{m}_\mathbf{f}$ of Gaussian approximate posterior or mode $\mathbf{m}_\mathbf{f}$ of the log-posterior $\Psi(\mathbf{f})$ can be found iteratively using the Newton-Rapson algorithm. In particular, given an initial estimate $\mathbf{m}_\mathbf{f}$, a new estimate is found iteratively according to

$$\begin{aligned}\mathbf{m}_\mathbf{f}^{new} &= \mathbf{m}_\mathbf{f} - (\nabla\nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}})^{-1} \nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}} \\ &= \mathbf{m}_\mathbf{f} + (\mathbf{K}^{-1} + \mathbf{W})^{-1}(\nabla \ln p(\mathbf{Y} \mid \mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}} - \mathbf{K}^{-1}\mathbf{m}_\mathbf{f}) \\ &= (\mathbf{K}^{-1} + \mathbf{W})^{-1}(\mathbf{W}\mathbf{m}_\mathbf{f} + \nabla \ln p(\mathbf{Y} \mid \mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{F}})\end{aligned} \tag{17}$$

Second, the matrix $\mathbf{W}$ at the covariance matrix $\mathbf{V} = (\mathbf{K}^{-1} + \mathbf{W})^{-1}$ can be given as the following form. Since the log-likelihood function $\ln p(\mathbf{Y} \mid \mathbf{f})$ can be expressed as $\sum_{i=1}^{n} \ln p(y_i^1, \cdots, y_i^C \mid \mathbf{f}_i)$, we obtain the following equation by differentiating the log-likelihood function $\ln p(\mathbf{Y} \mid \mathbf{f})$ with respect to $\mathbf{f}$,

$$\begin{aligned}\nabla_\mathbf{f} \ln p(\mathbf{Y} \mid \mathbf{f}) &= \nabla_\mathbf{f} \Big(\sum_{i=1}^{n} \ln p(y_i^1, \cdots, y_i^C \mid \mathbf{f}_i)\Big) \\ &= \nabla_\mathbf{F} \left(\sum_{i=1}^{n}\sum_{c=1}^{C} y_i^c \mathrm{f}_i^c - \sum_{i=1}^{n} \ln \Big(\sum_{c=1}^{C} \exp(\mathrm{f}_i^c)\Big) \right) \\ &= \mathbf{Y} - \boldsymbol{\pi}\end{aligned} \tag{18}$$

where a vector $\boldsymbol{\pi}$ is defined by

$$\begin{aligned}\boldsymbol{\pi}_{(nC\times 1)} &= (\pi_1^1, \cdots, \pi_i^c, \cdots, \pi_n^C)^T \\ \pi_i^c &= \frac{\exp(\mathrm{f}_i^c)}{\sum_{c^*=1}^{C} \exp(\mathrm{f}_i^{c^*})},\ i = 1, \cdots, \mathrm{n},\ c = 1, \cdots, C\end{aligned} \tag{19}$$

Hence, the matrix $\mathbf{W}$ can be given as

$$\mathbf{W} = -\nabla\nabla \ln p(\mathbf{Y} \mid \mathbf{f}) = \left(-\frac{\partial \ln p(\mathbf{Y} \mid \mathbf{f})}{\partial \mathbf{f}\, \partial \mathbf{f}^T} \right) = \mathrm{diag}(\boldsymbol{\pi}) - \boldsymbol{\Pi}^T \boldsymbol{\Pi} \tag{20}$$

where $\boldsymbol{\Pi}$ is a $(n \times nC)$ matrix obtained by stacking horizontally the diagonal matrices $\mathrm{diag}(\boldsymbol{\pi}^c)$. This is given as the following form:

$$\boldsymbol{\Pi} = \begin{bmatrix} \pi_1^1 & \cdots & \pi_1^C & & \\ & \ddots & \cdots & \ddots & \\ & \pi_n^1 & \cdots & & \pi_n^C \end{bmatrix} \tag{21}$$

C. Variational EM Algorithm

Thus far, we have considered the Laplace approximation algorithm for a posterior distribution of latent variables in Gaussian process classification with multiclass. Another major objective in this research area is to estimate the hyper-parameters of the covariance function. Here, we propose an algorithm to estimate the hyper-parameters of the covariance

function in the framework of Gaussian process classification with incomplete data. One possible approach is to consider the EM-like algorithm that is widely used with the incomplete data.

During the E-step of the EM-like algorithm, we drive the Gaussian approximation posterior $q(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta)$ for latent function values $\mathbf{f}$ using Laplace approximation. In the M-step of the EM-like algorithm, we seek a hyper-parameter Θ that can maximize a lower bound on a logarithm of the marginal likelihood $q(\mathbf{Y} \mid \mathbf{X}, \Theta)$ using the approximate posterior $q(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta)$ obtained during the E-step. The E and M-steps are iteratively repeated until a convergence condition satisfies. We describe our EM algorithm in detail as follows.

E-step 1: Assume that initial values Θ_0 for hyper-parameters Θ are given. Using the Laplace approximation, the true posterior $p(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta)$ of latent function $\mathbf{f}$ is approximated as a Gaussian posterior $q(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta)$ such as in the following:

$$q(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta) \sim \mathcal{N}(\mathbf{m}_{\mathbf{F}}, \mathbf{V} = (\mathbf{K}^{-1} + \mathbf{W})^{-1}) \tag{22}$$

M-step 1: With a Gaussian posterior $q(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta)$ held fixed, we seek a new value Θ^{new} so that the lower bound $F(q, \Theta)$ given in the following (23) can be maximized with respect to Θ:

$$\begin{aligned} \ln p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\Theta}) &= \ln \int p(\mathbf{f} \mid \mathbf{X}, \Theta)\, p(\mathbf{Y} \mid \mathbf{f})\, d\mathbf{f} \\ &= \int q(\mathbf{f}) \ln\left(\frac{p(\mathbf{f}, \mathbf{Y} \mid \mathbf{X}, \Theta)}{q(\mathbf{f})}\right) d\mathbf{f} + \int q(\mathbf{f}) \ln\left(\frac{q(\mathbf{f})}{p(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)}\right) d\mathbf{f} \\ &\geq F(q, \Theta) = \int q(\mathbf{f}) \ln\left(\frac{p(\mathbf{f}, \mathbf{Y} \mid \mathbf{X}, \Theta)}{q(\mathbf{f})}\right) d\mathbf{f} \end{aligned} \tag{23}$$

Here, the low bound $F(q, \Theta)$ can be written as

$$\begin{aligned} F(q, \Theta) &= \int q(\mathbf{f}) \ln\left(\frac{p(\mathbf{f} \mid \mathbf{X}, \Theta)\, p(\mathbf{Y} \mid \mathbf{f})}{q(\mathbf{f})}\right) d\mathbf{f} \\ &= \int q(\mathbf{f}) \ln p(\mathbf{f} \mid \mathbf{X}, \Theta) d\mathbf{f} + \int q(\mathbf{f}) \ln p(\mathbf{Y} \mid \mathbf{f}) d\mathbf{f} \\ &\quad - \int q(\mathbf{f}) \ln q(\mathbf{f}) d\mathbf{f} \\ &= E_{q(\mathbf{f})}(\ln p(\mathbf{f} \mid \mathbf{X}, \Theta)) + E_{q(\mathbf{f})}(\ln p(\mathbf{Y} \mid \mathbf{f})) + \mathrm{H}(q(\mathbf{f})). \end{aligned} \tag{24}$$

Moreover, because the second and third terms are independent with hyper-parameters Θ, we must only maximize the first term $E_{q(\mathbf{f})}(\ln p(\mathbf{f} \mid \mathbf{X}, \Theta))$ with respect to Θ. By computing $E_{q(\mathbf{f})}(\ln p(\mathbf{f} \mid \mathbf{X}, \Theta))$ using a Gaussian approximate posterior, the first term can be given as:

$$\begin{aligned} &E_{q(\mathbf{f})}(\ln p(\mathbf{f} \mid \mathbf{X}, \Theta)) \\ &= -\frac{nc}{2} \ln 2\pi - \frac{1}{2} \ln |\mathbf{K}(\Theta)| - \frac{1}{2} E_{q(\mathbf{f})}\left(\mathbf{f}^T \mathbf{K}(\Theta)^{-1} \mathbf{f}\right) \\ &= -\frac{nc}{2} \ln 2\pi - \frac{1}{2} \ln |\mathbf{K}(\Theta)| - \frac{1}{2}\left(E_{q(\mathbf{f})}(\mathbf{f})^T \mathbf{K}(\Theta)^{-1} E_{q(\mathbf{f})}(\mathbf{f})\right) \\ &\quad - \frac{1}{2} tr\left(\mathbf{K}(\Theta)^{-1} Cov(\mathbf{f})\right) \end{aligned} \tag{25}$$

Here, by differentiating $E_{q(\mathbf{f})}(\ln p(\mathbf{f} \mid \mathbf{X}, \Theta))$ with respect to Θ using the E-step result, we obtain

$$\begin{aligned} \frac{\partial E_{q(\mathbf{f})}(\ln p(\mathbf{f} \mid \mathbf{X}, \Theta))}{\partial \Theta} &= -\frac{1}{2} \mathrm{tr}\left(\mathbf{K}(\Theta)^{-1} \frac{\partial \mathbf{K}(\Theta)}{\partial \Theta}\right) \\ &\quad + \frac{1}{2}\left(\mathbf{m}_{\mathbf{f}}{}^T \mathbf{K}(\Theta)^{-1} \frac{\mathbf{K}(\Theta)}{\partial \Theta} \mathbf{K}(\Theta)^{-1} \mathbf{m}_{\mathbf{f}}\right) \\ &\quad + \frac{1}{2} tr\left(\mathbf{K}(\Theta)^{-1} \frac{\mathbf{K}(\Theta)}{\partial \Theta} \mathbf{K}(\Theta)^{-1} Cov(\mathbf{f})\right). \end{aligned} \tag{26}$$

Therefore, we can obtain the hyper-parameter maximizing the free energy by the following gradient update rule:

$$\Theta^{new} = \Theta^{old} + \eta \left(\frac{\partial E_{q(\mathbf{f})}(\ln p(\mathbf{f} \mid \mathbf{X}, \Theta))}{\partial \Theta}\right)_{\Theta = \Theta^{old}} \tag{27}$$

D.Prediction Method

Here, if we denote a vector $\mathbf{f}_*$ as the latent function values that correspond to a test point $\mathbf{x}_*$, then the joint prior distribution of the training latent function $\mathbf{f}$ and test latent function $\mathbf{f}_*$ is

$$p(\mathbf{f}, \mathbf{f}_* \mid \mathbf{x}_*, \mathbf{X}, \Theta) \sim G\left(\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \mid \mathbf{0}, \begin{bmatrix} \mathbf{K} & \mathbf{k}_* \\ \mathbf{k}_*^T & k_{**} \end{bmatrix}\right), \tag{28}$$

where $\mathbf{k}_* = diag(\mathbf{k}_*^1(\mathbf{x}, \mathbf{x}_*), \cdots, \mathbf{k}_*^C(\mathbf{x}, \mathbf{x}_*))$ is a $(nc \times c)$ matrix obtained by stacking vertically the vectors $\mathbf{k}_*^c(\mathbf{x}, \mathbf{x}_*) = (k^c(\mathbf{x}_1, \mathbf{x}_*), \cdots, k^c(\mathbf{x}_n, \mathbf{x}_*))^T, c = 1, \cdots, C$ and $\mathbf{k}_{**} = \mathrm{diag}(k^1(\mathbf{x}_*, \mathbf{x}_*), .., k^C(\mathbf{x}_*, \mathbf{x}_*))$. Hence, given a test point $\mathbf{x}_*$, the posterior distribution of latent function $\mathbf{f}_*$ corresponding to test point $\mathbf{x}_*$ is given by marginalizing over the training set latent functions $\mathbf{f}$:

$$\begin{aligned} p(\mathbf{f}_* \mid \mathbf{x}_*, \mathbf{Y}, \mathbf{X}, \Theta) &= \int p(\mathbf{f}_*, \mathbf{f} \mid \mathbf{x}_*, \mathbf{Y}, \mathbf{X}, \Theta)\, \mathrm{d}\mathbf{f} \\ &= \int p(\mathbf{f}_* \mid \mathbf{f}, \mathbf{x}_*, \mathbf{X}, \Theta)\, p(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)\, \mathrm{d}\mathbf{f} \end{aligned}. \tag{29}$$

where the conditional prior is given by

$$p(\mathbf{f}_* \mid \mathbf{f}, \mathbf{x}_*, \mathbf{X}, \Theta) \sim \mathrm{G}(\mathbf{f}_* \mid \mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{f}, \mathbf{k}_{**} - \mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{k}_*) \tag{30}$$

Moreover, by using the Gaussian approximation $q(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta)$ to the true posterior $p(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta)$, we have finally obtained the approximate posterior distribution of latent function $\mathbf{f}_*$:

$$q(\mathbf{f}_* \mid \mathbf{x}_*, \mathbf{Y}, \mathbf{X}, \Theta) \sim G(\mathbf{f}_* \mid \mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{m}, \mathbf{k}_{**} - \mathbf{k}_*^T (\mathbf{K} + \mathbf{W}^{-1})^{-1} \mathbf{k}_*). \quad (31)$$

In addition, the predictive distribution of class membership vector $\mathbf{y}_*$ is obtained by integrating out $\mathbf{f}_*$,

$$q(\mathbf{y}_* \mid \mathbf{x}_*, \mathbf{Y}, \mathbf{X}, \Theta) = \int p(\mathbf{y}_* \mid \mathbf{f}_*) q(\mathbf{f}_* \mid \mathbf{x}_*, \mathbf{Y}, \mathbf{X}, \Theta) \, \mathrm{d}\mathbf{f}_*. \quad (32)$$

Hence, the predictive mean vector for class c is given by

$$\mathrm{E}_q(\mathrm{f}_*^c \mid \mathbf{X}, \mathbf{Y}, \boldsymbol{\theta}, \mathbf{X}_*) = \mathbf{k}_*^{cT} (\mathbf{K}^c)^{-1} \mathbf{m}_\mathbf{F}^c = \mathbf{k}_*^{cT} (\mathbf{y}^c - \boldsymbol{\pi}^c) \quad (33)$$

Moreover, if these are inserted into a vector form, then the expectation of latent function $\mathbf{f}_*$ under the Laplacian approximation is given as

$$\boldsymbol{\mu}_* = \mathrm{E}_q(\mathbf{f}_* \mid \mathbf{X}, \mathbf{Y}, \boldsymbol{\theta}, X_*) = \mathbf{Q}_*^T (\mathbf{y} - \boldsymbol{\pi}) \quad (34)$$

where a matrix $\mathbf{Q}_*$ is defined as the $(nC \times C)$ matrix:

$$\mathbf{Q}_* = \begin{pmatrix} \mathbf{k}_*^1(\mathbf{x}, \mathbf{x}_*) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{k}_*^2(\mathbf{x}, \mathbf{x}_*) & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{k}_*^C(\mathbf{x}, \mathbf{x}_*) \end{pmatrix} \quad (35)$$

In addition, the covariance matrix of the latent function $\mathbf{f}_*$ can be represented as

$$\boldsymbol{\Sigma}_* = Cov_q(\mathbf{f}_* \mid \mathbf{x}_*, \mathbf{X}, \mathbf{Y}, \Theta) = \mathbf{k}_{**} - \mathbf{Q}_*^T (\mathbf{K} + \mathbf{W}^{-1})^{-1} \mathbf{Q}_* \quad (36)$$

III. Human Action Classification

We employ two steps to classify a human action video. We first extract a sequence of a multidimensional feature vector from each video from a given training data and we derived a posterior distribution of latent function using extracted feature vectors. To classify the input human action video, we then extract a test feature vector from a query video, and derive a posterior of latent function that corresponds to the test feature vector. Next, by computing the posterior probabilities that a test video belongs to every class when this posterior is used, we can then classify a query video with a class that maximizes a posterior probability.

A. Training Step

In the training step, an algorithm that computes the Gaussian approximation posterior distribution of the latent function using a given set of training data is described as follows:

E-step:

1. Extract n time series of multidimensional feature vectors $\Omega = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n)$ from a training video dataset that describes several human actions.
2. Assume the prior distribution of the latent function $\mathbf{f}(\mathbf{x})$ as a Gaussian process model, and compute the covariance matrix $\mathbf{K}$ by using the extracted feature vectors.
3. By combining the likelihood function of target vector $\mathbf{Y} = (\mathbf{Y}_1, \cdots, \mathbf{Y}_n)$ and a prior of latent function $\mathbf{f}(\mathbf{x})$, we obtain a logarithm of true posterior distribution of $\mathbf{f}(\mathbf{x})$ as

$$\ln p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}, \Theta) \approx \ln p(\mathbf{y} \mid \mathbf{f}) + \ln p(\mathbf{f} \mid \mathbf{X}, \Theta) \quad (37)$$

4. Through Laplace approximation, we drive an approximation posterior distribution of the latent function $\mathbf{f}(\mathbf{x})$ as

$$q(\mathbf{f} \mid \mathbf{Y}, \mathbf{X}, \Theta) \sim N(\mathbf{m}_\mathbf{f}, (\mathbf{K}^{-1} + \mathbf{W})^{-1}) \quad (38)$$

M-step:

1. Using the approximate Gaussian posterior $q(\mathbf{f} \mid \mathbf{X}, \mathbf{Y}, \Theta)$, we compute the lower bound $F(q, \Theta)$ for a log marginal likelihood $\ln p(\mathbf{Y} \mid \mathbf{X}, \Theta)$.
2. Using the gradient update rule, we derive the hyper-parameter Θ that maximizes the free energy $F(q, \Theta)$.

B. Classification Step

During the classification step, we classify the new input video by using the following algorithms.

1. Extract feature vector $\mathbf{x}_*$ from the new input video, and compute two covariance matrix $\mathbf{K}_*$ and $\mathbf{k}_{**}$ between a new data $\mathbf{x}_*$ and the existing training data $\mathbf{X}$.
2. Compute the predicted posterior distribution of a new latent function $\mathbf{f}_*$:

$$q(\mathbf{f}_* \mid \mathbf{x}_*, \mathbf{Y}, \mathbf{X}, \Theta) \sim G(\mathbf{f}_* \mid \mathbf{k}_*^T \mathbf{K}^{-1} \mathbf{m}, \mathbf{k}_{**} - \mathbf{k}_*^T (\mathbf{K} + \mathbf{W}^{-1})^{-1} \mathbf{k}_*) \quad (39)$$

3. Extract the n samples $\mathbf{f}_{*1}, \cdots, \mathbf{f}_{*\mathrm{n}}$ from the Gaussian posterior distribution of latent function $\mathbf{f}_*$ with mean vector $\boldsymbol{\mu}_*$ and covariance matrix $\boldsymbol{\Sigma}_*$.
4. Calculate the probability of classification $\boldsymbol{\pi}_{*1}, \cdots, \boldsymbol{\pi}_{*\mathrm{n}}$ using the extracted random sample $\mathbf{f}_{*1}, \cdots, \mathbf{f}_{*\mathrm{n}}$, and compute the average $\overline{\boldsymbol{\pi}}_* = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\pi}_{*i}$ of these probabilities.

5. Classify the input human action video into a class c^* with the maximum classification probability as:

$$c^* = \text{argmax}_{1\le c\le C}\, \bar{\pi}_*^c .$$

IV. Experiment Results

A. KTH Dataset

The KTH human action dataset used in our experiment included 25 people performing six action classes, namely: walking, running, jogging, boxing, hand waving, and hand clapping. Each video sequence contained one actor performing an action. All video sequences were resized to $64\times64\times32$. To standardize to a length of 64 frames, we retained the middle 64 frames, and recycled frames for video containing fewer than 64 frames. Several types of data sets can be used to recognize human behavior. KTH data set and Weizman data sets are the most commonly used. In our experiments, we employed the KTH data set. Our KTH human action dataset consisted of 600 video sets. All videos were shot in black and white with a 160 160×120 resolution. In this study, we use the training data consisting of ten peoples performing six actions.

Defining the feature vector to classify exactly human behavior is difficult. In this study, we used the histograms oriented of gradients (HOG) features that are mostly used in pedestrian detection to express the behavioral changes of each video sequence into a feature vector. To calculate the HOG feature vector, we divide each frame into 3 by 5 blocks, and used eight histogram bins from each block. Therefore, the total dimension of the feature vector for each frame is 120. In addition, the calculated HOG feature vector was normalized using:

$$\bar{F} = \frac{F}{\sqrt{\|F\|^2 + 1}} \tag{40}$$

B. Performance of Proposed Method

To evaluate the performance of the proposed method, we used 150 video sequences consisting of 25 peoples performing six human behaviors as validation data. Table I presents a matrix form showing classification rates of six human behaviors in relation to the proposed method. The results from Table I, indicate that the six human behaviors can be divided into two categories of similar behavior. The first category of similar actions includes boxing, hand-clapping, and hand-waving, whereas the second category includes jogging, running, and walking. Here, we can see that a misclassification occurred often with actions belonging to the same category. Consequently, we note that a correct classification rate of 85.33% occurs when the proposed method is employed.

Table II shows the results of the recognition rate for the proposed method compared to conventional methods when using the KTH data set. Conventional methods included the local spatio-temporal characteristics or HOG / HOF feature vector. SVM or GMM were used as the classification method. However, the proposed method employed a feature vector representing an image that was universally altered.

TABLE I
Classification Rates of Actions for the Proposed Method

Classification rate	Boxing	Hand clapping	Hand waving	Jogging	Running	Walking
Boxing	1	0	0	0	0	0
Hand-clapping	0.12	0.88	0	0	0	0
Hand-waving	0.12	0.08	0.8	0	0	0
Jogging	0	0	0	0.8	0.2	0
Running	0	0	0	0.08	0.92	0
Walking	0	0	0	0.12	0.16	0.72

TABLE II
Comparison of Proposed and Existing Method

Methods	Correct classification rate
Laptev.et al. [10]	91.80%
Mikolajczyk et al. [11]	93.20%
Yuan et al. [12]	93.30%
Kaaniche et al. [13]	90.57%
Kovashka et al. [14]	94.53%
Yin et al. [15]	82%
Proposed method	85.33%

V. Conclusion

This study propose an inference algorithm that can drive simultaneously both a posterior distribution of a latent function and estimators of hyper-parameters in the Gaussian process classification model. The proposed algorithm is performed in two steps: called expectation step and maximization steps. First, during the expectation step, we use the Bayesian formula and Laplace approximation to derive approximately the posterior distribution of the latent function based on the learning data. Furthermore, we considered a method of calculating a mean vector and covariance matrix of the latent function. Second, during the classification step, we use a derived posterior distribution of the latent function to derive the maximum likelihood estimator for hyper-parameters necessary to define a covariance matrix.

In our experiments, we applied the proposed algorithm to a human action classification problem using a public database, namely, the KTH human action data set. Experimental results showed that our method performs extremely well with public video dataset and thus better than others method.

Our future work will extend the proposed method to other video recognition problems such as 3D human action recognition, gesture recognition, and those associated with surveillances system.

References

[1] C. E. Rasmussen, and C. K. I. Williams, "*Gaussian Processes for Machine Learning*," MIT Press, 2006.
[2] H. Nicklisch, and C. E. Rasmussen, "Approximation for Binary Gaussian process Classification," JMLR, 2008, pp. 2035-75.
[3] A. B. Chan, and D. Dong, "Generalized Gaussian process model," IEEE Conf. on Computer Vision and Pattern Recognition, Colorado Spring, 2011.

[4] A. C. Chan, "Multivariate generalized Gaussian process models," eprint arXiv: 1311.0360, 2013.
[5] H. Kim, and Z. Ghahramani, "Bayesian Gaussian Process Classification with the EM-EP algorithm," IEEE Trans. on PAMI, vol. 28, no. 12, pp 1948-1959, 2006.
[6] C. E. Rasmussen, and H. Nickisch, The GPML Toolbox version 3.4, gaussianprocess.org.
[7] L. Raskin, E. Rivlin, and M. Rudzsky, "Using Gaussian Processes for Human tracking and Action Classification", ISVC 2007, Part 1, LNCS 4841, pp 36-45, 2007.
[8] H. Zhou, L. Wang, D. Sutter, "Human action recognition by fearture-reduced Gaussian process classification", Pattern Recognition Letters, v0l. 30, pp 1059-1065, 2009.
[9] Q. Zhao, L. Zhang, A. Cjchocki, "A Tensor-Variate Gaussian Process for Classification of Multidinensional Structured Data", Proceeding of the Twenty-Seventh AAAI Conference on Artificial Intelligence, pp 1041-1047, 2013.
[10] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in CVPR 2008.
[11] K. Mikolajczyk and H. Uemura. "Action recognition with motion-appearance vocabulary forest," CVPR, 2008.
[12] J. Yuan, Z. Liu, and Y. Wu, "Discriminative Subvolume Search for Efficient Action Detection," CVPR, 2009.
[13] M. B. Kaaniche and F. Bremond, "Gesture Recognition by Learning Local Motion Signatures," In CVPR, 2010.
[14] A. Kovashka and K. Grauman, "Learning a Hierarchy of Discriminative Space-Time Neighborhood Features for Human Action Recognition," In CVPR, 2010.
[15] J. Yin and Y. Meng, "Human Activity Recognition in Video using a Hierarchical Probabilistic Latent Model," In CVPR, 2010.

Wan-Hyun Cho received both B.S. degree and M.S. degree from the Department of Mathematics, Chonnam National University, Korea in 1977 and 1981, respectively and Ph.D. degree from the Department of Statistics, Korea University, Korea in 1988. He is now teaching in Chonnam National University. His research interests are statistical modeling, pattern recognition, image processing, and medical image processing.

Soon-Ja Kang received both B.S. degree and M.S. degree from the Department of Mathematics, Chonnam National University, Korea in 1979 and 1981, respectively and Ph.D. degree from the Department of Mathematics, Seogang University, Korea in 1988. She is now teaching in Chonnam National University. Her research fields are mathematical education, advanced calculus, and education for the gifted children.

Sang-Kyoon Kim received the B.S., M.S. and Ph.D. degrees in Electronics Engineering, Mokpo National University, Korea in 1998, 2000 and 2015 respectively. From 2011 to 2015, he was a Visiting Professor in the Department of Information & Electronics Engineering, Mokpo National University, Korea. His research interests include image processing, pattern recognition and computer vision.

Soon-Young Park received B.S. degree in Electronics Engineering from Yonsei University, Korea in 1982 and M.S and Ph.D. degrees in Electrical and Computer Engineering from State University of New York at Buffalo, in 1986 and 1989, respectively. From 1989 to 1990 he was a Postdoctoral Research Fellow in the department of Electrical and Computer Engineering at the State University of New York at Buffalo. Since 1990, he has been a Professor with Department of Electronics Engineering, Mokpo National University, Korea. His research interests include image and video processing, image protection and authentication and image retrieval techniques.