

Received April 6, 2017, accepted April 18, 2017, date of publication April 25, 2017, date of current version June 7, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2697408

Non-Linear Dimensionality Reduction and Gaussian Process Based Classification Method for Smoke Detection

FEINIU YUAN¹, (Senior Member, IEEE), XUE XIA¹, JINTING SHI^{1,2}, HONGDI LI¹, AND GANG LI¹

¹School of Information Technology, Jiangxi University of Finance and Economics, Nanchang 330032, China

²Vocational School of Teachers and Technology, Jiangxi Agricultural University, Nanchang 330045, China

Corresponding authors: Feiniu Yuan (yfn@ustc.edu); Xue Xia (375661427@qq.com)

This work was supported in part by the Natural Science Foundation of China under Grant 61363038, in part by the Cultivated Talent Program for Young Scientists of Jiangxi Province under Grant 20142BCB23014, in part by the Science Technology Application Project of Jiangxi Province under Grant KJLD12066 and Grant GJJ150459, in part by the Key Technology Research and Development Program of Jiangxi Province under Grant 2015ZBBE50013, and in part by the Knowledge Innovation Fund for the Graduate Students of Jiangxi Province under Grant YC2016-B064.

ABSTRACT To improve smoke detection accuracy, we combine local binary pattern (LBP) like features, kernel principal component analysis (KPCA), and Gaussian process regression (GPR) to propose a novel data processing pipeline for smoke detection. The data processing pipeline consists of three steps including original feature extraction, dimensionality reduction, and classification. We use LBP-like methods to extract original features. To obtain a more discriminant feature, KPCA is used to non-linearly map the original features into a discriminant feature space, where manifold structures are embedded. Finally, in order to improve generalization performance, we apply GPR to model classification as a Gaussian process by imposing Gaussian priors on both data and hyper-parameters. In addition, we can replace any steps of the pipeline by similar methods for further improvement or exploration, so the pipeline is flexible and extensible. Experimental results show that KPCA and GPR are truly able to improve the performance of smoke detection and texture classification, and our method obviously outperforms the same features with Support Vector Machine (SVM).

INDEX TERMS Smoke detection, kernel principal component analysis (KPCA), gaussian process regression (GPR), classification pipeline.

I. INTRODUCTION

Smoke provides important clues for early fire detection. Traditional fire detectors, such as ionized sensors, require to be installed closely to fire or smoke. Thus traditional sensor based fire detection techniques are always applied indoors. Since smoke often occurs before flame does in many fires, detecting smoke from images is more suitable than detecting flames for outdoor fires, such as forest fires. Image based smoke detection has two basic tasks, which are recognition of smoke and localization of smoke, respectively. Apparently, the key factor to implement smoke detection from images is to recognize smoke from a sub-image. In addition, unlike objects, such as faces or man-made buildings, the variation of smoke shapes, color, density and texture are largely different from each other, so it is very difficult to extract robust features from images to represent smoke.

Statistical measures are widely used to detect smoke in videos or images. For example, mean and standard

deviation, are calculated to determine the presence of smoke in [1]. Gubbi *et al.* [2] used arithmetic mean, geometric mean, standard deviation, skewness, kurtosis, and entropy to detect smoke from videos. Ferrari *et al.* [3] constructed a Hidden Markov Tree Model with wavelet transform to detect steam. Yuan *et al.* proposed a fast accumulative motion orientation model based on integral image [4], and shape-invariant features on multi-scale partitions with AdaBoost [5], [6] for smoke detection. Local Binary Patterns (LBP) [7] encode differences between local regions and describes the texture information well. LBP features have been widely used in face recognition, texture classification and smoke detection because of its discriminative capability, computational efficiency, and low illumination sensitivity. To improve discriminative capabilities of LBP, many LBP variants have been developed. Yuan *et al.* [8] used Hamming distances to acquire spatial relationships between LBP code pairs according to gradients of LBP codes.

Qian *et al.* [9] proposed Pyramid based Local Binary Patterns (PLBP) to suppress the influence of noise. Yuan [10] used scale space to propose high order Derivative Local Binary Patterns based on Circular shift sub-uniform and Scale space (DLBPCS) and adopted the histograms of DLBPCS to detect smoke.

LBP-like features with SVM have performed well in texture classification. Although smoke can be viewed as a specific kind of texture, LBP features are not good enough for smoke detection. The main reason is that the texture in smoke is not as clear as that in other objects, such as leaves or rocks. Gaussian Process Regression (GPR) is based on a non-parametric Bayesian methodology, which models classification as a Gaussian Process. In theory, GPR provides a satisfying and smoothing fit for observed data of complicated distribution. The smoothness is achieved by imposing Gaussian priors over both data and hyper-parameters. Bayesian methodology methods are able to deal with high-dimensional or large-scale data without the problems of overfitting [11]. The hyper-parameters of GPR based methods can be learned automatically from data instead of manually setting.

In this paper, we combine LBP, Kernel Principal Component Analysis (KPCA) and GPR to propose a novel data processing pipeline for smoke detection. The pipeline aims at improving detection rates and reducing false alarm rates at the same time by non-linearly mapping data to feature spaces and involving Gaussian priors to capture intrinsic structures of features. LBP or its variants are used to extract original features. KPCA is adopted to discover manifold structures of original features. GPR is used to model classification as a Gaussian Process, which takes the joint distribution of features to improve generalization. The data processing pipeline consists of three steps that are original feature extraction, dimensionality reduction and classification. The pipeline is flexible and extensible, so we can replace any steps of the pipeline by other similar methods for further improvement.

This paper has at least two main contributions. First, we combine LBP variants, KPCA and GPR together to perform classification of smoke. LBP-like features are non-linearly mapped to discriminant features in a low-dimensional manifold. Classification of the mapped features is modelled as a Gaussian Process, thus we can maximize generalization of classification method by imposing Gaussian priors on both data and hyper-parameters. Second, we propose a novel data processing pipeline to provide a generalized framework for smoke detection and texture classification. Each processing step of the pipeline is responsible for different task, and it can be easily replaced by similar methods for performance improvement. To our knowledge, KPCA and GPR methods have not been reported in smoke detection.

II. RELATED WORK

Wen *et al.* [12] proposed a difference vector plus KPCA method to optimize the features for face detection. Lei *et al.* [13] used Kernel Principal Component Analysis

(KPCA) to improve the discriminative power of features for 3D face recognition. Xu *et al.* [14] applied KPCA in patch-based texture descriptors to obtain compact and discriminative texture features with Gaussian-like noise characteristics for sea ice segmentation, which belonged to texture classification. Zhou *et al.* [15] proposed a single-image super-resolution method in which KPCA was adopted to learn the dictionary. Savic *et al.* [16] proposed a range estimation method based on KPCA to address problems occurred in ultra-wideband (UWB) technology, which provided the most accurate range estimates. Zhang *et al.* [17] reduced the feature dimensionality with KPCA in the proposed second-order local ternary pattern for median filtering detection.

Cheng *et al.* [18] proposed a video anomaly detection and localization method via Gaussian Process Regression, which was mainly used for modeling the interaction templates codebook and for computing likelihood. Liu *et al.* [19] proposed an L1 construction and a local approximation covariance weight updating method to improve the Gaussian Processes based realistic action recognition framework. Zhu *et al.* [20] automatically estimated age from facial images by proposing the orthogonal Gaussian Process. Challis *et al.* [21] applied Bayesian Gaussian Process Logistic Regression for disease classification. Lee *et al.* [22] used Gaussian Process Regression trees for face alignment. Dhall *et al.* [23] proposed an expression intensity estimation method based on Gaussian process regression. Markov *et al.* [24] investigated the feasibility and applicability of GP models for music genre classification and music emotion estimation. Fink *et al.* [25] used spatial Gaussian processes in the estimation of wireless propagation environment to identify viable point-to-point communication links. Long *et al.* [26] applied the Gaussian process model to multi-class visual recognition together with active learning, in which a generalized EM-EP algorithm was derived to estimate the parameters and approximate Bayesian inference. The multi-class classification was achieved by one-versus-all binary classification [26].

Kernel Linear Discriminant Analysis (KLDA) and Gaussian Processes (GP) were combined as a discriminant form of Gaussian Process Latent Variable Model (GPLVM) for classification [27]. The method in [27] was known as DGPLVM and applied to face recognition [28]. And the authors claimed that it surpassed human-level recognition [28]. However, LDA projects features to a space of dimension at most $c - 1$, where c is the number of classes. For a binary classification in smoke detection, the projected space is one-dimensional. As an alternative, we adopt unsupervised KPCA to retain more information and simplify feature representation. Besides, GP based model was also combined with Probabilistic PCA to model high dimensional data [29]. PCA and KPCA are adopted mainly as a dimensionality reduction method. To our knowledge, combination of KPCA and GPR has not been proposed in smoke detection. Therefore, we propose to combine KPCA with GPR to improve the performance of smoke detection.

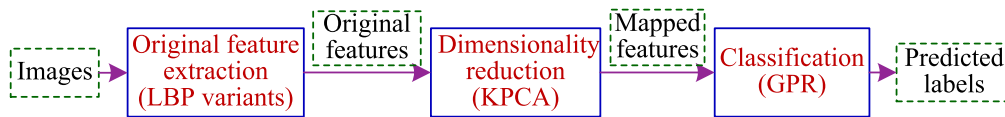


FIGURE 1. The overall flowchart of the data processing pipeline.

III. KERNEL PRINCIPAL COMPONENT ANALYSIS AND GAUSSIAN PROCESS REGRESSION

The overall flowchart of the data processing pipeline is shown in Fig. 1. The rectangles in solid lines represent approaches that are involved in our method, while those in dashed lines are the inputs and outputs in every step.

As shown in Fig. 1, the overall flowchart of the pipeline mainly includes three steps that are original feature extraction, dimensionality reduction, and classification. We use LBP variants to extract original features from input images. Then Kernel Principal Component Analysis (KPCA) is adopted for dimensionality reduction to obtain compact representations of images, which are known as mapped features. At last, the mapped features are used as final features and are sent to Gaussian Process Regression (GPR) for classification to output predicted labels for the images. Below we will describe each step in details.

A. LBP FEATURES

The original LBP code [7] is obtained by comparing the value of a center pixel g_c with the values of its neighborhood pixels g_i ($i = 0, 1, \dots, 7$), as shown in the two equations below.

$$\text{LBP} = \sum_{i=0}^{P-1} s(g_i - g_c)2^i \quad (1)$$

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2)$$

where P points are re-sampled in a circular neighborhood with radius R around the center pixel g_c . As an alternative, there are three mapping patterns in original LBP, which are ‘‘Uniform’’ (U2), ‘‘Rotation Invariant’’ (RI) and ‘‘Rotation Invariant and Uniform’’ (RIU2).

The most frequently used parameters are $P = 8$ and $R = 1$, thus the dimensions of $\text{LBP}_{P,R}^{\text{U2}}$, $\text{LBP}_{P,R}^{\text{RI}}$ and $\text{LBP}_{P,R}^{\text{RIU2}}$ are 59, 36, and 10.

For the sake of simplicity, we just use histograms of LBP codes extracted by existing methods as the original features for Kernel Principal Component Analysis (KPCA). In the future, we will focus on research of specific feature extraction methods, which are even not limited to LBP-like features, in order to generate more stable, compact and robust representations for images.

For the i -th gray-scale training image $f_i(x, y)$, we compute the histogram of original LBP codes as follows:

$$x_i(k) = \sum_{y=0}^{h-1} \sum_{x=0}^{w-1} \delta(\text{LBP}(x, y) - k) \quad (3)$$

where $\delta(v)$ is the delta function that returns 1 if $v = 0$ and 0 for $v \neq 0$, $x_i(k)$ stands for the k -th bin of the histogram, w and h are the width and height of the image $f_i(x, y)$, and $\text{LBP}(x, y)$ denotes the LBP code map of the image $f_i(x, y)$.

B. KERNEL PRINCIPAL COMPONENT ANALYSIS (KPCA)

Principal Component Analysis (PCA) is defined as an orthogonal linear transformation, which is often used to transform the data onto a new sub-space. The transformation matrix is formed by the top- d eigenvectors of covariance matrix. $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ stands for the input data with zero mean. The i -th sample is denoted by \mathbf{x}_i , while the mean vector of all the samples \mathbf{x}_i ($i = 1, \dots, N$) is denoted by \mathbf{x}^C . The optimized projection matrix is obtained by minimizing the following goal function:

$$\mathbf{W} = \arg \min_{\mathbf{W}} \left\{ \text{tr}(\mathbf{W}^T \mathbf{C} \mathbf{W}) - \lambda(\mathbf{I} - \mathbf{W}^T \mathbf{W}) \right\} \quad (4)$$

Where $\text{tr}(\mathbf{A})$ denotes the trace of a matrix \mathbf{A} . The covariance matrix \mathbf{C} is defined as follows.

$$\mathbf{C} = \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \mathbf{x}^C)(\mathbf{x}_i - \mathbf{x}^C)^T \quad (5)$$

By solving the above equations, \mathbf{W} is just formed by the first- d eigenvectors, sorted in descending order by eigenvalues, of the covariance matrix \mathbf{C} , i.e., $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_d]$. As the dimension is reduced, some information is lost but not too much because the discarded eigenvalues are small [30]. For a given sample \mathbf{x} , the projected data \mathbf{y} is computed as follows:

$$\mathbf{y} = \mathbf{W}^T \mathbf{x} \quad (6)$$

The dimension of \mathbf{y} is often less than \mathbf{x} .

Kernel Principal Component Analysis (KPCA) was proposed by [31] to extend the original linear PCA to non-linear data distributions [32]. Before applying a PCA, the input data should be nonlinearly mapped into a high dimensional feature space [33], [34]. The mapping function is defined as:

$$\phi : \mathbf{x} \rightarrow \phi(\mathbf{x}) \quad (7)$$

Generally, the mapping function $\phi(\mathbf{x})$ is unknown. A kernel function $k(\mathbf{x}_i, \mathbf{y}_j)$ is used to avoid explicitly specifying the unknown mapping function. We empirically choose the Gaussian kernel in KPCA to keep consistent with the kernel function of the covariance function for simplicity of computation. The kernel function $k(\mathbf{x}_i, \mathbf{y}_j)$ used here is defined as a Radius Basis Function (RBF), which is defined in Eq. 8. Then the mapped samples are fully represented by the kernel

matrix \mathbf{K} . The i -th row and j -th column element of the kernel matrix \mathbf{K} is often calculated by a Gaussian kernel:

$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) \quad (8)$$

It is impossible to explicitly center the data in the feature space since the mapping function is unknown. The kernel trick is used to indirectly center the kernel matrix \mathbf{K} . The mean of the mapped samples $\phi(\mathbf{x}_i)$ ($i = 1, \dots, N$) in the feature space is computed by

$$\phi_0 = \frac{1}{N} \sum_{i=1}^N \phi(\mathbf{x}_i) \quad (9)$$

Then, the centered version $\phi^C(\mathbf{x}_i)$ of a mapped sample $\phi(\mathbf{x}_i)$ is equal to $\phi^C(\mathbf{x}_i) = \phi(\mathbf{x}_i) - \phi_0$. The i -th row and j -th column element of the centered kernel matrix \mathbf{K}^C is calculated as follows:

$$\begin{aligned} K_{ij}^C &= \langle \phi^C(\mathbf{x}_i), \phi^C(\mathbf{x}_j) \rangle = [\phi^C(\mathbf{x}_i)]^T \phi^C(\mathbf{x}_j) \\ &= \left[\phi(\mathbf{x}_i) - \frac{1}{N} \sum_{k=1}^N \phi(\mathbf{x}_k) \right]^T \left[\phi(\mathbf{x}_j) - \frac{1}{N} \sum_{l=1}^N \phi(\mathbf{x}_l) \right] \\ &= K_{ij} - \frac{1}{N} \sum_{l=1}^N K_{il} - \frac{1}{N} \sum_{k=1}^N K_{kj} + \frac{1}{N^2} \sum_{k=1}^N \sum_{l=1}^N K_{kl} \end{aligned} \quad (10)$$

The computation of the centered kernel matrix \mathbf{K}^C can be written in matrix forms:

$$\mathbf{K}^C = \mathbf{K} - \mathbf{1}_N \mathbf{K} - \mathbf{K} \mathbf{1}_N + \mathbf{1}_N \mathbf{K} \mathbf{1}_N \quad (11)$$

where $\mathbf{1}_N$ is an $N \times N$ matrix whose elements are all $1/N$, and N stands for the number of the input samples.

The a -th eigenvector \mathbf{w}_a of the covariance matrix \mathbf{C} can be expressed by a linear combination of $\phi^C(\mathbf{x}_i)$, so the coefficients $\alpha_1^a, \dots, \alpha_N^a$ are required [35]. In other words, the a -th eigenvector \mathbf{w}_a is conversely converted to a point α^a in the mapped sample space, which is spanned by the centered mapped samples $\phi^C(\mathbf{x}_i)$ ($i = 1, \dots, N$). The combination is defined as follows:

$$\mathbf{w}_a = \sum_{i=1}^N \alpha_i^a \phi^C(\mathbf{x}_i) \quad (12)$$

The coefficient vector α^a is just the eigenvector of the centered kernel matrix \mathbf{K}^C , which is represented as:

$$\mathbf{K}^C \alpha^a = N \lambda_a \alpha^a \quad (13)$$

The eigenvector \mathbf{w}_a has unit length, so we normalize the eigenvector as follows:

$$\alpha^a = \alpha^a / \sqrt{N \lambda_a} \quad (14)$$

Once a coefficient vector α^a is acquired, corresponding projection vector \mathbf{w}_a is uniquely determined by Eq. 12. Since the mapped data $\phi^C(\mathbf{x}_i)$ is unknown, we can not compute \mathbf{w}_a explicitly. However, we can directly project a new data \mathbf{z} onto the a -th eigenvector \mathbf{w}_a by the kernel trick. We map and center

the new data \mathbf{z} to obtain a centered feature $\phi^C(\mathbf{z}) = \phi(\mathbf{z}) - \phi_0$, then we compute the projection of the centered feature $\phi^C(\mathbf{z})$ on \mathbf{w}_a as follows:

$$\mathbf{w}^T \phi^C(\mathbf{z}) = \sum_{i=1}^N a_i^a [\phi^C(\mathbf{x}_i)]^T \phi^C(\mathbf{z}) = \sum_{i=1}^N a_i^a k^C(\mathbf{x}_i, \mathbf{z}) \quad (15)$$

The centered kernel $k^C(\mathbf{x}_i, \mathbf{z})$ between the i -th mapped training sample $\phi^C(\mathbf{x}_i)$ and the new mapped data $\phi^C(\mathbf{z})$ is computed as follows:

$$\begin{aligned} k^C(\mathbf{x}_i, \mathbf{z}) &= \langle \phi^C(\mathbf{x}_i), \phi^C(\mathbf{z}) \rangle = [\phi^C(\mathbf{x}_i)]^T \phi^C(\mathbf{z}) \\ &= \left[\phi(\mathbf{x}_i) - \frac{1}{N} \sum_{k=1}^N \phi(\mathbf{x}_k) \right]^T \left[\phi(\mathbf{z}) - \frac{1}{N} \sum_{l=1}^N \phi(\mathbf{x}_l) \right] \\ &= k(\mathbf{x}_i, \mathbf{z}) - \frac{1}{N} \sum_{l=1}^N K_{il} - \frac{1}{N} \sum_{k=1}^N k(\mathbf{x}_k, \mathbf{z}) \\ &\quad - \frac{1}{N^2} \sum_{k=1}^N \sum_{l=1}^N K_{kl} \end{aligned} \quad (16)$$

C. GAUSSIAN PROCESS REGRESSION (GPR)

The central limit theorem gives that the sum of a sufficiently large number of independent random variables will be approximately normally distributed [36]. Hence, in smoke detection, the samples and their corresponding labels can be regarded as random variables that follow the zero mean normal distribution.

Given a set of training samples $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ and corresponding labels $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$, the relationship between \mathbf{X} and \mathbf{y} is modeled as a multivariate Gaussian distribution with zero mean, covariance matrix \mathbf{K} :

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{K}) \quad (17)$$

The i -th row and j -th column element $k(\mathbf{x}_i, \mathbf{x}_j)$ of covariance matrix \mathbf{K} is usually defined as a squared exponential kernel function of any two samples $(\mathbf{x}_i, \mathbf{x}_j)$, which reflects the similarity between the two samples. Therefore, \mathbf{K} can be regarded as an $N \times N$ covariance matrix controlled by the parameter set $\theta = \{\theta_0, \theta_1, \theta_2\}$.

$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \theta_0 \exp(-\theta_1 \|\mathbf{x}_i - \mathbf{x}_j\|) + \theta_2 + \delta^2 \theta_{ij} \quad (18)$$

where δ_{ij} is a Kronecker delta which returns 1 if $i = j$ and 0 otherwise.

The joint distribution of the training samples \mathbf{X} and a new test sample \mathbf{x}_* can be modeled as the following multivariate Gaussian distribution:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}_* \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} \end{bmatrix} \right) \quad (19)$$

where \mathbf{y}_* is the predicted label of \mathbf{x}_* and follows the normal distribution. $\mathbf{K}_* = [k(\mathbf{x}_1, \mathbf{x}_*), k(\mathbf{x}_2, \mathbf{x}_*), \dots, k(\mathbf{x}_N, \mathbf{x}_*)]^T$, and $\mathbf{K}_{**} = k(\mathbf{x}_*, \mathbf{x}_*)$.

The marginal likelihood is specified as a prior that is a Gaussian function $p(\mathbf{y}|\mathbf{X}, \theta) \sim \mathcal{N}(\mathbf{0}, \mathbf{K} + \sigma^2 \mathbf{I})$, so the

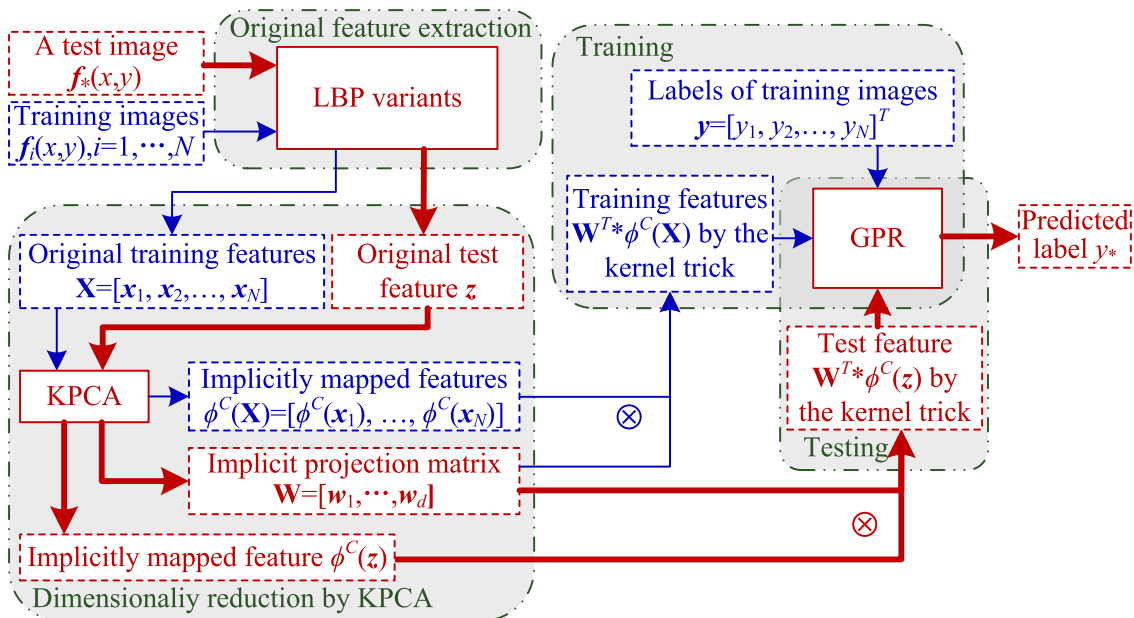


FIGURE 2. The detailed data flow procedure of the pipeline.

parameters θ can be solved by minimizing the negative log marginal likelihood defined in Eq. 20.

$$-p(\mathbf{y}|\mathbf{X}, \theta) = \frac{1}{2} \ln \det(\mathbf{K} + \sigma^2 \mathbf{I}) + \frac{1}{2} \ln \mathbf{y}^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{y} + C \quad (20)$$

Since the predictive distribution is represented by Eq. 18 and Eq. 19, the distribution of the prediction value y_* can be calculated by Eq. 21.

$$\begin{cases} p(y_* | \mathbf{x}_*, \mathbf{X}, \mathbf{y}) = \mathcal{N}(\mu_*, \sigma_*^2) \\ \sigma_*^2 = \mathbf{K}_{**} - \mathbf{K}_*^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{K}_* + \sigma^2 \\ \mu_* = \mathbf{K}_*^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{y} \end{cases} \quad (21)$$

D. THE DATA FLOW OF THE PIPELINE

The detailed data flow of the pipeline is demonstrated in Fig. 2. The pipeline has two phases including training and testing. The training phase consists of original feature extraction, dimensionality reduction by KPCA and training of GPR. The testing phase includes original feature extraction, dimensionality reduction by KPCA and testing of GPR. So there are total four steps drawn in gray rounded rectangles, which are original feature extraction, dimensionality reduction by KPCA, training and testing of GPR.

Bold red arrows stand for the flowing direction of data in the testing phase while thin blue arrows denote the data flowing direction during learning. Solid red rectangles are processing methods. Blue dashed rectangles are input and output data for training methods, while red dashed rectangles are inputs and outputs for testing methods.

In the original feature extraction step of the training phase, LBP like features are first extracted from training images $f_i(x, y)$ ($i = 1, \dots, N$) using LBP variants. The features

extracted from the training images are aggregated into a matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$, whose columns are the original features of all the training images.

Then we use unsupervised KPCA to learn an implicit mapped feature $\phi^C(x_i)$ for each image $f_i(x, y)$ and obtain an implicit projection matrix $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_d]$. In fact, we can not explicitly achieve the mapped features and the projection matrix. However, we can directly compute the projection of the mapped features, $\mathbf{W}^T * \phi^C(\mathbf{X})$, by the kernel trick of KPCA using Eq. 11 and Eq. 12.

In the last step of the training phase, we input the projected features, $\mathbf{W}^T * \phi^C(\mathbf{X})$, of all training images and the labels of training images, $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$, to the Gaussian Process Regression (GPR), and we learned a GPR classification model described by Eq. 21.

As shown in Fig. 2, the training data in the data processing pipeline flows along thin arrows from a module to another.

In the testing phase, we use the similar data flow procedure to process a test image $f_*(x, y)$ to output a predicted label y_* , as shown in Fig. 2. For the testing image $f_*(x, y)$, we first use LBP variants to extract the original LBP feature, which is denoted by a feature vector \mathbf{z} . Then we use the kernel trick of KPCA to compute the projection of the original feature \mathbf{z} in the sub-space of the mapped feature space, which is just the final mapped feature $\mathbf{W}^T * \phi^C(\mathbf{z})$. Finally, we use the GPR model to calculate the predicted label y_* for the test image $f_*(x, y)$. The data in the test phase flows along bold arrows, as shown in Fig. 2.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

To demonstrate performance of the proposed method, we compared our method with existing methods on smoke image datasets [37], which are publicly available and can

TABLE 1. Datasets for smoke detection.

Datasets	Number of positive samples	Number of negative samples	Proportion of positive samples
Set1 (1383)	552	831	39.9 %
Set2 (1505)	688	817	45.7 %
Set3 (10712)	2201	8511	20.5 %
Set4 (10617)	2254	8363	21.2 %

be downloaded via <http://staff.ustc.edu.cn/~yfn/vsd.html>. Smoke images are defined as positive samples while non-smoke images are regarded as negative samples. There are four datasets containing 1383, 1505, 10712 and 10617 images, respectively, as shown in Table 1. **Set1** with least images among all the datasets was used for training while the remaining three for testing. We also compared our method with the same features classified by LIBSVM, which was implemented by Chang *et al.* [38].

The evaluation indices, which are DR for detection rate, FAR for false alarm rate and ERR for error rate, are commonly used in smoke detection. In fact, DR and FAR are the same as true positive rate and false positive rate, respectively. ERR is equal to the ratio of the number of false positive and false negative samples divided by the total number of samples. A good classification method should achieve high DR, low FAR and ERR at the same time.

In our implementation, LBPs with RI, U2, RIU2 patterns and original version were all extracted to obtain different dimensions of normalized histograms, and Euclidean distance was adopted to measure the similarity between every two samples for computation of kernel matrices.

A. EXPERIMENTS ON SMOKE DETECTION

For fair comparisons, we set the kernel function of SVM to RBF because the squared exponential kernel was also used in KPCA and GPR. Furthermore, the parameters of SVM were carefully fine-tuned to obtain better results for smoke detection.

To evaluate the influences of KPCA, we implemented and compared two classification methods. The first one is that we directly used GPR to classify the original LBP features without KPCA for dimensionality reduction, which is denoted as GPR. The second one is that we first used KPCA to reduce the dimensions of the original LBP features to generate the lower dimensional mapped features, and then classified the mapped features using GPR. The second method is called KPCA-GPR. Also, we used different LBP pattern mapping modes to extract original features. As shown in Table 2, the features were extracted from the smoke data sets in Table 1 using original LBP methods with different mapping modes, $P = 1$ and $R = 1$. The LBP histogram dimensions with RIU2, RI, U2 and original mapping modes are 10, 36, 59 and 256, respectively.

To evaluate effectiveness of GPR, we compared GPR, KPCA-GPR, SVM with linear kernels denoted as SVM-lin and SVM with RBF denoted as SVM-RBF. The performance

of GPRs mainly depends on kernel functions. Since the kernel trick achieves obvious improvement, we introduce a kernel version of PCA, which is KPCA, to model class structures in kernel space [28]. But KPCA is not combined with SVM because SVM does not mainly count on kernel function. For fair comparisons, the four methods adopted the same features. As for SVM based methods, we adapted the parameters to balance the numbers of positive and negative samples for improvement of performance.

We list experimental results in Table 2 and highlight some of better results. Although the highest DRs on all the test sets for all kinds of LBP features occurred were obtained by SVM-lin and SVM-RBF, the lowest FARs and ERRs were achieved by GPR based methods including GPR and KPCA-GPR. In other words, SVM gets higher DRs at the expenses of higher FARs and ERRs. Therefore, GPR based methods also have the same excellent performances as SVM based methods. The original LBP features are not powerful for smoke detection, so KPCA-GPR can not demonstrate overwhelming merits over SVM-RBF. In addition, we find that the SVM-RBF obviously outperforms SVM, and KPCA-GPR outperforms GPR.

To further demonstrate the performance of our method, we also used two of the state-of-the-art methods to extract robust LBP like features for smoke detection. We first used Multichannel Decoded Local Binary Patterns (MDLBP) [39] to extract features whose dimensions are 2048. Then we adopted Pairwise Rotation Invariant Co-occurrence Local Binary Pattern (PRICoLBP) [40] to extract 1180-dimensional features. Then we used SVM-RBF and KPCA-GPR to train the features of MDLBP and PRICoLBP extracted from **Set1**. At last, we adopted SVM-RBF and KPCA-GPR to classify the features of MDLBP and PRICoLBP extracted from **Set2**, **Set3** and **Set4**. The experimental results are listed in Table 3.

DRs, FARs and ERRs in Table 3 for **Set2** are consistently lower than those in Table 2, so we cannot decide which features are better on **Set2**. However, DRs for **Set3** and **Set4** in Table 3 are higher than those in Table 2, and at the same time FARs and ERRs in Table 3 are lower than those in Table 2. Therefore, we can conclude that MDLP and PRICoLBP are more discriminative on large datasets than on small datasets.

It must be pointed out that we do not compare feature extraction methods with each other, since the objective of this paper is to validate that the processing pipeline for smoke detection is more effective than other state-of-the-art classification procedures, such as SVM based methods. In addition, we can replace the feature extraction method in the pipeline with other methods.

KPCA-GPR obviously outperformed SVM-RBF when we used robust and discriminant features for smoke detection. As shown in Table 3, KPCA-GPR with MDLBP and PRICoLBP achieved higher detection rates (DR), obviously lower false alarm rates (FAR) and error rates (ERR) than SVM-RBF with the same features. As for MDLBP on **Set4**, DR of KPCA-GPR is the same as SVM-RBF, but FAR and ERR

TABLE 2. Smoke detection with original LBP features.

Methods	Patterns	Set2			Set3			Set4		
		DR	FAR	ERR	DR	FAR	ERR	DR	FAR	ERR
SVM-lin	RIU2	98.4	6.00	3.99	97.8	8.98	7.58	97.9	8.23	6.93
	RI	98.4	6.24	4.12	98.1	8.14	6.85	98.2	7.66	6.42
	U2	97.4	4.04	3.39	96.8	5.49	5.01	97.0	4.99	4.56
	all	98.0	4.16	3.19	97.0	5.17	4.72	97.2	4.78	4.37
SVM-RBF	RIU2	98.4	6.00	3.99	97.8	8.98	7.58	97.9	8.23	6.93
	RI	98.4	6.24	4.12	98.1	8.14	6.85	98.2	7.66	6.42
	U2	97.4	4.04	3.39	96.8	5.49	5.01	97.0	4.99	4.56
	all	98.0	4.16	3.19	97.0	5.17	4.72	97.2	4.78	4.37
GPR	RIU2	98.1	4.53	3.32	96.4	5.63	5.21	96.7	5.43	4.98
	RI	98.0	4.04	3.12	96.5	4.15	4.02	96.6	3.73	3.66
	U2	97.7	2.69	2.52	94.3	4.46	4.72	94.7	4.41	4.60
	all	97.4	1.96	2.26	94.9	3.80	4.07	94.9	3.72	4.01
KPCA-GPR	RIU2	98.3	4.41	3.19	96.2	5.49	5.13	96.7	5.40	4.96
	RI	97.7	3.67	3.06	96.4	4.25	4.12	96.6	3.73	3.65
	U2	97.7	2.08	2.19	94.4	3.49	3.93	94.9	3.96	4.20
	all	97.1	1.96	2.39	95.0	3.04	3.45	95.0	3.41	3.75

TABLE 3. Smoke detection with robust LBP-like features.

Original feature types	Methods	set2			set3			set4		
		DR	FAR	ERR	DR	FAR	ERR	DR	FAR	ERR
MDLBP Dimension=2048	SVM-RBF	96.7	5.26	4.39	97.8	7.17	6.15	98.2	6.34	5.37
	KPCA-GPR	97.4	3.92	3.32	98.2	5.46	4.71	98.2	5.26	4.52
PRICoLBP Dimension=1180	SVM-RBF	97.5	2.82	2.66	97.1	4.69	4.32	96.6	3.71	3.65
	KPCA-GPR	97.8	1.96	2.06	97.2	4.56	4.20	97.0	3.41	3.32

of KPCA-GPR are obviously lower than those obtained by SVM-RBF. Therefore, our method is very useful for performance improvement especially in the case of using powerful features.

B. EXPERIMENTS ON TEXTURE CLASSIFICATION

Smoke can be regarded as a kind of texture. Actually, texture descriptors, such as LBP-like features, are used to represent smoke. To validate that our KPCA and GPR based method has powerful generalization performance and the ability of dealing with different kinds of texture features, we also tested our methods on some publicly available texture data sets. To facilitate comparisons, we used some of the texture data sets that were tested by [41], and adopted the same experimental scheme as [40] for performance evaluation. The data sets include Bordatz, Food and kth-tips.

We used High-order Local Ternary Patterns based on Magnitudes of noise removed derivatives and values of Center pixels (HLTPMC) [41] to extract original features from the texture data set of Bordatz. The data set of Bordatz consists of 999 images from 111 classes. Each class has 9 images. We randomly selected 3 images for training and used the remaining 6 images for testing. The dimension of HLTPMC is 62.

The PRICoLBP [40] features were extracted from the datasets of Food and kth-tips. The data set of Food includes

TABLE 4. Texture classification with LBP-like features.

Data sets	Original features	Methods	Accuracy
Bordatz	HLTPMC Dimension=62	GPR	95.01
		KPCA-GPR	95.07
		SVM-RBF	91.35
Food	PRICoLBP Dimension=1180	GPR	61.91
		KPCA-GPR	61.93
		SVM-RBF	61.48
kth-tips	PRICoLBP Dimension=1180	GPR	93.52
		KPCA-GPR	94.02
		SVM-RBF	94.22

61 classes and every class has 6 images. We randomly used 3 images for training and the remainder 3 images for testing. There are 10 classes on the data set of kth-tips. Every class consists of 81 images. 30 images were used at random for training and other 51 images were used for testing.

Every comparison experiment was repeated for 100 times and the averages of the 100 accuracy rates were computed for evaluation, respectively. One-versus-all mechanism was adopted to realize multi-classification.

As shown in Table 4, KPCA-GPR achieved the highest accuracy rate among the three methods that are GPR, KPCA-GPR and SVM-RBF on Bordatz. As for the data set of Food, KPCA-GPR also obtained the best performance among the three methods. On the data set of kth-tips, SVM-RBF

achieved an accuracy rate of 94.22%, which is slightly higher than an accuracy rate of 94.02% obtained by KPCA-GPR. In summary, KPCA-GPR outperformed SVM-RBF on the texture data sets.

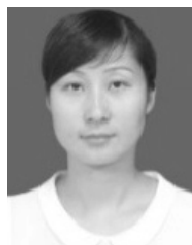
V. CONCLUSION

Smoke detection can be regarded as a specific kind of two-class texture classification. We improve smoke classification accuracy by non-linearly mapping LBP-like features into a low-dimensional space and modeling classification as a Gaussian Process. We extract LBP-like features, which are one of the best texture descriptors. To further obtain discriminant features, KPCA is adopted to non-linearly map the LBP-like features into a low-dimensional space, where the manifold structures of data are resided. To improve generalization performance, GPR is used to model classification as a Gaussian Process without the assumptions about the structures of data. Therefore, we combine LBP-like features, KPCA and GPR together to propose a novel smoke detection pipeline. In summary, the pipeline consists of three steps. In the first step, we use LBP-like methods to extract high-dimensional features. The second step uses KPCA to map the high dimensional features to low-dimensional features. In the last step, we apply GPR to classify the low dimensional features. To further improve performance or exploration, we can easily replace any of the three steps in the pipeline by similar methods. Experimental results show that combination of LBP, KPCA and GPR is able to improve the performance of smoke detection and texture classification, and our method obviously outperforms LBP-like methods with SVM.

REFERENCES

- [1] R. J. Ferrari, H. Zhang, and C. R. Kube, "Real-time detection of steam in video images," *Pattern Recognit.*, vol. 40, no. 3, pp. 1148–1159, Mar. 2007.
- [2] J. Gubbi, S. Marusic, and M. Palaniswami, "Smoke detection in video using wavelets and support vector machines," *Fire Safety J.*, vol. 44, no. 8, pp. 1110–1115, Nov. 2009.
- [3] R. J. Ferrari, H. Zhang, and C. R. Kube, "Real-time detection of steam in video images," *Opt. Eng.*, vol. 40, no. 3, pp. 1148–1159, Mar. 2007.
- [4] F. Yuan, "A fast accumulative motion orientation model based on integral image for video smoke detection," *Pattern Recognit. Lett.*, vol. 29, no. 7, pp. 925–932, May 2008.
- [5] F. Yuan, "A double mapping framework for extraction of shape-invariant features based on multi-scale partitions with Adaboost for video smoke detection," *Pattern Recognit.*, vol. 45, no. 12, pp. 4326–4336, Dec. 2012.
- [6] F. Yuan, Z. Fang, S. Wu, Y. Yang and Y. Fang, "Real-time image smoke detection using staircase searching-based dual threshold AdaBoost and dynamic analysis," *IET Image Process.*, vol. 9, no. 10, pp. 849–856, Oct. 2015.
- [7] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.
- [8] F. Yuan, J. Shi, X. Xia, Y. Yang, Y. Fang, and R. Wang, "Sub oriented histograms of local binary patterns for smoke detection and texture classification," *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 4, pp. 1807–1823, Apr. 2016.
- [9] X. Qian, X.-S. Hua, P. Chen, and L. Ke, "PLBP: An effective local binary patterns texture descriptor with pyramid representation," *Pattern Recognit.*, vol. 4, nos. 10–11, pp. 2502–2515, Oct./Nov. 2011.
- [10] F. Yuan, "Rotation and scale invariant local binary pattern based on high order directional derivatives for texture classification," *Digit. Signal Process.*, vol. 26, no. 3, pp. 142–152, Mar. 2014.
- [11] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2005, p. 13.
- [12] Y. Wen, L. He, and P. Shi, "Face recognition using difference vector plus KPCA," *Digit. Signal Process.*, vol. 22, no. 1, pp. 140–146, Jan. 2012.
- [13] Y. Lei, M. Bennamoun, M. Hayat, and Y. Guo, "An efficient 3D face recognition approach using local geometrical signatures," *Pattern Recognit.*, vol. 47, no. 2, pp. 509–524, Feb. 2014.
- [14] L. Xu, J. Li, A. Wong, and C. Wang, "A KPCA texture feature model for efficient segmentation of RADARSAT-2 SAR sea ice imagery," *Int. J. Remote Sens.*, vol. 35, no. 13, pp. 5053–5072, 2014.
- [15] F. Zhou, T. Yuan, W. Yang, and Q. Liao, "Single-image super-resolution based on compact KPCA coding and kernel regression," *IEEE Signal Process. Lett.*, vol. 22, no. 3, pp. 336–340, Mar. 2015.
- [16] V. Savic, E. G. Larsson, J. Ferrer-Coll, and P. Stenumgaard, "Kernel methods for accurate UWB-based ranging with reduced complexity," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1783–1793, Mar. 2016.
- [17] Y. Zhang, S. Li, S. Wang, and Y. Q. Shi, "Revealing the traces of median filtering using high-order binary local ternary patterns," *IEEE Signal Process. Lett.*, vol. 21, no. 3, pp. 275–279, Mar. 2014.
- [18] K.-W. Cheng, Y.-T. Chen, and W.-H. Fang, "Gaussian process regression-based video anomaly detection and localization with hierarchical feature representation," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5288–5301, Dec. 2015.
- [19] L. Liu, L. Shao, F. Zheng, and X. Li, "Realistic action recognition via sparsely-constructed Gaussian processes," *Pattern Recognit.*, vol. 47, no. 12, pp. 3819–3827, Dec. 2014.
- [20] K. Zhu, D. Gong, Z. Li, and X. Tang, "Orthogonal Gaussian process for automatic age estimation," in *Proc. 22nd ACM Int. Conf. Multimedia*, vol. 19, 2014, pp. 857–860.
- [21] E. Challis, P. Hurley, L. Serra, M. Bozzali, S. Oliver, and M. Cercignani, "Gaussian process classification of Alzheimer's disease and mild cognitive impairment from resting-state fMRI," *NeuroImage*, vol. 112, pp. 232–243, May 2015.
- [22] D. Lee, H. Park, and C. D. Yoo, "Face alignment using cascade Gaussian process regression trees," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 4204–4212.
- [23] A. Dhall and R. Goecke, "Group expression intensity estimation in videos via Gaussian processes," in *Proc. 21st Int. Conf. Pattern Recognit. (ICPR)*, Nov. 2012, pp. 3525–3528.
- [24] K. Markov and T. Matsui, "Music genre and emotion recognition using Gaussian processes," *IEEE Access*, vol. 2, pp. 688–697, 2014.
- [25] J. Fink, A. Ribeiro, and V. Kumar, "Robust control of mobility and communications in autonomous robot teams," *IEEE Access*, vol. 1, pp. 290–309, 2013.
- [26] C. Long and G. Hua, "Multi-class multi-annotator active learning with robust Gaussian process for visual recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2839–2847.
- [27] R. Urtasun and T. Darrell, "Discriminative Gaussian process latent variable model for classification," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 927–934.
- [28] C. Lu and X. Tang, "Surpassing human-level face verification performance on LFW with GaussianFace," in *Proc. Nat. Conf. Artif. Intell.*, 2015, pp. 3811–3819.
- [29] N. Lawrence, "Probabilistic non-linear principal component analysis with Gaussian process latent variable models," *J. Mach. Learn. Res.*, vol. 6, pp. 1783–1816, Nov. 2005.
- [30] L. I. Smith, "A tutorial on principal components analysis," *Inf. Fusion*, vol. 51, no. 3, p. 16, 2002.
- [31] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.
- [32] H. Hoffmann, "Kernel PCA for novelty detection," *Pattern Recognit.*, vol. 40, no. 3, pp. 863–874, Mar. 2007.
- [33] G. Baudat and F. Anouar, "Generalized discriminant analysis using a kernel approach," *Neural Comput.*, vol. 12, no. 10, pp. 2385–2404, 2000.
- [34] D. Cai, X. He, and J. Han, "Speed up kernel discriminant analysis," *Vldb J.*, vol. 20, no. 1, pp. 21–33, 2011.
- [35] B. Schölkopf and A. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2002.
- [36] J. A. Rice, *Mathematical Statistics and Data Analysis*, 3rd ed. Belmont, CA, USA: Duxbury Press, 2007.
- [37] F. Yuan, "Video-based smoke detection with histogram sequence of LBP and LBPV pyramids," *Fire Safety J.*, vol. 46, no. 3, pp. 132–139, Apr. 2011.

- [38] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 389–396, 2011.
- [39] S. R. Dubey, S. K. Singh, and R. K. Singh, "Multichannel decoded local binary patterns for content-based image retrieval," *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 4018–4032, Sep. 2016.
- [40] X. Qi, R. Xiao, C.-C. Li, Y. Qiao, J. Guo, and X. Tang, "Pairwise rotation invariant co-occurrence local binary pattern," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2199–2213, Nov. 2014.
- [41] F. Yuan, J. Shi, X. Xia, Y. Fang, Z. Fang, and T. Mei, "High-order local ternary patterns with locality preserving projection for smoke detection and image classification," *Inf. Sci.*, vol. 372, no. 1, pp. 225–240, Dec. 2016.



JINTING SHI received the B.E. degree in computer science and technology from Jiangxi Normal University, Nanchang, China, in 2003, and the M.S. degree in computer science and technology from Jiangxi Agricultural University, Nanchang, China, in 2008. She is currently pursuing the Ph.D. degree with the School of Information Technology, Jiangxi University of Finance and Economics, Nanchang. Her research interests include image processing and pattern recognition.



FEINIU YUAN received the B.Eng. and M.E. degrees in mechanical engineering from the Hefei University of Technology, Hefei, China, in 1998 and 2001, respectively, and the Ph.D. degree in pattern recognition and intelligence system from the University of Science and Technology of China (USTC), Hefei, in 2004. From 2004 to 2006, he was a Post-Doctoral Researcher with the State Key Laboratory of Fire Science, USTC. From 2010 to 2012, he was a Senior Research Fellow with the

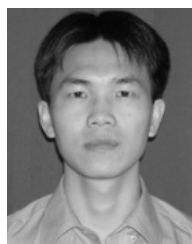
Singapore Bioimaging Consortium, Agency for Science, Technology and Research, Singapore. He is currently a Full Professor and a Ph.D. supervisor with the Jiangxi University of Finance and Economics. His research interests include 3-D modeling, image processing, and pattern recognition.



HONGDI LI received the B.E. degree in information management and information system from the Hefei University of Technology, Hefei, China, in 2007. She is currently pursuing the master's degree with the School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, China. Her research interests include information management, image processing, and pattern recognition.



XUE XIA received the B.E. degree in film and TV arts and technology and the M.E. degree in communication and information engineering from Shanghai University, Shanghai, in 2011 and 2014, respectively. She is currently pursuing the Ph.D. degree with the School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, China. Her research interests include 3-D display technology, image processing, and pattern recognition.



GANG LI received the B.E. degree in computer science and technology from Yichun University, Yichun, China, in 2000, and the M.S. degree in computer science and technology from Nanchang Hangkong University, Nanchang, China, in 2009. He is currently pursuing the Ph.D. degree with the School of Information Technology, Jiangxi University of Finance and Economics, Nanchang. His research interests include image processing and pattern recognition.

...