

Preferred Vector Machine for Forest Fire Detection

Xubing Yang^{1*}, Zhichun Hua¹, Li Zhang¹, Xijian Fan¹, Fuquan Zhang¹,
Qiaolin Ye², and Liyong Fu^{3*}

¹*College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China*

²*State Key Laboratory of Tree Genetics and Breeding, Co-Innovation Center for Sustainable Forestry in Southern China, Key Laboratory of Tree Genetics and Biotechnology of Educational Department of China, Key Laboratory of Tree Genetics and Silvicultural Sciences of Jiangsu Province, College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China*

³*Institute of Forest Resource Information Techniques, Chinese Academy of Forestry, Beijing, 100091, China*

Abstract

Machine learning-based fire detection/recognition is very popular in forest-monitoring systems. However, without considering the prior knowledge, e.g., equal attention on both classes of the fire and non-fire samples, fire miss-detected phenomena frequently appeared in the current methods. In this work, considering model's interpretability and the limited data for model-training, we propose a novel pixel-precision method, termed as PreVM (Preferred Vector Machine). To guarantee high fire detection rate under precise control, a new L_0 norm constraint is introduced to the fire class. Computationally, instead of the traditional L_1 re-weighted techniques in L_0 norm approximation, this L_0 constraint can be converted into linear inequality and incorporated into the process of parameter selection. To further speed up model-training and reduce error warning rate, we also present a kernel-based L_1 norm PreVM (L_1 -PreVM). Theoretically, we firstly prove the existence of dual representation for the general L_p ($p \geq 1$) norm regularization problems in RKHS (Reproducing Kernel Hilbert Space). Then, we provide a mathematical evidence for L_1 norm kernelization to conquer the

*Corresponding authors. Tel: +86-25-85427-687; Tax: +86-25-85427-212; Email Addresses: fuly@ifrit.ac.cn (Liyong Fu) and xbyang@njfu.edu.cn (Xubing Yang)

case when feature samples do not appear in pairs. The work also includes an extensive experimentation on the real forest fire images and videos. Compared with the-state-of-art methods, the results show that our PreVM is capable of simultaneously achieving higher fire detection rates and lower error warning rates, and L_1 -PreVM is also superior in real-time detection.

Keywords: forest fire detection, fire detection rate, error warning rate, SVM, dual representation

1. INTRODUCTION

Forest fire is the most destructive in three forest disasters (fire, pest and disease). Global warming and intense human activities greatly increase the frequency of wildfire incidents in recent years, as Greenpeace research laboratories and climate change research of US Environmental Protection Agency reported[1]. According to Statistical Yearbook of China, there were the total of 2936 and 3223 forest fire incidents in 2015 and 2017, respectively. Amongst 30 specific ignition causes, more than 95% incidents involved in human activities[2]. People have devoted more efforts on forest fire prevention systems, e.g., monitoring by observation towers, cruising aircrafts, meteorological satellites, or Internet of Things based networks[3]. Due to large-area, all-weather, and low-cost, especially the ability to monitor accidental fires caused by human behaviors, the vision-based watchtower-monitoring, usually equipped with high-definition dome (video) cameras for capturing a full 360° horizontal and 180° vertical view [1, 2], has been increasingly predominant in forest surveillance systems.

In vision-based fire detection/recognition, the existing methods are commonly divided into three categories[4, 5]. The first is usually called *color- or rule-based detection* because it focused on the refinement of color-rule (or pixel-threshold), empirically refined from the specific color space or multi-space combination[6]. The fire decision is originated from pixel classification[7, 8], i.e., if a given pixel can pass all rules, it will be identified as a fire pixel; otherwise it is a non-fire pixel. Although it is easy to use, the pixel-by-pixel verification

cannot meet the requirement of real-time detection. To speed video detection, the *motion-based* method appeared. It firstly concentrated on detecting
25 chaotic moving objects besides fire flames, and this step is called motion detection. Then only the detected movable objects will be further recognized, e.g., by using the rule-based methods [9, 10]. Later, this category is also extended to fire feature construction, such as multi-channel color components, two-order variance or higher-order statistics, local neighborhood information,
30 and other auxiliary features like wavelet transformation, etc[11]. For the first two categories, they may be efficient for the dead-directional videos, but fail to detect fire from omni-directional videos[12], e.g., obtained from the tower-monitoring forest scene, because all video objects are movable across frames. Moreover, due to the individual difference between different forest scenes, it is
35 also unrealistic if want to construct a universal rule suitable for all scenarios. In last decade, computer vision and machine learning algorithms have been widely incorporated into fire detection and gradually occupied the dominant position. Generally, this category needs to train a model, thus it is named *model-based* fire detection. According to the architecture of selected models, it has two research
40 directions. The mainstream one is following deep learning, including CNN (convolutional neural network), deep CNN, Fast R-CNN (region-based CNN), DSN (deep saliency network) [13] and so-called light-weight networks MobileNet[14], ShuffleNet[15], and GhostNet [16]. The other direction inherits conventional machine learning methods (shallow learning), involving in artificial neural networks (ANN)[17], unsupervised data clustering[18, 4], supervised classification or regression[11, 19, 20]. In contrast, the advantage of model-based methods lies in two-fold: (1) It outperforms the first two categories in real-time because the decision-making just relies on function values, rather than pixel-by-pixel and rule-by-rule verifications. Moreover, the decisions can also be made in batch; (2)
45 (2) It is more potential in achieving a high fire detection rate (i.e., true positive rate in pattern recognition, or shortly TP, if fire samples are viewed as drawn from positive class.) and a low error warning rate (false positive rate, FP) because most supervised models are trained on the studying scene.

However, in view of pattern recognition, these methods are just simple machine learning-based applications, without considering prior knowledge of forest fire problem. A typical prior is like that, for example, the loss caused by the miss-detected fire (TN: true negative) is much greater than that caused by error warning (FP). However, in the existing methods, taking the shallow-learning SVM as an example for the interpretability, both classes of losses are mixed together and usually measured by the same-scale misclassification rate. Additionally, although both losses are generated by sample misclassification, the resulting consequences are quite different. In forest fire recognition, the modest FP loss (error warning) is tolerable because there must be some plans about fire contingency; While the TN loss, caused by the fire miss-detected misclassification, is fatal or even disastrous because it may lead to an uncontrollable forest fire. However, to our best knowledge, the difference between TN and FP is always ignored by the existing fire detection methods besides deep networks. In essence, this ignoring is rooted in the selected models, in which both fire and non-fire samples are paid equal attentions, especially in the training phrase. Even worse, adopting the unique criteria for the performance assessment further confuses both types of misclassifications. As a result, the model is trained to the error direction towards fire-missed detection. To clarify this opinion, Figure 1 gives an example, in which Figs 1a and 1b show a forest fire image and the ground-truth of fire pixels. The figures 1d to 1f are the results of detected fire pixels, obtained by rule-reasoning[6], K-medoids clustering [18] and SVM-based detectors [11], respectively. Both types of misclassifications also appeared in the detected figures: (1) the TN errors of the miss-detected fire pixels are shown as black “holes”, e.g., the central area of fire flames; and (2) the FP errors of error-warning are shown as non-fire objects, e.g., the white sky or green leaves in Fig. 1e. This example states that, due to equal attention on the two-class samples and the single criteria for two types of misclassifications, there always exist more or less fire pixels that cannot be detected out by the existing methods. For the general object recognition, it may be nothing but just loses a little classification accuracy rate; but for fire detection it is fatal and even resulting

85 in a forest fire. Intuitively, a good fire detector should be able to achieve a very HIGH fire detection rate and a LOW error warning rate at the same time.
 90 However, in real applications, it is unrealistic or even impossible to approach both goals simultaneously. Then at least we should preferably emphasize high fire detection rate, even if sacrificing a little error warning rate. Naturally, a straightforward ambition is to learn a detector which can pay more attention
 95 on fire samples. We name the proposed scheme as PreVM (Preferred Vector Machine), following the naming of SVM in the traditional model-based fire detection. Fig. 1c shows our desired output: all fire pixels can be detected out, though some non-fire ones, e.g., the pixels located at the sky area, are wrongly recognized. Next we will also discuss how to reduce this type of misclassification for the non-fire class.

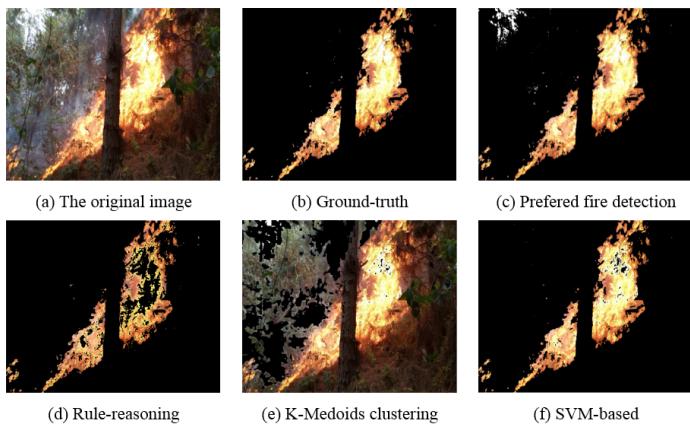


Figure 1: Illustration for the fire miss-detected problem. A given fire image and the ground-truth for fire pixels are shown in Figs. (a) and (b). The fire detected results are shown in Figs (d) to (f), obtained by rule-reasoning[6], K-medoids clustering[18] and SVM-based [11] methods, respectively. Fig. c is our desired output.

In contrast to shallow learning, the deep-version visual recognition is the mainstream in forest fire detection studies. However, scientists and technologists generally believe that a deep network, worked as a black-box, lacks interpretability, robustness and generalization. A typical counterexample is about one-pixel attack [21], which states that the output of deep networks is easily altered just
 100

by adding relatively small perturbations, or even one-pixel modification, to the input image. Reflecting in forest fire detection, it means that, although deep methods are capable of handling vast amounts of data, they ignore the variation in complexity of among training samples and as a result, the performance of a trained model is limited. Furthermore, deep learning approaches with little data and features perform poorly in some real-world challenging applications, e.g., with over-high cost of misclassification in the intelligent aerospace control and high-precision forest fire detection. In this work, we follow the route of shallow learning for the easy interpretability, as well as robustness and generalization.

In pattern recognition and machine learning, the most related work to our PreVM should be one-class SVM [22] and cost-sensitive SVM [23, 24], though they have not been used for fire detection in the literature. The former is an effective outlier detector, which aims to find a decision plane to separate one-class data (majority) from the origin where the outliers (minority) are supposed to lie. While for the latter, it is usually used to control false alarms with respect to Neyman-Pearson criterion. In the literature, there have two versions of the cost-sensitive SVM: C-SVM and 2C-SVM. The C-SVM, or called bias-shifting, is achieved by varying the bias of the standard SVM. While 2C-SVM imposes different penalties on two types of errors. Intuitively, our PreVM differs from them in the following aspects: (1) If using one-class SVM for fire detection, the fire pixels are naturally used as the normal samples (one-class). Thus in the use of prior knowledge, a large amount of non-fire information will be wasted though it may be useful for model-training. Furthermore, according to pixel distribution, the majority of samples should be non-fire pixels, especially in early fire stage; (2) The cost-sensitive SVM is essentially designed for unbalance classification. Then if using C-SVM for fire detection, it is easy to know that improving fire detection rate will inevitably increase error warning rate because the bias is shifting to the negative class. Likewise, for 2C-SVM, it may be useful for fire-missed detection problem if appropriate penalty factors are imposed on both types of misclassification losses. However, how to accurately control mis-

classification is still unknown. According to the Neyman-Pearson criterion[23] and the given ground truth of the negative class, a false alarm rate (FP) is paired to the FN, i.e., $FP = 1-FN$ in percentage, not our desired TP; (3) The difference between models will be addressed in the next section.

In contrast, PreVM is designed for the preferred high fire detection rate, then considering how to reduce error warning rate. Our contributions are highlighted as follows.

- We purposefully develop a simple and efficient method named PreVM for forest fire detection.
- To guarantee high fire detection rate under precise control, we introduce L_0 -norm constraint into the fire (positive) samples.
- To reduce error warning rate and speed training, we also provide L_1 -norm PreVM and its nonlinear version.
- Theoretically, we prove the existence of dual representations for the general L_p ($p \geq 1$) norm regularization problems in RKHS (Reproducing Kernel Hilbert Space). Then, we also give a mathematical evidence for L_1 norm kernelization when feature samples do not appear in pairs.
- The leading L_2 - and L_1 -PreVM regularization problems can be solved by standard quadratic and linear programmings, respectively.

The remainder of this paper is organized as follows. First, we briefly review the related work in Section 2. Then, in Section 3, we present the details of the multi-version PreVMs. Next in Section 4, we validate the performance of the proposed PreVM method, compared with the-state-of-art methods. Finally, we conclude the whole paper in Section 5.

2. RELATED WORK

In this section, we briefly review two categories of fire detectors and the most related methods to our PreVM.

¹⁶⁰ 2.1. Fire detectors

Here we take the YCbCr rule-reasoning and SVM based fire detectors as examples, and review them as follows.

2.1.1. YCbCr-rule based fire detection

As described in[6], if a YCbCr pixel satisfies all of the following rules, it is classified to the fire class; otherwise it is classified to the non-fire.

$$\begin{aligned} Y(i,j) &> Cb(i,j), \quad Cr(i,j) > Cb(i,j), \quad |Cb(i,j) - Cr(i,j)| \geq \tau \\ Y(i,j) &> Y_m, \quad Cb(i,j) < Cb_m, \quad Cr(i,j) > Cr_m \end{aligned} \quad (1)$$

where the subscript m denotes the mean of the corresponding channel, and the symbol τ in the third rule is an empirical parameter.
¹⁶⁵

2.1.2. SVM-based fire detection

Generally, training a model-based detector consists of two steps. The first step is to construct training samples, e.g., by using pixels or pixel blobs in shallow learning, or by normalizing images or video frames in deep learning[11, 19]. The second is about model selection (or network architecture) and model-training. It is worth mentioning that in shallow learning, SVM has been paid wide attention[19, 20]. Here we briefly review Ko's work[11], due to the fact that SVM is first used for fire detection. After selecting fire and non-fire pixels by the Gaussian function, SVM is trained on the selected two-class pixel-pattern samples $\{\mathbf{x}_i, y_i\} \in R^d \times \{-1, 1\}$, $i = 1, 2, \dots, l$, constructed by color values and the auxiliary features from 1-level wavelet. A standard SVM is usually defined as

$$\min_{\mathbf{w}, \boldsymbol{\delta}, b} \|\mathbf{w}\|_2^2 / 2 + C \mathbf{1}^T \boldsymbol{\delta} \quad (2)$$

$$s.t. \quad y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1 + \delta_i \geq 0 \quad (3)$$

$$\delta_i \geq 0, i = 1 \sim l. \quad (4)$$

The constraint (3) states that two-class training samples are paid equal attentions, and misclassification is measured by the same-scale empirical errors.

2.2. The related SVM variants

¹⁷⁰ In view of paying different attentions on training samples, the relevant one-class SVM and cost-sensitive 2C-SVM are briefly reviewed as follows.

2.2.1. One-class SVM

In addition to the previous notations, a one-class SVM is defined as,

$$\begin{aligned} & \min_{\mathbf{w}, \boldsymbol{\delta}, \rho} \|\mathbf{w}\|_2^2 / 2 + C \mathbf{1}^T \boldsymbol{\delta} - \rho \\ & \text{s.t. } \mathbf{w}^T \mathbf{x}_i^+ - \rho + \delta_i \geq 0. \end{aligned} \quad (5)$$

where \mathbf{x}_i^+ denotes a positive sample. In contrast to SVM, it can be explained that it just focuses on the fire samples, and discards all non-fire ones.

¹⁷⁵

2.2.2. 2C-SVM

The C-SVM is simpler, which can be obtained by shifting the bias of SVM. Here we just briefly review 2C-SVM, defined as the follow optimization, in which the same constraints to SVM are omitted.

$$\min_{\mathbf{w}, \boldsymbol{\delta}, b} \|\mathbf{w}\|_2^2 / 2 + C\gamma \sum_{\{y_i=1\}} \delta_i + C(1-\gamma) \sum_{\{y_i=-1\}} \delta_i \quad (6)$$

where the cost parameter γ ($0 \leq \gamma \leq 1$) is used to control the ratio of two types of errors: $C\gamma$ for the positive class and $C(1-\gamma)$ for the negative.

¹⁸⁰ As aforementioned, a 2C-SVM is designed for unbalance classification. When it is used for the fixed probability of false alarm, Mata-Moya et al. proposed an approximation method [23] based on the likelihood ratio. However, it is not easy for selecting an appropriate threshold of this ratio, which involves in the varied γ , probability or likelihood function estimations of the null hypothesis and the alternative hypothesis, Receiver Operating Characteristic (ROC) curve, etc. Moreover, in view of optimization, the margin term $\|\mathbf{w}\|_2^2 / 2$ in the above objective functions is just the substitute of $\|\mathbf{w}\|_2$ for easier calculation. ¹⁸⁵ However, this substitution also brings some negative effects into SVMs, e.g., non-robustness and higher-order optimization (from the original one-order).

3. OUR PREFERRED SUPPORT MACHINE

For a given learning problem, No Free Lunch (NFL) theorem[25] states that
 190 there is no well performing universal metaheuristic, but the heuristics must be tailored to the problem using prior knowledge. However, this has not attracted enough attention in many applications. To emphasize fire detection rate, we incorporate the aforementioned knowledge into SVM, and propose our PreVM.

3.1. Modeling

For a good fire detector f , it should be able to correctly classify all fire samples to the positive class. Here f denotes a decision function, usually defined as $f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b$. To approach this goal, SVM aims to classify samples at least at margin 1 for the better generalization. On the other hand, to maintain the non-emptiness and non-degeneration of feasible domain, we continue to tolerate the misclassification of negative samples. Thus, the constraints (3) and (4) are modified as below.

$$\mathbf{w}^T \mathbf{x}_i + b \geq 1, \quad \mathbf{x}_i \in \omega^+ \quad (7)$$

$$\delta_i \geq 0, \quad i = 1, 2, \dots, l_2. \quad (8)$$

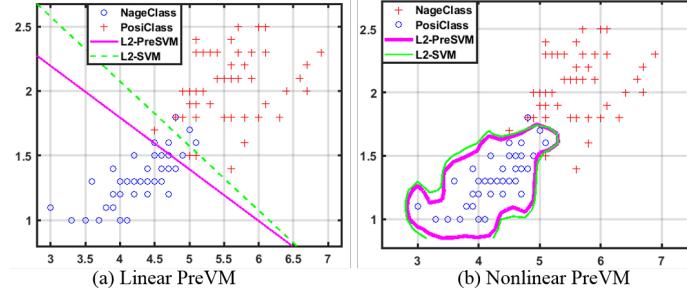


Figure 2: An example for PreVM.

195 To explain this modification, Figure 2 shows an example, in which two types of decision planes, linear or nonlinear, are generated from L_2 -norm SVM (shortly L_2 -SVM, hereafter) and our L_2 -PreVM, respectively. In linear case, as shown

in Fig. 2a, the main differences lie in three-fold: (1) Linear PreVM is able to correctly classify positive samples, while SVM is not due to equal attentions on two-class samples; 2) The non-parallelism between decision planes states that PreVM is not a C-SVM, the translation version of SVM; 3) PreVM runs faster than SVM because of less slack variables need to be optimized. Similarly, the nonlinear PreVM is still superior to SVM in shorter training time and tighter decision boundary for the negative class, as shown in Fig. 2b, with RBF kernel parameters 0.08 for PreVM and 0.1 for SVM, respectively. Note that there may exist those points which have the same property values but from different classes, e.g., the point (4.8,1.8) in Fig. 2. Intuitively, in order to resist fire miss-detected problem, those points should be classified to the fire class, which is significantly different from the general pattern recognition.

For a real forest scene, inevitably, there may have some fake fire-like objects, such as lamp light, reddish brown rocks or soil, firefighting garments or facilities, etc. To resist the oversensitivity to fake objects, we attempt to slightly relax the constraint to the fire class, but this relaxation should be moderate, e.g., guided by forest fire experts or fire risk index (FRI) models[26]. For simplicity, if letting fire detection rate be a (in percentage), then fire miss-detected misclassification is bounded by $(1 - a)$. Our PreVM is defined in 9.

$$\begin{aligned} & \min_{\mathbf{w}, b, \boldsymbol{\delta}, \boldsymbol{\eta}} \|\mathbf{w}\|_2^2 / 2 + C_1 \mathbf{1}^T \boldsymbol{\eta} + C_2 \mathbf{1}^T \boldsymbol{\delta} \\ & \text{s.t. } \mathbf{w}^T \mathbf{x}_i + b \geq 1 - \eta_i, \quad \mathbf{x}_i \in \omega^+ \\ & \quad \mathbf{w}^T \mathbf{x}_j + b \leq -1 + \delta_j, \quad \mathbf{x}_j \in \omega^- \\ & \quad \|\boldsymbol{\eta}\|_0 \leq \Delta, \quad \boldsymbol{\delta} \geq \mathbf{0}, \quad \boldsymbol{\eta} \geq \mathbf{0} \end{aligned} \tag{9}$$

where $\boldsymbol{\eta} = (\eta_1, \dots, \eta_{l_1})^T$, $\boldsymbol{\delta} = (\delta_1, \dots, \delta_{l_2})^T$, and $\Delta = l_1 * (1 - a)$, where a is the pre-specified fire detection rate. The constraint $\|\boldsymbol{\eta}\|_0 \leq \Delta$ is used for the precise control on the misclassification of the fire samples.

Owing to nice property of L_1 -SVM in simultaneously performing classification and feature selection[27], our L_1 -PreVM is defined in 10, with the same

constraints to L_2 -PreVM.

$$\min_{\mathbf{w}, \mathbf{b}, \boldsymbol{\delta}, \boldsymbol{\eta}} \|\mathbf{w}\|_1 + C_1 \mathbf{1}^T \boldsymbol{\eta} + C_2 \mathbf{1}^T \boldsymbol{\delta} \quad (10)$$

Our PreVM differs from 2C-SVM or the fixed probability of false alarm based 2C-SVM. The main differences lie in two-fold: (1) It is easy to use, but 2C-SVM based false alarm model is a two-step method; (2) It has a clear geometrical interpretation; while 2C-SVM does not because the relationship between the γ in 2C-SVM and the threshold of likelihood ratios is still unclear; (3) For PreVM, the misclassification of the fire samples can be accurately controlled by L_0 norm, e.g., using the number (or probability, according to the Law of Large Numbers [23]) of misclassified samples, rather than the values of misclassification loss as 2C-SVM does. Next we come to model solutions.

3.2. Solution to linear model

Although L_0 is not a norm, the attractive characteristics, e.g. strong sparsity, make it energetic in fields of compressive sensing and sparse signal process. Presently, L_0 norm optimizations commonly consist of approximate solutions and heuristic approaches[28]. Among them, due to the fast convergence, L_1 norm re-weighted techniques are popular in L_0 optimization. In order to explain our models easily, we briefly review this approximation in (11) and (12).

$$\|\boldsymbol{\omega} \boldsymbol{\eta}\|_1 \approx \|\boldsymbol{\eta}\|_0 \quad (11)$$

where $\boldsymbol{\omega} = diag(\omega_1, \omega_2, \dots, \omega_{l_1})$ is a diagonal matrix, w.r.t.,

$$\omega_i \cdot |\eta_i| = \begin{cases} 1, & \eta_i = 1 \\ 0, & otherwise \end{cases} \quad (12)$$

From (9) and (12), $\boldsymbol{\eta} \geq \mathbf{0}$ means $\boldsymbol{\omega} \geq \mathbf{0}$, such that $\|\boldsymbol{\omega} \boldsymbol{\eta}\|_1 = \mathbf{1}^T \boldsymbol{\omega} \boldsymbol{\eta}$ holds. Thus, the non-convex problem (9) can be converted into a convex one.

²²⁵ 3.2.1. Solution to L2-PreVM

Replacing $\|\boldsymbol{\eta}\|_0$ with $\mathbf{1}^T \boldsymbol{\omega} \boldsymbol{\eta}$, the problem (9) becomes a QP optimization.

$$\begin{aligned} & \min_{\boldsymbol{\omega}} \frac{1}{2} \boldsymbol{\omega}^T \mathbf{H} \boldsymbol{\omega} + \mathbf{f}^T \boldsymbol{\omega} \\ & \text{s.t. } \mathbf{A} \boldsymbol{\omega} \leq \mathbf{c} \end{aligned} \quad (13)$$

where $\boldsymbol{\omega} = (\mathbf{w}^T, b, \boldsymbol{\eta}^T, \boldsymbol{\delta}^T)^T$. The related coefficient matrices in (13) are

$$\mathbf{H} = \begin{bmatrix} \mathbf{I} & \\ & \mathbf{0} \end{bmatrix}, \mathbf{f} = \begin{bmatrix} \mathbf{0} \\ C_1 \mathbf{1} \\ C_2 \mathbf{1} \end{bmatrix}, \mathbf{c} = \begin{bmatrix} -1 \\ \Delta \\ \mathbf{0} \end{bmatrix}, \mathbf{A} = \begin{bmatrix} -\mathbf{X}_1 & -1 & -\mathbf{I} & -\mathbf{0} \\ \mathbf{X}_2 & 1 & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & 0 & \mathbf{1}^T \boldsymbol{\omega} & \mathbf{0} \\ \mathbf{0} & 0 & -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & 0 & \mathbf{0} & -\mathbf{I} \end{bmatrix}. \quad (14)$$

where both \mathbf{X}_1 and \mathbf{X}_2 denote sample matrices. The optimal solution (\mathbf{w}^*, b^*) to the original problem (9), w.r.t. $\boldsymbol{\omega}^*$, can be formulated by $(\boldsymbol{\omega}_{1:d}^*, \boldsymbol{\omega}_{(d+1):(d+1)}^*)$, where $\boldsymbol{\omega}_{m:n}$ denotes a sub-vector of $\boldsymbol{\omega}$, drawn from the m - to n -th dimensions.

3.2.2. Solution to L1-PreVM

Likewise, let $\mathbf{w} = \mathbf{p} - \mathbf{q}$ ($\mathbf{p}, \mathbf{q} \geq \mathbf{0}$), such that $\|\mathbf{w}\|_1 = \mathbf{1}^T(\mathbf{p} + \mathbf{q})$. L_1 -PreVM defined in (10) is led to the following LP problem.

$$\begin{aligned} & \min_{\boldsymbol{\omega}} \mathbf{f}^T \boldsymbol{\omega} \\ & \text{s.t. } \mathbf{A} \boldsymbol{\omega} \leq \mathbf{c} \end{aligned} \quad (15)$$

²³⁰ where $\boldsymbol{\omega} = (\mathbf{p}^T, \mathbf{q}^T, b, \boldsymbol{\eta}^T, \boldsymbol{\delta}^T)^T$, and the coefficient matrices in (15) are

$$\mathbf{f} = \begin{bmatrix} \mathbf{1} \\ 0 \\ C_1 \cdot \mathbf{1} \\ C_2 \cdot \mathbf{1} \end{bmatrix}, \mathbf{c} = \begin{bmatrix} -1 \\ \Delta \\ \mathbf{0} \end{bmatrix}, \mathbf{A} = \begin{bmatrix} -\mathbf{X}_1 & \mathbf{X}_1 & -1 & -\mathbf{I} & \mathbf{0} \\ \mathbf{X}_2 & -\mathbf{X}_2 & 1 & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{0} & 0 & \mathbf{1}^T \boldsymbol{\omega} & \mathbf{0} \\ -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} \end{bmatrix}. \quad (16)$$

The optimal solution to (10) is obtained by solving (15), i.e., $\mathbf{w}^* = \boldsymbol{\varpi}_{1:d}^* - \boldsymbol{\varpi}_{(d+1):2d}^*$, $b^* = \boldsymbol{\varpi}_{(2d+1):(2d+1)}^*$. We conclude the aforesaid in Algorithm 1, where blue italic text is for training L_1 -PreVM.

Algorithm 1: Training PreVMs

Input : Sample sets \mathbf{X}_1 , \mathbf{X}_2 and pre-specified fire detection rate a ;

Output: The solution $\boldsymbol{\varpi}$.

- 1 Initialize parameters C_1 , C_2 , and Δ ; Set $\boldsymbol{\omega} = \mathbf{I}$, and Compute \mathbf{H} , \mathbf{f} , \mathbf{A} , and \mathbf{b} by (14) for L_2 -PreVM (*or compute \mathbf{f} , \mathbf{A} , and \mathbf{b} by (16) for L_1 -PreVM*);
 - 2 **repeat**
 - 3 Solve QP problem (13) (*or solve LP problem(15)*) and obtain $\boldsymbol{\varpi}$;
 - 4 Update the sub-vector $\boldsymbol{\eta}$ w.r.t. $\boldsymbol{\varpi}$;
 - 5 Solve re-weighted L1 norm minimization problem $\min \|\boldsymbol{\omega}\boldsymbol{\eta}\|_1$;
 - 6 Update $\boldsymbol{\omega}$ and \mathbf{A} by (14) (*or by (16)*) ;
 - 7 **until** $\boldsymbol{\varpi}$ convergence;
-

235 In each updating step, Algorithm 1 has to solve a L_1 norm minimization problem, obviously it is time-consuming. Note that $\|\boldsymbol{\eta}\|_1 = \mathbf{1}^T \boldsymbol{\eta}$ holds iff $\boldsymbol{\eta} \geq \mathbf{0}$. This reminds us that, if replacing $\|\boldsymbol{\eta}\|_0 \leq \Delta$ with $\mathbf{1}^T \boldsymbol{\eta} \leq \Delta'$, the problems (9) and (10) are still convex. Moreover, without solving L_1 re-weighted minimization problem, the model-training can be speed-up. The difference between Δ and Δ' is that: the Δ in (9) is fixed, i.e., $\Delta = l_1(1 - a)$; while here Δ' is changing but subject to Δ . Thus, choosing Δ' can be incorporated into tuning parameter step by increasing or decreasing Δ' . That is, for the $\boldsymbol{\eta}$ in the current tuning step, if $\|\boldsymbol{\eta}\|_0 \leq \Delta$ holds, Δ' should be decreased because the current upper bound Δ' for $\mathbf{1}^T \boldsymbol{\eta}$ is loose; otherwise it should be increased. For the 240 convenience of readers, let $\boldsymbol{\omega} = \mathbf{I}$ in Eqs. (14) and (16), and conclude the above in Algorithm 2, titled “Fast PreVM”. To explain the precise control, we also provide an example in Figure 3.

Algorithm 2: Training Fast PreVMs

Input : Training samples $\mathbf{X}_1, \mathbf{X}_2$ and the pre-specified fire detection rate a ;

Output: The solution ϖ .

- 1 Set $\omega = \mathbf{I}$ and $\Delta' = \Delta$;
 - 2 Parameter selection for C_1, C_2 , and Δ' ;
 - 3 Compute $\mathbf{H}, \mathbf{f}, \mathbf{A}$, and \mathbf{c} by (14) for L_2 -PreVM (*or compute \mathbf{f}, \mathbf{A} , and \mathbf{c} by (16) for L_1 -PreVM*) ;
 - 4 Solve QP problem (13) (*or solve LP problem(15)*) to obtain ϖ .
-

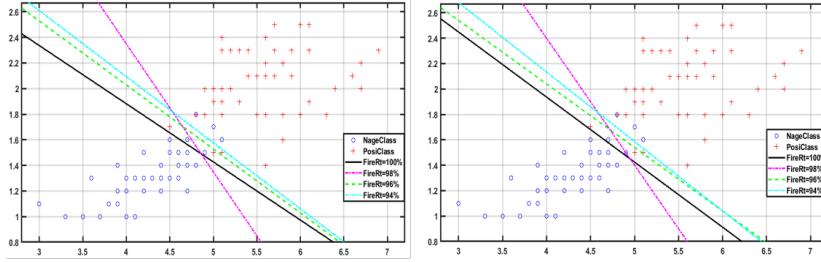


Figure 3: Precise control on L_2 -PreVM (left) and L_1 -PreVM (right), with different fire detection rates (FireRt).

Figure 3 shows that multiple decision planes are precisely controlled by the fire detection rates (FireRt) of $100\% \sim 94\%$, which exactly correspond to the number of misclassified positive samples, $0 \sim 3$, respectively. The planes of L_1 -PreVM seem to be closer to the positive class than those of L_2 -PreVM. This can be explained that a L_1 regularization has more accurate solution than that of L_2 norm. Here the results of re-weighted PreVMs are not provided because they have similar decision planes except for more training time. On the real forest fire images, Figure 4 shows the detected fire pixels with the predefined FireRt $a = 99\%$. Visually, it shows that almost all fire pixels can be detected out by the proposed PreVMs, however some of them are missed by SVM. After training on the 2,000 pixel-pattern samples, they achieve the average fire detection rates

of 99.87%, 99.92% and 92.43%, and the error warning rates of 3.20%, 1.98%
260 and 2.15%, respectively. In training time, L_1 -PreVM wins the best, achieving
the time ratios at 1:276 to L_2 -PreVM and 1:260 to L_2 -SVM, respectively. An
explanation is about the complexity of time. For example, the complexity of
solving a L_1 -SVM is just $\mathcal{O}(l(q \sim q^3))$ [29], where l and q denote the number
265 of training samples and non-zero components of the solution, respectively. It is
almost two magnitudes lower than that of L_2 -SVM ($\mathcal{O}(l^2 \sim l^3)$) because $l \gg q$
holds in most cases.

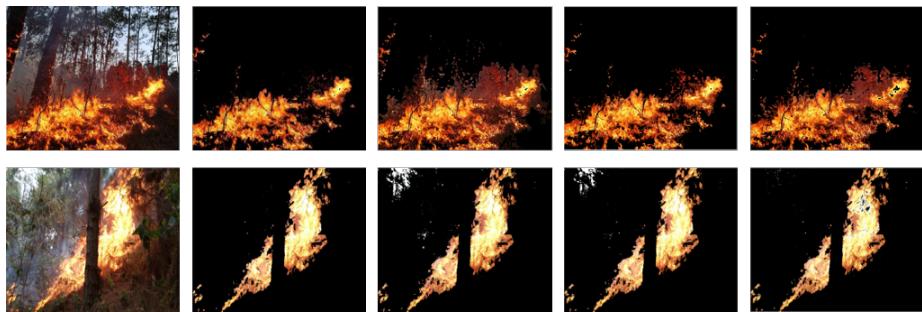


Figure 4: Visualization for forest fire detection by linear PreVMs and L_2 -SVM. The figures in Column 1 and 2 are the original and annotation images; and the columns 3 to 5 show the detected fire, by L_2 -PreVM, L_1 -PreVM and L_2 -SVM, respectively.

Compared with the L_2 -norm based SVM and PreVM, L_1 -PreVM shows the
superiority in training time, high fire detection rate and low error warning rate,
as shown in Fig. 4. However, for a real classification task, it is not always linear
270 but nonlinear (i.e. linearly inseparable). Moreover, the nonlinear example in
Fig. 2b shows that kernel-based learning machines have powerful performance
in coping with linearly inseparable problems. It will be benefit to forest fire de-
tection if L_1 -PreVM can also be extended to the nonlinear version and discussed
275 in the kernel-induced feature space. Next we firstly come to dual representation
problem, which is very important for the L_1 norm nonlinearization or called
kernelization.

3.3. Related theorems

3.3.1. Representer theorems for L_p -norm

For L_2 -norm learning machines such as SVM, perceptron[30], and ridge regression[31], nice algebra descriptions in the dual forms, generally expressed by $\mathbf{w} = \sum_i \alpha_i \mathbf{x}_i$, make them successful in the nonlinearization. As a natural extension, the dual form in the higher-dimensional feature space can also be expressed by $\mathbf{w}^\phi = \sum_i \alpha_i \phi(\mathbf{x}_i)$, where ϕ is the kernel-induced implicit mapping. If replacing \mathbf{w} with \mathbf{w}^ϕ , solving \mathbf{w}^ϕ in the original problem can be converted into solving its dual vector $\boldsymbol{\alpha}$, where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots)^T$. Moreover, these duals not only can be directly derived from models, sometimes also have some special properties, e.g., the sparsity in L2-SVM (support vectors), matrix singularity correction in ridge regression, convergent learning rate in perceptron, etc. However, due to non-smoothness and non-differentiability, it is not easy for L_1 -norm machines if want to obtain similar dual representations. To our best knowledge, there still has no rigorous proofs in the literature. Next we begin with the representer theorem[32], described in Lemma 1.

Lemma 1. *For a given nonempty set X , a positive definite kernel k on $X \times X$, a training set with l instances $\{(\mathbf{x}_i, y_i) \in X \times R\}$, a strictly monotonically increasing function $g \in [0, +\infty]$, an arbitrary cost function $c : (X \times R \times R)^l \rightarrow R$, and a function set defined as $\mathcal{F} = \{f \in R^X | f(\cdot) = \sum_{i=1}^l \beta_i k(\cdot, \mathbf{z}_i), \beta_i \in R, \mathbf{z}_i \in X, \|f\| < \infty\}$. Then for $\forall f \in \mathcal{F}$ minimizing the regularized risk functional $c((\mathbf{x}_i, y_i, f(\mathbf{x}_i))) + g(\|f\|)$ admits a representation of the form*

$$f(\cdot) = \sum_{i=1}^l \alpha_i k(\cdot, \mathbf{x}_i). \quad (17)$$

where $\|\cdot\|$ is a norm in the RKHS (Reproducing Kernel Hilbert Space) associated with the kernel k [32].

Lemma 1 states that a whole (infinite-dimensional) range learning machine, saying f , can be expressed as (finite-dimensional) expansions in terms of the limit training samples if it satisfies three prerequisites: non-empty training set,

positive definite kernel and strictly monotonic regularization function. Next we discuss the representation in RKHS.

300 *3.3.2. Dual representation for L_p -norm*

Assume ϕ is a mapping, defined as $\phi : X \mapsto \phi(X)$. $\phi(S)$ denotes a subspace spanned by l given samples $S = \{\mathbf{x}_i | \mathbf{x}_i \in X, i = 1 \sim l\}$, i.e., $\phi(S) = \{\phi(\mathbf{x}) | \phi(\mathbf{x}) = \sum_{i=1}^l \alpha_i \phi(\mathbf{x}_i), \alpha_i \in R\}$. Since $S \subset X$, $\phi(S) \subset \phi(X)$. For a given linear function $h(\mathbf{x}) = \langle \mathbf{w}^\phi, \phi(\mathbf{x}) \rangle$, or the associated equation (plane) $h(\mathbf{x}) = 0$ in feature space $\phi(X)$, where $\langle \cdot, \cdot \rangle$ denotes inner product, we have the following conclusions.

Theorem 1 . *Given a linear plane $\langle \mathbf{w}^\phi, \phi(\mathbf{x}) \rangle = 0$ in feature space $\phi(X)$, the weight \mathbf{w}^ϕ can be linearly expressed by the given training samples, i.e.,*

$$\mathbf{w}^\phi = \sum_{i=1}^l \alpha_i \phi(\mathbf{x}_i). \quad (18)$$

The proof is provided in Appendix A1. In essential, Theorem 1 states that (18) holds when h is homogeneous linear. Without loss of generality, if it is non-homogeneous, i.e. $h(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle - r$, it is easy to know that there must exist a point $\mathbf{x}_0 (\in X)$, such that $\langle \mathbf{w}, \mathbf{x}_0 \rangle = r$. Let $\mathbf{x}_0 = \mathbf{w}^+ r$, then $h(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle - r$ can be converted into $h(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} - \mathbf{x}_0 \rangle$, where $\mathbf{w}^+ = (\mathbf{w}\mathbf{w}^T)^{-1}\mathbf{w}$. However, if want to seek such an \mathbf{x}_0 in feature space, it is not easy because it is related to the space completeness. We describe it in Theorem 2.

Theorem 2 . *For any linear function $h(\mathbf{x}) = \langle \mathbf{w}^\phi, \phi(\mathbf{x}) \rangle - r$ defined in feature space $\phi(X)$, it can be expressed by $h(\mathbf{x}) = \langle \mathbf{w}^\phi, \phi(\mathbf{x}) - \phi(\mathbf{x}_0) \rangle$ iff $h(\mathbf{x})$ is complete. That is, in the subspace spanned by feature samples $\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_l)$, there must be a $\phi(\mathbf{x}_0)$, satisfying $\langle \mathbf{w}^\phi, \phi(\mathbf{x}_0) \rangle = r$.*

The proof is provided in Appendix A2. Theorem 2 states that, in a complete inner product space, a non-homogeneous linear function can also be expressed in the homogeneous form. Both of them share the same weight, \mathbf{w}^ϕ , thus the weight in the non-homogeneous function can also be linearly expressed by feature samples. This is concluded in Corollary 1.

Corollary 1 . For any linear function $h(\mathbf{x}) = \langle \mathbf{w}^\phi, \phi(\mathbf{x}) \rangle - r$ and a feature sample set $\phi(S)$ in feature space $\phi(X)$, the weight \mathbf{w}^ϕ can be linearly expressed by the given feature samples.

Proof. According to Theorem 2, for any h there must exist its homogeneous form, denoted by h' . By Theorem 1, we know that both h and h' share the same weight \mathbf{w}^ϕ , and $\mathbf{w}^\phi = \sum_i \alpha_i \phi(\mathbf{x}_i)$. ■

Note that the above conclusions hold in a complete RKHS. From the proof of Theorem 2, for any inner product space with the L_p -norm metric ($1 \leq p \leq +\infty$), it is complete. That is, the dual form (18) holds for all L_p -norm regularization problems. In practical view, this has been applied in the literature. For example, Zheng et al.[33] proposed a kernel L_1 -LDA (linear discriminant analysis) for feature extraction, which is just an application but no interpretation and evidence. Similar description can also be found in[29]. Guided by the above theorems, next we will interpret how to solve nonlinear L_1 -PreVM.

3.4. Solution to nonlinear model

3.4.1. L_1 -norm regularization in feature space

Firstly we consider L_1 norm regularization term $\|\mathbf{w}\|_1$. For simplicity, we rewrite (18) in matrix form, $\mathbf{w}^\phi = \phi(\mathbf{X})\boldsymbol{\alpha}$, where $\phi(\mathbf{X})$ denotes feature sample matrix, i.e., $\phi(\mathbf{X}) = (\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_l))$. In feature space, substituting (18) into $\|\mathbf{w}^\phi\|_1$, the margin term in (10) is led to the optimization $\min_{\boldsymbol{\alpha}} \|\phi(\mathbf{X})\boldsymbol{\alpha}\|_1$. In L_2 -norm case, because feature samples always appear in pairs, the kernel-induced learning machine is easily expressed by the inner products of the paired feature samples. Then they can be replaced with kernel function values. However, in L_1 -norm case, e.g., $\|\phi(\mathbf{X})\boldsymbol{\alpha}\|_1$, due to neither paired feature samples nor explicit expression for ϕ , we have to abandon the above-mentioned conventional methods and manage to find a new way to the L_1 solution. Fortunately, kernel pattern analysis[34] reminds us that selecting a kernel, including functional expression and kernel parameters, is independent from algorithms. Moreover, once this selection has been accomplished, the current ϕ is actually a confirmed mapping, and usually named kernel-induced mapping. Reflecting in $\|\phi(\mathbf{X})\boldsymbol{\alpha}\|_1$,

the derived feature samples in $\phi(\mathbf{X})$ are also independent from $\boldsymbol{\alpha}$, though they are still unknown. Therefore, we have the following conclusion.

Theorem 3 . *Given a real matrix $\Phi (\in R^{n \times m})$, if it is bounded, then for $\forall \boldsymbol{\alpha} \in R^m$, the following inequality holds.*

$$\|\Phi\boldsymbol{\alpha}\|_1 \leq M\|\boldsymbol{\alpha}\|_1. \quad (19)$$

355 where M is a positive constant.

Proof. It is obvious that (19) holds for $\boldsymbol{\alpha} = \mathbf{0}$. Without loss of generality, the next proof is for the general case $\boldsymbol{\alpha} \neq \mathbf{0}$. Rewrite Φ in the form of vectors, $\Phi = (\Phi_1, \dots, \Phi_m)$. Due to the boundedness of Φ , the constant M can be defined as $M = \max_i \|\Phi_i\|_1$, subject to $M < +\infty$. Let $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_m)^T$, such that $\|\Phi\boldsymbol{\alpha}\|_1 = \left\| \sum_{i=1}^m \alpha_i \Phi_i \right\|_1 \leq \sum_{i=1}^m |\alpha_i| \cdot \|\Phi_i\|_1 \leq M\|\boldsymbol{\alpha}\|_1$ holds. This is the end of proof. ■

3.4.2. Solution to nonlinear L1-PreVM

Here we only provide the solution to the nonlinear L1-PreVM. Similar processes can be extended to other L_1 regularizations. Let $f(\mathbf{x}) = \phi(\mathbf{x})^T \mathbf{w}^\phi + b$ be a linear decision function in feature space. By replacing \mathbf{w} with \mathbf{w}^ϕ , and using the theorems 2 and 3, we rewrite (10) in matrix form.

$$\begin{aligned} & \min_{\boldsymbol{\alpha}, b, \boldsymbol{\delta}, \boldsymbol{\eta}} \|\boldsymbol{\alpha}\|_1 + C_1 \mathbf{1}^T \boldsymbol{\eta} + C_2 \mathbf{1}^T \boldsymbol{\delta} \\ & \text{s.t. } \phi(\mathbf{X}_1)^T \phi(\mathbf{X}) \boldsymbol{\alpha} + \mathbf{1}^T b \geq \mathbf{1} - \boldsymbol{\eta} \\ & \quad \phi(\mathbf{X}_2)^T \phi(\mathbf{X}) \boldsymbol{\alpha} + \mathbf{1}^T b \leq -\mathbf{1} + \boldsymbol{\delta} \\ & \quad \mathbf{1}^T \boldsymbol{\eta} \leq \Delta' \\ & \quad \boldsymbol{\delta} \geq \mathbf{0}, \quad \boldsymbol{\eta} \geq \mathbf{0} \end{aligned} \quad (20)$$

where $\phi(\mathbf{X}_i)$ denotes the i -th class feature matrix, $i = 1, 2$.

Obviously, (20) can also be solved by a LP. For the sake of brevity and readability, here the technical details for (20) are omitted. Finally, a unseen sample \mathbf{z} can be determined by (21).

$$f(\mathbf{z}) = \phi(\mathbf{z})^T \phi(\mathbf{X}) \boldsymbol{\alpha} + b = \boldsymbol{\alpha}^T \mathbf{K}_{\mathbf{z}} + b \quad (21)$$

where $\mathbf{K}_{\mathbf{z}} = (k(\mathbf{x}_1, \mathbf{z}), \dots, k(\mathbf{x}_l, \mathbf{z}))^T$, called Empirical Kernel Mapping[34].

³⁶⁵ Analogously, the nonlinear L_1 -SVM can also be generated from (20), e.g., letting $C_1 = C_2$ and removing $\mathbf{1}^T \boldsymbol{\eta} \leq \Delta'$. Due to the direct optimization on $\|\boldsymbol{\alpha}\|_1$, the L_1 solution, $\boldsymbol{\alpha}$, should be more sparse than that in the corresponding L_2 version. This is expected to be helpful for real-time detection.

³⁷⁰ To show the efficiency of the proposed L_1 nonlinearization, a toy example on spiral data is given in Figure 5. Both L_1 -SVM and L_1 -PreVM are able to classify the linearly inseparable samples completely by the corresponding decision planes. Moreover, they both have less support vectors (SV) than their L_2 versions. In this example, L_1 -SVM (L_1 -PreVM) achieves the ratio at 3.7% (3.8%), between the number of SVs and total training samples; while L_2 -SVM (³⁷⁵ L_2 -PreVM) achieves the ratio at 23.1% (26.5%), respectively.

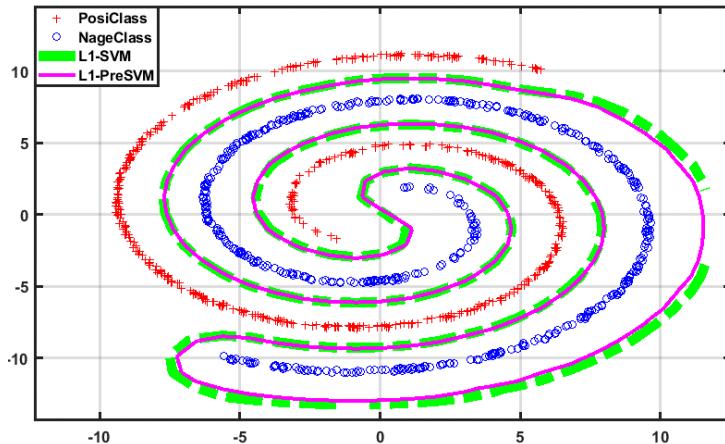


Figure 5: Nonlinear L_1 -SVM and L_1 -PreVM on two-class spiral data.

4. EXPERIMENTS

In this section, we perform a comparison between our PreVMs and state-of-the-art methods, including the typical fire detectors, e.g., the color-based rule-reasoning, the model-based K-medoids clustering or SVM, and the related ³⁸⁰ SVM variants one-class SVM and 2C-SVM. Considering the robustness, non-linearization and the one-order optimization, we also take L_1 -SVM as a base

method. The data used in this section is composed of the public (VisiFire, FLAME) and our collected images and videos. The ground-truth for fire pixels is obtained by the pixel-level annotation [35]. Since SVM is very popular in
 385 fire detection, it is trained in the manner of balance classification. That is, although two-class training samples are randomly drawn from the given dataset, the number of samples relies on the minority class. In most cases, it is determined by the number of fire pixels, especially in early fire detection. Then this training set is also used to train other models or guide K-medoids clustering,
 390 e.g., training one-class SVM with the positive samples, or selecting the fire cluster from the obtained clusters of K-medoids if it is the closest cluster to the positive samples. Instead of the single classification accuracy rate in machine learning, here two indicators, fire detection rate (TP) and error warning rate (FP), are adopted for the performance assessment, as fire detection methods
 395 usually do. Additionally, another two indicators are also adopted to meet the requirements of real-time detection and fire pixel positioning. To meet the demand of positive-definite kernel, as addressed in Lemma 1, RBF kernel function is chosen for the nonlinear machines. Parameter selection is carried on the tuning set, a 5% of training set. Concretely, the regularizers C , C_1 and C_2 are
 400 selected in the interval $\{10^i | i = -4, -3, \dots, 3, 4\}$; The kernel parameter ranges in $\{10^i | i = -4, \dots, 0, 1\}$. To keep the pace with the pre-specified TP a , the Δ' in PreVMs increases or decreases at the step 0.01. Likewise, the γ in 2C-SVM is set to $[0.5, 1]$. All comparisons are carried out on a Dell PC with a 2.83GHz Core 2 Quad CPU (4G RAM), running MATLAB 2017b with a Windows 7
 405 operating system.

The experiments are divided into three subsections. The first is carried on the fire images, collected from various forest scenes; The second is on the fire videos, obtained from dead-directional or omni-directional cameras; In the third subsection, we show the performance of the proposed PreVM on more
 410 applications.

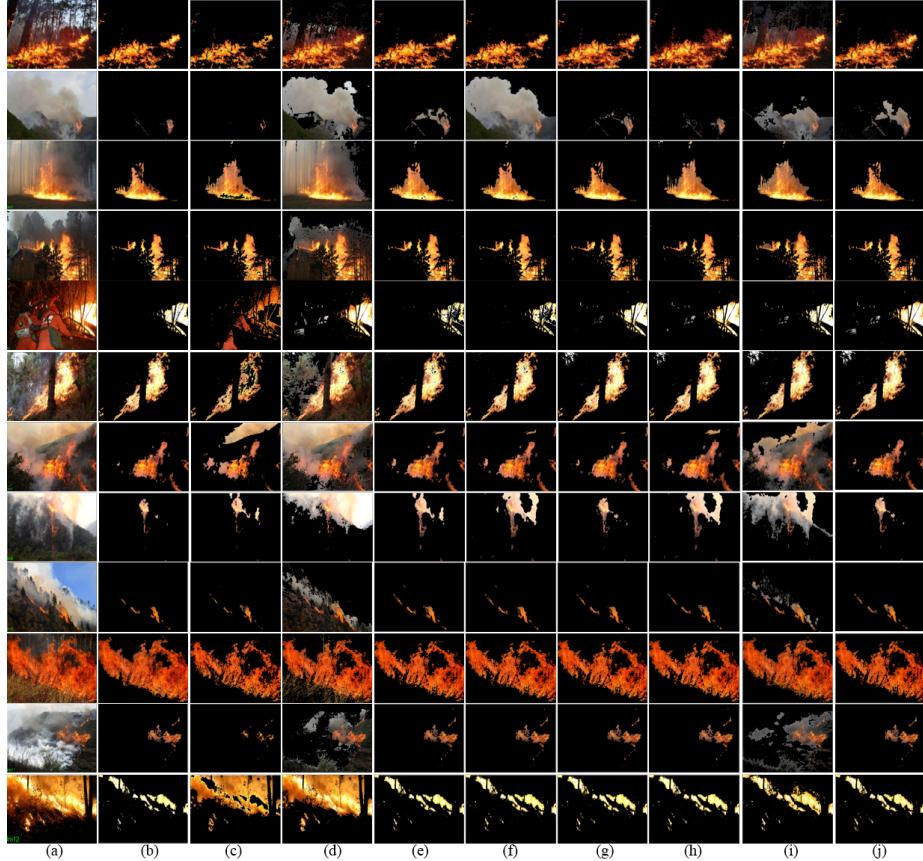


Figure 6: Visualization for the results of fire detection. The leftmost two columns are original images (a) and the annotated fire pixels (b). The rest are the results of detected fire pixels, obtained by YCbCr rules (c), K-medoids (d), L_1 -SVM (e), L_2 -SVM (f), L_1 -PreVM (g), L_2 -PreVM (h), One-class SVM (i) and 2C-SVM (j), respectively.

4.1. Forest fire images

The fire images, numbered by “Im1” to “Im12” respectively, are selected from various real forest environments, involving in the faked fire-like objects, close- or long-range fires, different light conditions, etc. On some high-resolution images, the superpixel technique SLIC[36] is also adopted, aiming at: 1) utilizing pixel neighborhood information, and 2) reducing the number of pixel-pattern samples for easy computation. If needed, SLIC is initialized by 3x3 or 5x5 blocks. Then a superpixel is labeled with the majority labels of pixels of the same block. For better visualization, a detected fire superpixel is also filled with the original pixels. To show the precise control, the parameter a is set to 98%. Experimental results are shown in Figure 6 for visualization and in Table 1 for the detailed numerical values, respectively.

Table 1: The comparison between our PreVMs and state-of-the-art.

Image m _x n	Training time/Testing time (seconds)							
	Training TP/Training FP (%)				Testing TP/Testing FP (%)			
	Rule	K-medoids	L_1 -SVM	L_2 -SVM	L_1 -PreVM	L_2 -PreVM	One-class	2C-SVM
Im1 300x400	-/0.312	-/18.39	0.429/0.003	173.384/0.004	1.013/0.025	422.656/0.040	203.190/0.143	426.068/0.075
	-/-	-/-	91.6/0.80	91.5/0.80	99.4/3.14	99.4/3.14	100.0/-	99.0/2.70
	64.6/0.31	99.0/39.00	87.6/1.35	87.5/1.34	98.8/1.35	98.8/1.56	99.5/51.63	96.6/5.18
Im2 542x900	-/2.323	-/37.443	0.424/0.016	161.945/0.023	1.066/0.016	366.384/0.016	132.450/0.527	361.341/0.332
	-/-	-/-	94.7/2.10	93.0/1.80	99.3/7.30	98.9/8.30	99.9/-	100.0/0.60
	17.1/0.00	92.9/27.09	89.4/8.40	91.7/10.18	98.4/8.52	97.5/6.50	99.9/27.85	97.8/15.38
Im3 375x500	-/0.817	-/2.903	1.331/0.006	331.124/0.008	1.399/0.006	524.784/0.006	128.120/0.201	402.140/0.082
	-/-	-/-	98.5/3.60	98.5/3.60	100.0/5.90	100.0/5.90	100.0/-	99.9/1.40
	86.8/9.88	98.4/41.15	95.2/3.33	95.2/3.33	100.0/3.16	100.0/4.61	99.9/10.24	99.1/5.62
Im4 575x950	-/2.581	-/42.191	1.663/0.006	377.277/0.008	1.364/0.611	227.232/0.006	127.437/0.573	320.784/0.118
	-/-	-/-	98.5/3.60	98.5/3.60	100.0/1.60	100.0/3.90	100.0/-	99.9/0.40
	97.3/4.25	100.0/15.72	98.2/3.33	98.2/3.33	100.0/3.85	100.0/3.61	100.0/6.30	99.4/4.69
Im5 514x900	-/2.218	-/27.559	1.101/0.015	214.347/0.015	1.342/0.013	496.426/0.017	270.784/1.541	390.678/0.242
	-/-	-/-	99.9/0.80	99.9/0.80	100.0/1.80	100.0/1.80	100.0/-	100.0/0.90
	12.3/13.31	94.8/7.74	66.6/0.30	66.6/0.30	99.8/2.06	99.3/2.44	84.4/1.45	98.8/4.58
Im6 963x1400	-/8.552	-/107.695	1.874/0.057	579.308/0.048	2.154/0.050	726.100/0.049	462.443/1.458	548.977/0.470
	-/-	-/-	99.9/1.20	99.9/1.20	99.9/1.90	99.9/1.90	100.0/-	100.0/1.50
	72.6/1.36	95.6/29.47	97.9/1.38	97.9/1.38	100.0/2.98	100.0/2.98	99.9/3.33	99.9/4.67
Im7 333x500	-/0.693	-/23.087	1.038/0.005	237.960/0.005	0.112/0.682	539.543/0.005	253.612/0.160	412.744/0.081
	-/-	-/-	98.1/2.10	98.1/2.10	100.0/0.50	100.0/5.70	100.0/-	100.0/1.00
	63.5/13.53	81.2/48.88	98.3/3.29	98.3/3.29	99.8/1.30	100.0/4.99	99.9/62.07	99.9/4.30
Im8 282x472	-/0.493	-/14.496	0.803/0.004	376.261/0.004	0.191/0.075	431.263/0.005	307.458/0.130	542.633/0.089
	-/-	-/-	97.2/4.00	97.2/4.00	99.9/0.80	100.0/9.70	99.9/-	99.8/1.90
	80.2/1.88	92.8/34.72	97.4/3.87	97.4/4.87	99.7/1.17	98.4/4.23	100.0/30.56	99.8/5.22
Im9 288x512	-/0.541	-/20.332	0.574/0.001	35.196/0.001	0.023/0.029	43.976/0.001	29.629/0.146	34.461/0.032
	-/-	-/-	100.0/0.60	100.0/0.60	100.0/0.50	100.0/0.70	99.9/-	100.0/0.30

	75.7/0.22	98.0/40.62	100.0/0.60	100.0/0.60	100.0/0.47	100.0/0.75	100.0/5.13	100.0/0.75
Im10 662x1000	-/3.809	-/30.563	1.842/0.064	535.130/1.004	1.042/0.352	532.881/1.004	275.229/0.709	316.089/0.349
	-/-	-/-	99.1/1.10	99.1/1.10	100.0/0.40	100.0/2.70	100.0/-	99.8/0.40
	78.0/4.19	98.3/18.86	98.8/1.68	98.8/1.68	99.9/1.15	100.0/3.32	99.7/19.40	99.4/2.63
Im11 300x450	-/0.476	-/42.216	0.348/0.001	23.110/0.001	0.026/0.001	27.274/0.001	28.165/0.145	29.509/0.038
	-/-	-/-	100.0/0.00	100.0/0.00	100.0/0.00	100.0/0.00	99.9/-	100.0/0.70
	22.3/0.00	93.4/28.23	100.0/0.33	100.0/0.33	100.0/0.33	100.0/0.33	100.0/28.65	100.0/1.43
Im12 180x300	-/0.212	-/0.650	0.622/0.000	26.809/0.000	0.028/0.000	22.912/0.000	28.650/0.132	29.623/0.016
	-/-	-/-	99.7/0.90	99.7/0.90	100.0/1.10	100.0/1.30	99.9/-	100.0/0.80
	19.2/35.45	99.9/36.94	99.4/0.68	99.4/0.68	99.9/1.01	99.9/1.44	99.8/5.15	100.0/1.25
Average	-/1.919	-/30.627	1.004/ 0.015	255.988/0.093	0.813 /0.155	363.453/0.096	187.264/0.489	317.921/0.16
	-/-	-/-	98.1/1.73	98.1/1.71	99.9/2.08	99.9/3.75	100.0 /-	99.9/1.05
	57.5/7.03	95.4/30.7	94.1/2.38	94.3/2.61	99.7/2.28	99.5/3.06	98.6/20.98	99.2/4.64

In our concerned fire detection rate and error warning rate, one-class SVM,
425 2C-SVM and our PreVMs are able to achieve more than 97.5% testing TPs, due
to paying special attention on the fire pixels. As expected, they significantly
outperform SVM and other methods. If considering error warning rate together,
it is no doubt that PreVMs win the best, then 2C-SVM follows, as shown in
Figs. 6 and 7. For easy reading, the testing numerical results in Table 1 are
430 also illustrated in Fig. 7. Because of equal attention on both classes, the
fire miss-detected phenomena frequently appeared in SVM, e.g., the results of
“Im1”, “Im3”, “Im5”, and “Im6”. In contrast to SVM, 2C-SVM can relieve
this problem moderately, but it also increases error warning rates, more or less.
On the data ”Im2”, it achieves a error warning rate at 15.38%, which is 5.2%
435 higher than that of L_2 -SVM. We should point out that 2C-SVM is originally
proposed for the unbalanced classification. However, for the fairness, here it
is also trained on the balanced two-class training samples. As a result, the
obtained decision plane is naturally biased to the negative class for the higher fire
440 detection rate. Similarly, one-class SVM also faces the over-high error warning
problem. Although without using non-fire supervisions for the model-training,
the obtained decision boundary may be too tight because one-class SVM aims
to find a minimum closed ”ball” to wrap the fire pixels inside. It seems unstable
for K-medoids and rule-reasoning based methods. For example, the fire detect
445 rates of rule-reasoning are varied from 12.3% to 97.3%. Specially, the results on
“Im5” and “Im12” show that almost all fire pixels are missed, with the testing

TPs at 12.3% and 19.2%, respectively. In contrast, two L_1 norm based machines need lower training time, running hundred times faster than L_2 norm versions. Additionally, L_1 -PreVM also shows its ability in error-correction, on “Im1” and “Im3” some error-annotated fire pixels are correctly discriminated to the fire.

The performance of pixel-precision fire positioning is easy to evaluate by visualization method. Fig. 6 shows that those positions where the detected fire pixels locate are visualized by the original pixel colors. Since the methods are all oriented to pixel classification, the fire pixel locations are easily achieved when the corresponding pixels are identified as the positive samples. Actually pixel-pattern samples of the testing image or video frame are rearranged in the order of pixel column priority, thus the indexes of samples can be used as pixel coordinates. Fig. 7 also shows the testing time in the right panel, in which two red dotted horizontal lines are used as the real-time reference lines, corresponding to the common acquisition frequency at 1/12 or 1/25 fps (frame per second), respectively. The results show that in most cases, the testing time of SVM and its variants is below the reference lines. It means that linear decision functions are more potential in pixel-precision real-time detection.

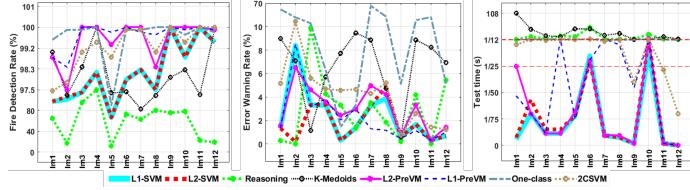


Figure 7: Numerical results on the testing fire detection rates (left), error warning rates (middle), and test time (right), respectively.

In this subsection, we aims to show the performance on different forest scenes. The main disadvantage is that the samples used for model-training and testing are drawn from the same image. In practical situations, the testing images should be unseen. Next the verification will be carried on forest fire videos, in which only one frame will be used for constructing training set, and the rest frames are for testing.

4.2. Forest fire videos

470 The data used in this subsection consists of total 16 videos, divided into two
groups: seven dead-directional and nine omni-directional videos. The first group
is from the public fire database Mivia, named fire3, ..., and fire11¹, as they
were named in the database. They had been standardized at the fixed resolution
256×300 at fps 15 (frames per second). The second is from Chinese CCTV
475 video news² and aerial imagery FLAME[37]³. Among the omni-directional
videos named Video1 ~ Video9, the first seven are fire clips drawn from CCTV
news, and the last two are from aerial drone (UAV) videos Zenmuse_X4S_1.mp4
and Zenmuse_X4S_2.mp4, respectively[35, 37]. For visualization, we report the
testing results on the discontinuous four frames, made up of the (f_r/i) -th
480 frame, $i = 4 \sim 1$, where f_r denotes the total of video frames. The testing
results are shown in Figs. 8 and 9 for visualization, and in Tabs. 2 and 3
for numerical values, respectively. In order to show the performance on long-
distance monitoring, some frames with the small proportion of fire area, or even
flameless, are used for testing, e.g., the test frames drawn from Video8 and
485 Video9. As shown in Fig. 9, the selected two flameless frames are boxed by
yellow rectangles. Likewise, the fire flames with small proportion of area are
highlighted by red circles. In the flameless case, the testing TP will be set
to 100%. And if non-fire pixels are misclassified, they will be reflected by the
testing FP, and shown with the original non-fire pixels.

490 The results on dead-directional videos are shown in Fig. 8 and Tab. 2.
Among the three methods that emphasize fire samples, one-class SVM is the
worst, achieving the averaged fire detection rate (testing TP) at 44.2%. This
means that it almost fails to detect fire from videos because the color of flame
is variable with the fire intensity. In contrast, 2C-SVM obtains the highest TP

¹Available at: <http://signal.ee.bilkent.edu.tr/VisiFire/Demo/FireClips/>.

²Available at: <https://github.com/xbyang1000/fire-pixel-annotation>.

³Available at: <https://ieee-dataport.org/open-access/flame-dataset-aerial-imagery-pile-burn-detection-using-drones-uavs>.

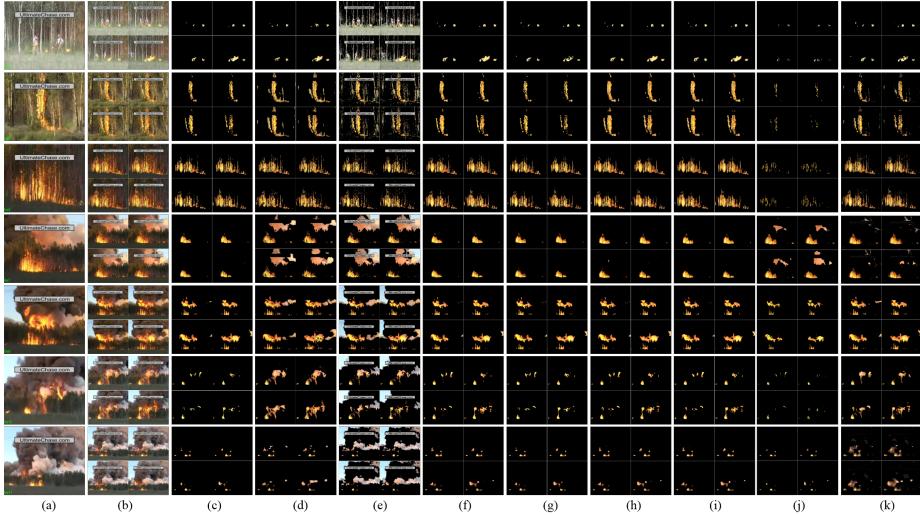


Figure 8: Fire detection results on the dead-directional videos. The leftmost image (a) is the first (training) frame of each video. The columns (b) and (c) show the selected 4 test frames and the fire pixel annotations, respectively. The testing results are shown in the rest columns, obtained by rule-reasoning (d), K-medoids (e), L1-SVM (f), L2-SVM (g), L1-PreVM (h), L2-PreVM (i), one-class SVM (j) and 2C-SVM (k), respectively.

value at 99.3%, but it also achieves the highest FP value at 3.67%. If considering both indicators together, PreVMs are still the best detectors. Especially for L_1 -PreVM, it achieves the averaged testing TP at 99.2% and testing FP at 1.09%. Generally, supervised detectors also outperform the unsupervised ones in both fire detection rates and error warning rates, as reported on the videos fire7, fire8 and fire10. In terms of real-time, Tab. 2 shows that with the fixed fps at 15, both rule and K-medoids based methods need 1.160 and 6.726 seconds (greater than $1/15 \approx 0.667$), respectively. While SVM and the variants have more potentials in real-time detection because of less testing time than $1/15$ seconds. As aforementioned, opposite to dead-directional monitoring, the omni-directional is more challenging in forest fire surveillance systems. Moreover, it is also closer to the practical situations. We continue to the verification on omni-directional fire videos, with the higher image resolutions and acquisition frequencies.

Table 2: Numerical value results on the dead-directional forest fire videos.

DeadVideo Frames	Training time/Testing time (seconds)							
	Rule	Testing TP/Testing FP (%)						
		K-medoids	L_1 -SVM	L_2 -SVM	L_1 -PreVM	L_2 -PreVM	One-class	2C-SVM
Fire3 260	-/1.138 91.1/5.63	-/8.045 100.0/44.46	0.017/0.002 99.6/0.40	0.054/0.002 97.6/0.40	0.018/0.002 99.6/0.40	0.058/0.002 99.6/0.40	0.052/0.002 31.8/0.00	0.066/0.002 95.9/0.15
Fire4 246	-/1.205 99.9/13.34	-/10.913 100.0/23.87	0.104/0.002 100.0/1.97	1.700/0.002 96.2/3.51	0.106/0.002 100.0/1.97	2.913/0.002 100.0/3.51	0.521/0.062 14.6/0.00	2.609/0.045 99.4/4.69
Fire5 208	-/1.173 96.7/18.10	-/6.666 85.2/33.63	0.107/0.002 99.7/1.93	2.319/0.002 98.0/2.32	0.113/0.002 99.7/2.12	3.208/0.002 98.0/2.32	2.632/0.002 14.9/0.00	2.672/0.003 100/6.34
Fire7 200	-/1.186 86.6/36.21	-/5.542 89.4/53.67	0.351/0.003 96.3/0.18	49.268/0.002 96.1/0.28	0.263/0.002 98.3/0.55	44.483/0.002 97.9/0.52	41.126/0.002 99.0/5.10	48.235/0.002 100/3.85
Fire8 245	-/1.137 98.1/23.99	-/4.086 96.1/42.22	0.071/0.004 95.8/0.10	1.001/0.003 95.0/0.16	0.079/0.003 97.4/0.08	1.242/0.004 95.0/0.16	0.521/0.025 52.7/0.00	0.550/0.022 99.9/0.62
Fire10 219	-/1.134 97.6/13.47	-/5.764 100.0/31.79	0.066/0.003 92.5/1.17	1.545/0.002 94.1/1.09	0.067/0.002 99.5/0.23	1.168/0.002 96.1/1.09	0.514/0.058 38.0/0.00	0.548/0.024 100.0/2.39
Fire11 216	-/1.147 94.3/28.69	-/6.067 86.4/40.09	0.060/0.002 99.8/2.71	0.716/0.002 94.9/1.78	0.060/0.002 99.8/2.27	0.722/0.002 99.8/2.31	0.580/0.024 58.6/0.06	0.580/0.036 100/7.67
Average	-/1.16	-/6.726	0.111/0.003	8.086/0.002	0.101/0.002	7.685/0.002	6.278/0.025	6.609/0.019
	94.9/19.92	93.9/38.53	97.7/1.21	96.0/1.36	99.2/ 1.09	98.1/1.47	44.2/0.74	99.3/3.67

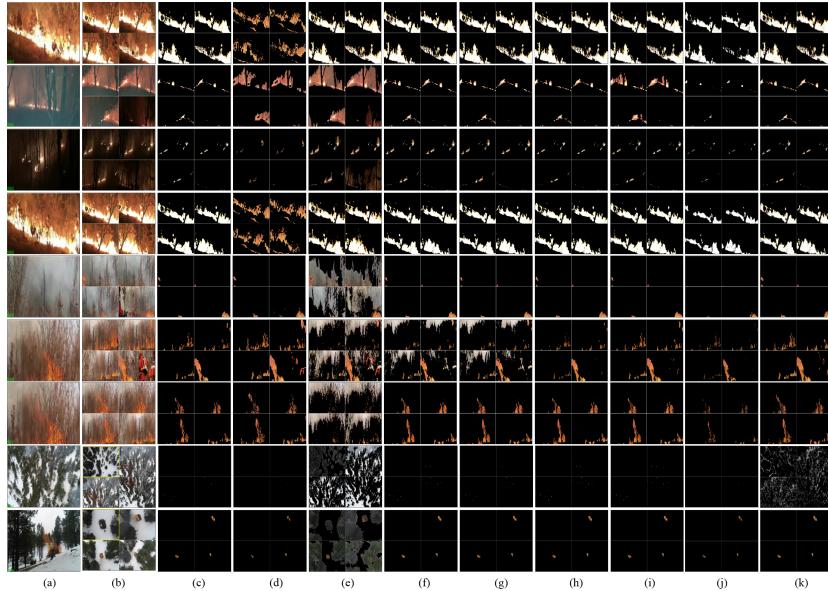


Figure 9: Fire detection results on the Omni-directional videos. The leftmost image (a) is the first (training) frame of each video. The columns (b) and (c) show the selected test frames and fire pixel annotations, respectively. The testing results are shown in the rest columns, obtained by rule-reasoning (d), K-medoids (e), L1-SVM (f), L2-SVM (g), L1-PreVM (h), L2-PreVM (i), one-class SVM (j) and 2C-SVM (k), respectively.

Table 3: Numerical value results on the omni-directional forest fire videos.

OmniVideo Frames/fps	Rule	Training time/Testing time (seconds)						
		K-medoids	Testing TP/Testing FP (%)					
			L_1 -SVM	L_2 -SVM	L_1 -PreVM	L_2 -PreVM	One-class	2C-SVM
Video1 278/25	-/2.841 20.0/80.97	-/18.326 72.1/0.21	0.487/0.008 98.2/0.18	120.540/0.008 98.2/0.23	0.645/0.008 100.0/0.12	151.899/0.008 100.0/0.23	48.732/0.006 68.9/0.00	84.822/0.006 100.0/0.81
Video2 291/25	-/2.841 69.7/33.08	-/10.212 100.0/54.40	0.115/0.008 100.0/3.68	3.115/0.009 100.0/3.52	0.121/0.008 100/3.19	4.307/0.008 100/4.42	4.209/0.007 21.1/0.00	6.107/0.008 100.0/3.37
Video3 251/25	-/2.760 87.0/19.4	-/22.213 100.0/37.13	0.105/0.008 100.0/0.50	1.529/0.008 100.0/0.36	0.115/0.009 100.0/0.12	1.832/0.008 100.0/0.16	1.529/0.008 23.4/0.00	1.582/0.008 100.0/1.89
Video4 101/25	-/2.938 21.13/33.32	-/28.933 100.0/4.37	0.338/0.008 95.2/0.03	64.557/0.008 95.2/0.04	0.365/0.008 99.4/0.03	88.964/0.008 99.4/0.04	64.110/0.007 62.2/0.00	82.605/0.008 99.0/1.50
Video5 176/25	-/2.843 96.9/3.58	-/19.559 98.0/35.41	0.396/0.010 98.0/0.03	29.324/0.008 98.0/0.03	0.408/0.008 98.5/0.03	31.568/0.008 98.0/0.04	29.752/0.013 27.9/0.00	31.001/0.007 99.7/0.73
Video6 376/25	-/2.884 99.4/4.42	-/17.573 95.1/43.53	0.722/0.008 94.5/14.78	113.733/0.008 94.5/14.49	0.578/0.008 98.5/1.78	102.147/0.008 98.9/2.49	95.644/0.008 35.2/0.00	101.310/0.008 93.6/12.74
Video7 151/25	-/2.885 97.4/4.56	-/16.519 87.1/38.74	0.150/0.001 94.8/1.05	2.449/0.001 94.7/1.01	0.124/0.001 98.8/1.05	26.622/0.001 98.8/1.01	11.162/0.002 35.8/0.00	11.058/0.007 97.1/0.62
Video8 966s/30	-/3.697 31.8/0.09	-/25.936 90.3/32.53	0.026/0.006 99.4/0.06	35.02/0.006 99.5/0.07	0.020/0.006 99.4/0.06	13.81/0.006 99.5/0.07	24.605/0.006 27.5/0.00	35.241/0.006 100.0/14.42
Video9 399s/30	-/3.726 75.3/0.000	-/31.934 61.2/32.03	0.179/0.006 100/0.01	95.22/0.746 97.2/0.02	0.011/0.006 100/0.01	13.87/0.030 100/0.02	16.614/0.072 54.2/0.00	25.614/0.025 99.3/0.01
Average	-/3.046 66.5/19.94	-/21.245 89.3/30.93	0.280/ 0.007 97.8/2.26	51.721/0.089 97.8/2.20	0.265/0.007 99.4/0.71	48.335/0.009 99.4/0.94	32.929/0.014 39.6/ 0.00	42.149/0.009 98.7/4.01

510

The results on omni-directional videos are shown in Figure 9 and Table 3.

In contrast to dead videos, the fire flames in the training frame may be different from those in the testing frames because of the changing forest scenes appeared in the same omni-video. Furthermore, as shown in the videos “Video2”, “Video3” and “Video8”, there may be multiple fire sites in the same frame. This difference makes one-class SVM even worse, and more fire pixels in the testing frames are miss-detected. Tab. 3 shows that, although no error warning is occurred, the testing TP values are even lower than those of unsupervised rule or K-medoids methods. For 2C-SVM, in most cases it continues to maintain the higher error warning rates in order to increase fire detection rates, in contrast to SVM. Especially on the data “Video6”, it achieves an error warning rate at 12.74%, but the testing TP value does not increase as expected. The reason may be that the obtained 2C-SVM is over-fitting, due to the excessive punishment on the positive samples in the training frame. Compared to the above, our PreVMs are stable in terms of high fire detection rates and low er-

ror warnings rates. They both achieve the average highest testing TP value of 99.4%. And at the same time, L_1 -PreVM achieves the average testing FP value of 0.71%, while L_2 -PreVM achieves the FP of 0.94%. In terms of testing time, the kernel-based machines with linear decision functions, e.g., SVM and the variants, are potential in pixel-precision real-time fire detection. Tab. 3 shows that their average testing time is far less than the given frame speed: 0.040 (1/25, with the fps at 25) or 0.033 (1/30) seconds. Compared with the L_2 norm machines, L_1 versions need less training time. Averagely, L_1 -norm models can run more than 180 times faster than L_2 ones, as Tab. 3 indicated. As for the fire positioning, the results are also visualized by fire pixel visualization, instead of tedious coordinates of fire pixels or regions.

In summary, in terms of the above-mentioned indicators, the comparisons show that PreVMs are superior to other detectors, especially for the L_1 -PreVM.

4.3. Other applications

Our PreVM can be also extended to other applications. For example, Fig. 10 shows the detected results of another two non-rigid interest objects: clouds and green plants. In satellite image processing, cloud detection is a challenging but fundamental task. If clouds can be well-detected, it will be very helpful for the next ground information extraction[38]. Green plant detection is related to vegetation coverage, an important evaluation indicator for forest ecology and desertification control[2]. The first three images are drawn from the Landsat-8 cloud imagery database with the resolution 1280*720[38]. The last three are our collected UAV images with the resolution 2400*2840, available at <https://pan.baidu.com/s/161xpXC4LYcj5EmESS5qkaA?pwd=9h2g>. For easy visualization, some regions are highlighted by the colored circles: the red circle is to show the detail preserving capability of interest objects, compared with other methods; and the blue one aims to show the detection performance, which even surpasses the given ground-truths. As shown in the blue-circled regions, some thin clouds or the green plants in the shadowed regions can also be well-detected by our PreVMs. For the limited space, more discussions will not be

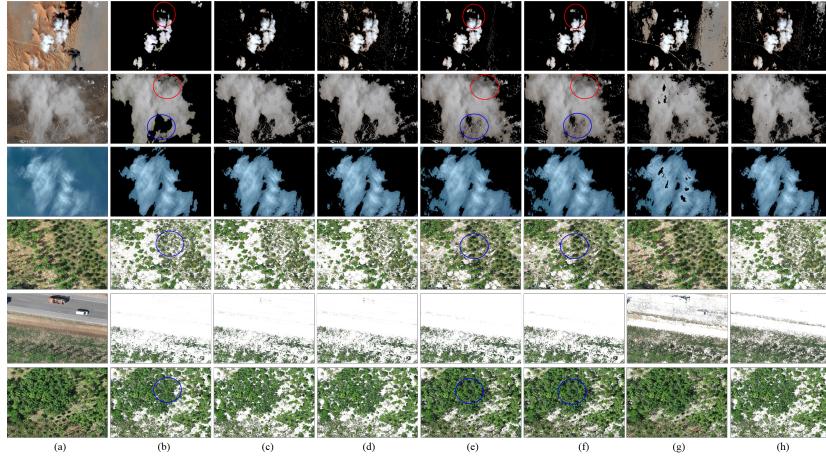


Figure 10: Visualization for cloud or green plant detection. The figures in the leftmost two columns show original images(a) and the ground-truths (b). The rest columns are the detected results, obtained by L1-SVM (c), L2-SVM (d), L1-PreVM (e), L2-PreVM (f), one-class SVM (g) and 2C-SVM (h), respectively.

continued here.

For the convenience of readers, high-resolution results (in the file named “visualization.pptx”), the related fire videos, and MATLAB codes for annotation are available at <https://github.com/xbyang1000/fire-pixel-annotation>.

560 5. CONCLUSION AND DISCUSSION

To distinguish the difference between fire miss-detection and error warning misclassification, we incorporate this prior knowledge into model-design, and propose our Preferred Vector Machine (PreVM). In terms of guaranteeing high fire detection rate, the L_0 norm constraint is introduced into the fire (positive) class. Simultaneously, in order to further reduce error warning rate, it is naturally extended to kernel-induced (nonlinear) L_1 -PreVM and L_2 -PreVM. Theoretically, we provide mathematical evidences for nonlinearizing L_1 -PreVM, derived from dual representations of the general L_p ($p \geq 1$) norm regularization problems. Experiments on forest fire images and videos show the superiorities of our proposals in high fire detection rate, low error warning rate, and real-time

570

fire detection. It must be remarked that this method can be easily extended into analogous applications in which the class of interest object should be paid more attention.

Due to the interpretability, this work follows the shallow-learning for the pixel-precision real-time fire detection. Considering the reality of the limit bandwidth and computing power on observation stations, especially, the limited fire data, we have no comparison with the existing deep learning methods. Different from the general rigid object recognition, the difficulties in fire detection are that: (1) Fire is non-rigid, shapeless, color-uncertainty, and it is made up of many complex substances. Until now, it is still unclear what features are conducive for fire flame extraction; (2) Public databases, especially associated with high-precision annotation, are unavailable; (3) A forest fire detector should be evaluated by multiple indicators, usually including fire detection rate, error warning rate, real-time and fire location (for fire spread trend prediction and fire rescue), rather than single indicator like test accuracy in the general object recognition; (4) It frequently suffers from unbalance classification problem, especially at the early fire stage. Additionally, in view of independent identically distributed (i.i.d.) hypothesis in machine learning, it is a little far-fetched for viewing fire detection as a binary classification. Actually, for example, it is unreasonable for the non-fire objects, composed of sky, ground, trees and plants, etc., to be viewed from the SAME class. However, if viewing it as multi-class classification to relax i.i.d. hypothesis, what should be the appropriate number of classes (categories) for a scene-changing forest environment? Additionally, for the large-scale classification task, SVM tends to be extremely cumbersome, e.g, intolerant time- or memory-consuming. To relieve this dilemma, many variants or called approximators have been proposed, such as LSSVM (least square SVM), TWSVM (TWin SVM), PSVM(Proximate SVM), and newly FSVC (fast SVM)[39]. In this paper, our method is oriented from pixel classification, and fire detection is expected to be accomplished in one-time scanning. In order to resist the aforesaid high-consuming, here we adopt superpixel technique to fit pixel classification. Additionally, for a given forest scene, sometimes fire infor-

mation is insufficient, or more extremely, it has never caught fire, how to learn a machine to match this scene? In human vision, forest smoke can be viewed as an early characteristic of a forest fire. For early warning, our potential avenue
605 for future research also involves in smoke detection [40] and the relation to fire.

6. ACKNOWLEDGMENT

This research was supported in part by Fundamental Research Funds for the Central Non-profit Research Institution of CAF under Grant. CAFYBB2022ZB002, and in part by the National Science Foundation of China under Grant 62072246.

610 APPENDIX A

A1. *The Proof to Theorem 1*

Proof. In addition to the previous symbols, let $\phi(S)^\perp$ be the orthogonal complement of $\phi(S)$, satisfying $\phi(S) \oplus \phi(S)^\perp = \phi(X)$ and $\phi(S) \cap \phi(S)^\perp = \emptyset$.

Since $\mathbf{w}^\phi \in \phi(X)$, there must exist a set of real values $\{\alpha_1, \dots, \alpha_l\}$ and a vector $\mathbf{v} \in \phi(S)^\perp$, such that \mathbf{w}^ϕ can be uniquely factorized into the following expression, by using Lemma 1 in the space $\phi(X)$.

$$\mathbf{w}^\phi = \sum_{i=1}^l \alpha_i \phi(\mathbf{x}_i) + \mathbf{v}, \quad \mathbf{v} \in \phi(S)^\perp. \quad (22)$$

For $\forall \mathbf{x} \in S$, because $\langle \phi(\mathbf{x}), \mathbf{v} \rangle = \mathbf{0}$, we have

$$h(\mathbf{x}) = \langle \mathbf{w}^\phi, \phi(\mathbf{x}) \rangle = \sum_{i=1}^l \alpha_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle. \quad (23)$$

Due to the additivity of inner products, (23) becomes,

$$h(\mathbf{x}) = \langle \mathbf{w}^\phi, \phi(\mathbf{x}) \rangle = \langle \sum_{i=1}^l \alpha_i \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle. \quad (24)$$

Eq. (24) holds for any $h \in F$ and $\mathbf{x} \in S$, i.e., $\mathbf{w}^\phi = \sum_{i=1}^l \alpha_i \phi(\mathbf{x}_i)$. ■

615 **A2. The Proof to Theorem 2**

Proof. In addition to the previous notations, let $\text{span}(\phi(S))$ denote the subspace spanned by feature samples $\{\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_l)\}$, and $\overline{\text{span}(\phi(S))}$ denote the closure of $\text{span}(\phi(S))$, respectively. If $\text{span}(\phi(S))$ is a closed set, $\overline{\text{span}(\phi(S))} = \text{span}(\phi(S))$ holds, otherwise let $\overline{\text{span}(\phi(S))} = \text{span}(\phi(S)) \cup \mathcal{B}(\text{span}(\phi(S)))$, where $\mathcal{B}(B)$ denotes the border of the set B . The following proof is given the closed space $\overline{\text{span}(\phi(S))}$, because the inner space $\text{span}(\phi(S))$ may be incomplete. Then a Cauchy sequence given in $\text{span}(\phi(S))$ may converge to the outer, e.g., the border of $\text{span}(\phi(S))$.

Take a L_p -norm ($1 \leq p \leq +\infty$) as the metric and use it over the closed subspace $\overline{\text{span}(\phi(S))}$, next we need to prove that the inner product space $(\overline{\text{span}(\phi(S))}, L_p)$ is also complete.

According to (17) and (18), i.e., Lemma 1 and Theorem 1, we have the following inner produce expression.

$$\langle \mathbf{w}^\phi, \phi(\mathbf{x}_0) \rangle = \sum_{i=1}^l \alpha_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_0) \rangle = \sum_{i=1}^l \alpha_i k(\mathbf{x}_i, \mathbf{x}_0). \quad (25)$$

For a fixed \mathbf{w}^ϕ and $\forall \phi(\mathbf{x}) \in \overline{\text{span}(\phi(S))}$, $\langle \mathbf{w}^\phi, \phi(\mathbf{x}) \rangle$ is a measurable value, which can be used to construct convergent Cauchy sequences. Without loss of generality, suppose that the needed sequence is represented as $\{\mathbf{y}_1, \mathbf{y}_2, \dots\}$, and converges to $\phi(\mathbf{x}_0)$, where $\mathbf{y}_i \in \overline{\text{span}(\phi(S))}$ for $\forall i$. Next we provide a strategy for the sequence construction.

Let \mathbf{y}_1 be an interior point drawn from $\text{span}(\phi(S))$, which satisfies $\mathbf{y}_1 = \sum_{j=1}^l \beta_j \phi(\mathbf{x}_j)$. Without loss of generality, for example, $\mathbf{y}_1 = \phi(\mathbf{x}_1)$, then by using Theorem 1, we have

$$\langle \mathbf{w}^\phi, \mathbf{y}_1 \rangle = \sum_{i=1}^l \alpha_i \langle \phi(\mathbf{x}_i), \mathbf{y}_1 \rangle = \sum_{i=1}^l \sum_{j=1}^l \alpha_i \bar{\beta}_j k(\mathbf{x}_i, \mathbf{x}_j), \quad (26)$$

where $\bar{\beta}$ denotes the conjugate of β .

Denote the value of (26) by r_1 , i.e., $r_1 = \langle \mathbf{w}^\phi, \mathbf{y}_1 \rangle$. Since \mathbf{y}_1 is an interior point, there must exist a δ -neighborhood $\delta(\mathbf{y}_1, \varepsilon_1)$, satisfying $\delta(\mathbf{y}_1, \varepsilon_1) \subset \overline{\text{span}(\phi(S))}$. Then \mathbf{y}_2 ($\mathbf{y}_2 \neq \mathbf{y}_1$) can be find out from $\delta(\mathbf{y}_1, \varepsilon_1)$, if $|r - \langle \mathbf{w}^\phi, \mathbf{y}_2 \rangle| <$

$|r - r_1|$. Likewise, denote it by $r_2 = \langle \mathbf{w}^\phi, \mathbf{y}_2 \rangle$. Repeat these steps, we can obtain sequences: $\{r_1, r_2, \dots\} (\subset R)$ and $\{\mathbf{y}_1, \mathbf{y}_2, \dots\} (\subset \overline{\text{span}(\phi(S))})$.

The obtained real sequence $\{|r - r_i|\}$ is monotonically bounded and converges to 0 because $|r - r_n| < |r - r_{n-1}| < \dots < |r - r_1|$. That is, $0 = \lim_{n \rightarrow \infty} |r - r_n| = |r - \lim_{n \rightarrow \infty} r_n|$. Thus we have $\lim_{n \rightarrow \infty} \langle \mathbf{w}^\phi, \mathbf{y}_n \rangle = r$, such that,

$$\langle \mathbf{w}^\phi, \lim_{n \rightarrow \infty} \mathbf{y}_n \rangle - r = \langle \mathbf{w}^\phi, \lim_{n \rightarrow \infty} \mathbf{y}_n - \phi(\mathbf{x}_0) \rangle = 0 \quad (27)$$

Note that \mathbf{w}^ϕ is independent of the sequence $\{\mathbf{y}_i\}$, Eq. (27) holds if and only if $\lim_{n \rightarrow \infty} \mathbf{y}_n = \phi(\mathbf{x}_0)$. For any given real number r , there must be a convergent Cauchy sequence $\{\mathbf{y}_i\}$, satisfying $\langle \mathbf{w}^\phi, \phi(\mathbf{x}_0) \rangle = r$. That is, the subspace $\overline{\text{span}(\phi(S))}$ is complete, thus it is also Hilbert. ■

References

References

- [1] S. Bao, N. Xiao, Z. Lai, et al., Optimizing watchtower locations for forest fire monitoring using location models, *Fire Safety Journal* 71 (2015) 100–109.
- [2] L. Ying, J. Han, Y. Du, et al., Forest fire characteristics in china: Spatial patterns and determinants with thresholds, *Forest Ecology and Management* 424 (2018) 345–354.
- [3] S. Zhang, D. Gao, H. Lin, et al., Wildfire detection using sound spectrum analysis based on the internet of things, *Sensors* 19 (23) (2019) 5039.
- [4] M. Hashemzadeh, A. Zademehd, Fire detection for video surveillance applications using ica k-medoids-based color model and efficient spatio-temporal visual features, *Expert Systems with Applications* 130 (2019) 60–78.
- [5] F. Bu, M. Gharajeh, Intelligent and vision-based fire detection systems: A survey, *Image and Vision Computing* 91 (2019) 103803.1–103803.15.

- [6] T. Çelik, H. Demirel, Fire detection in video sequences using a generic color model, *Fire Safety Journal* 44 (2) (2009) 147–158.
- [7] B. Yang, X. Guan, J. Wu, et al., Svms multi-class loss feedback based discriminative dictionary learning for image classification, *Pattern Recognition* 112 (2021) 107690.
- [8] H. Wang, Y. Feng, Y. Sa, et al., Pattern recognition and classification of two cancer cell lines by diffraction imaging at multiple pixel distances, *Pattern Recognition* 61 (2017) 234–244.
- [9] X. Han, J. Jin, M. Wang, et al., Video fire detection based on gaussian mixture model and multi-color features, *Signal Image and Video Processing* 11 (2017) 1419–1425.
- [10] P. Foggia, A. Saggese, M. Vento, Real-time fire detection for video-surveillance applications using a combination of experts based on color, shape, and motion, *IEEE Transactions on Circuits and Systems for Video Technology* 25 (9) (2015) 1545–1556.
- [11] B. Ko, K. Cheong, J. Nam, Fire detection based on vision sensor and support vector machines, *Fire Safety Journal* 44 (3) (2009) 322–329.
- [12] W. Qureshi, M. Ekpanyapong, M. Dailey, et al., Quickblaze: Early fire detection using a combined video processing approach, *Fire Technology* 52 (5) (2016) 1293–1317.
- [13] S. Geetha, C. Abhishek, C. Akshayanat, Machine vision based fire detection techniques: a survey, *Fire Technology* 57 (2) (2021) 591–623.
- [14] L. Huang, G. Liu, Y. Wang, et al., Fire detection in video surveillances using convolutional neural networks and wavelet transform, *Engineering Applications of Artificial Intelligence* 110 (2022) 104737.
- [15] P. Wang, J. Zhang, H. Zhu, Fire detection in video surveillance using superpixel-based region proposal and ese-shufflenet, *Multimedia Tools and Applications* (2021) 1–28.

- 685 [16] R. Zhang, W. Zhang, Y. Liu, et al., An efficient deep neural network with
color-weighted loss for fire detection, *Multimedia Tools and Applications*
(2022) 1–19.
- 690 [17] D. Bui, N. Hoang, P. Samui, Spatial pattern analysis and prediction of
forest fire using new machine learning approach of multivariate adaptive
regression splines and differential flower pollination optimization: A case
study at lao cai province (viet nam), *Journal of Environmental Management* 237 (2019) 476–487.
- 695 [18] A. Khatami, S. Mirghasemi, A. Khosravi, A new pso-based approach to
fire flame detection using k-medoids clustering, *Expert Systems with Ap-*
plications 68 (2017) 69–80.
- 700 [19] R. Zhang, S. Lu, H. Yu, Recognition method of cement rotary kiln burning
state based on otsu-kmeans flame image segmentation and svm, *Optik* 243
(2021) 167418.
- [20] X. Tang, T. Machimura, J. Li, A novel optimized repeatedly random un-
dersampling for selecting negative samples: A case study in an svm-based
forest fire susceptibility assessment, *Journal of Environmental Management*
271 (2020) 111014.
- 705 [21] J. Su, D. V. Vargas, K. Sakurai, One pixel attack for fooling deep neural
networks, *IEEE Transactions on Evolutionary Computation* 23 (5) (2019)
828–841.
- [22] O. U. Lenz, D. Peralta, C. Cornelis, Average localised proximity: A new
data descriptor with good default one-class classification performance, *Pat-
tern Recognition* 118 (2021) 107991.
- 710 [23] D. de la Mata-Moya, M. P. Jarabo-Amores, et al., Approximating the
neyman–pearson detector with 2c-svms. application to radar detection, *Sig-
nal Processing* 131 (2017) 364–375.

- [24] A. Iranmehr, H. Masnadi-Shirazi, N. Vasconcelos, Cost-sensitive support vector machines, *Neurocomputing* 343 (2019) 50–64.
- [25] C. Igel, No free lunch theorems: Limitations and perspectives of metaheuristics, in: Theory and principled methods for the design of metaheuristics, Springer, 2014, pp. 1–23.
- [26] F. Zhang, P. Zhao, S. Xu, Integrating multiple factors to optimize watchtower deployment for wildfire detection, *Science of The Total Environment* 737 (2020) 139561.
- [27] L. Chen, H. Zhang, Statistical margin error bounds for l1-norm support vector machines, *Neurocomputing* 339 (2019) 210–216.
- [28] A. Smg, A Ckh, A. Tjh, et al., Effective zero-norm minimization algorithms for noisy compressed sensing, *Journal of the Franklin Institute* 357 (11) (2020) 7159–7187.
- [29] S. Rosset, G. Swirszcz, N. Srebro, et al., L1 regularization in infinite dimensional feature spaces, in: N. Bshouty, C. Gentile (Eds.), *Learning Theory, COLT 2007*, Vol. 4539, Springer Berlin Heidelberg, 2007, pp. 544–558.
- [30] A. Bordes, S. Ertekin, J. Weston, Fast kernel classifiers with online and active learning, *Journal of Machine Learning Research* 6 (54) (2005) 1579–1619.
- [31] R. Yin, Y. Liu, W. Wang, Sketch kernel ridge regression using circulant matrix: Algorithm and theory, *IEEE Transactions on Neural Networks and Learning Systems* 31 (9) (2020) 3512–3524.
- [32] B. Schölkopf, R. Herbrich, A. J. Smola, A generalized representer theorem, in: In Proceedings of the Annual Conference on Computational Learning Theory, 2001, pp. 416–426.
- [33] W. Zheng, Z. Lin, H. Wang, L1-norm kernel discriminant analysis via bayes error bound optimization for robust feature extraction, *IEEE Transactions on Neural Networks and Learning Systems* 25 (4) (2014) 793–805.

- 740 [34] S. Marukatat, Kernel matrix decomposition via empirical kernel map, Pattern Recognition Letters 77 (2016) 50–57.
- [35] X. Yang, R. Chen, F. Zhang, et al., Pixel-level automatic annotation for forest fire image, Engineering Applications of Artificial Intelligence 104 (2021) 104353.
- 745 [36] R. Achanta, A. Shaji, K. Smith, et al., Slic superpixels compared to state-of-the-art superpixel methods, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (11) (2012) 2274–2282.
- [37] A. As, A. Fa, A. Ar, et al, Aerial imagery pile burn detection using deep learning: The flame dataset, Computer Networks 193 (5) (2021) 108001.
- 750 [38] L. Zhang, J. Sun, X. Yang, et al., Improving deep learning-based cloud detection for satellite images with attention mechanism, IEEE Geoscience and Remote Sensing Letters 19 (2022) 1–5. doi:10.1109/LGRS.2021.3133872.
- [39] Z. Ali Hammouri, M. Delgado, E. Cernadas, et al., Fast svc for large-scale classification problems, IEEE Transactions on Pattern Analysis and Machine Intelligence (2021) 1–1doi:10.1109/TPAMI.2021.3085969.
- 755 [40] Y. Hu, J. Zhan, G. Zhou, et al., Fast forest fire smoke detection using mvmnet, Knowledge-Based Systems 241 (2022) 108219.