# Video Event Detection Using Kernel Support Vector Machine with Isotropic Gaussian Sample Uncertainty (KSVM-iGSU)

Christos Tzelepis[1,2(✉)], Vasileios Mezaris[1], and Ioannis Patras[2]

[1] Information Technologies Institute (ITI), CERTH, 57001 Thermi, Greece
{tzelepis,bmezaris}@iti.gr
[2] Queen Mary University of London, Mile End Campus, London E14NS, UK
i.patras@qmul.ac.uk

**Abstract.** In this paper, we propose an algorithm that learns from uncertain data and exploits related videos for the problem of event detection; related videos are those that are closely associated, though not fully depicting the event of interest. In particular, two extensions of the linear SVM with Gaussian Sample Uncertainty are presented, which (a) lead to non-linear decision boundaries and (b) incorporate related class samples in the optimization problem. The resulting learning methods are especially useful in problems where only a limited number of positive and related training observations are provided, e.g., for the 10Ex subtask of TRECVID MED, where only ten positive and five related samples are provided for the training of a complex event detector. Experimental results on the TRECVID MED 2014 dataset verify the effectiveness of the proposed methods.

**Keywords:** Video event detection · Very few positive samples · Related samples · Learning with uncertainty · Kernel methods · Relevance degree SVMs

## 1 Introduction

High-level video event detection is concerned with determining whether a certain video depicts a given event or not. Typically, a high-level (or complex) event is defined as an interaction among humans, or between humans and physical objects [16]. Some typical examples of complex events are those provided in the Multimedia Event Detection (MED) task of the TRECVID benchmarking activity [22]. For instance, indicative complex events defined in MED 2014 include "Attempting a bike trick", "Cleaning an appliance", or "Beekeeping", to name a few.

There are numerous challenges associated with building effective video event detectors. One of them is that often there is only a limited number of positive video examples available for training. Another challenge is that video representation techniques usually introduce uncertainty in the input that is fed to


Q. Tian et al. (Eds.): MMM 2016, Part I, LNCS 9516, pp. 3–15, 2016.
DOI: 10.1007/978-3-319-27671-7_1

the classifiers, and this also needs to be taken into consideration during classifier training. In this work we deal with the problem of learning video event detectors when a limited number of positive and related (i.e., videos that are closely related with the event, but do not meet the exact requirements for being a positive event instance [22]) event videos are provided. For this, we exploit the uncertainty of the training videos by extending the linear Support Vector Machine with Gaussian Sample Uncertainty (LSVM-GSU), presented in [27], in order to arrive at non-linear decision functions. Specifically, we extend this version of LSVM-GSU that assumes isotropic uncertainty (hereafter denoted LSVM-iGSU) into a new kernel-based algorithm, which we call KSVM-iGSU. We also further extend KSVM-iGSU, drawing inspiration from the Relevance Degree kernel SVM (RD-KSVM) proposed in [28], such that related samples can be effectively exploited as positive or negative examples with automatic weighting. We refer to this algorithm as RD-KSVM-iGSU. We show that the RD-KSVM-iGSU algorithm results in more accurate event detectors than the state of the art techniques used in related works, such as the standard kernel SVM and RD-KSVM.

The paper is organized as follows. In Sect. 2 we review related work. In Sect. 3 the two proposed SVM extensions are presented. Video event detection results, by application of the proposed KSVM-iGSU and RD-KSVM-iGSU to the TRECVID MED 2014 dataset, are provided in Sect. 4, while conclusions are drawn and future work is discussed in Sect. 5.

## 2 Related Work

There are many works dealing with event detection in video (e.g. [2,5,7,9,11–15,19,21]), several of them being in the context of the TRECVID MED task. Despite the attention that video event detection has received, though, there is only a limited number of studies that have explicitly examined the problem of learning event detectors from very few (e.g. 10) positive training examples [13,28], and developed methods for addressing this exact problem. In [13], for instance, the authors present VideoStory, a video representation scheme for learning event detectors from a few training examples by exploiting freely available Web videos together with their textual descriptions. Several other works (e.g. [2]) treat the few-example problem in the same way that they deal with event detection when more examples are available (e.g. training standard kernel SVMs). Learning video event detectors from a few examples is a problem that is simulated in the TRECVID MED task [22] by the 10Ex subtask, where only 10 positive samples are available for training.

In the case of learning from very few positive samples, it is of high interest to further exploit video samples that do not exactly meet the requirements for being characterized as true positive examples of an event, but nevertheless are closely related to an event class and can be seen as "related" examples of it. This is simulated in the TRECVID MED task [22] by the "near-miss" video examples provided for each target event class. Except for [28], none of the above works

takes full advantage of these related videos for learning from few positive samples; instead, the "related" samples are either excluded from the training procedure [2,11], or they are mistreated as true positive or true negative instances [7]. In contrast, in [28] the authors exploit related samples by handling them as weighted positive or negative ones, applying an automatic weighting technique during the training stage. To this end, a relevance degree in $(0, 1]$ is automatically assigned to all the related samples, indicating the degree of relevance of these observations with the class they are related to. It was shown that this weighting resulted in learning more accurate event detectors.

Regardless of whether the above works address the problem of learning from a few positive examples or assume that an abundance of such examples is available, they all treat the training video representations as noise-free observations in the SVM input space. Looking beyond the event detection applications, though, assuming uncertainty in input under the SVM paradigm is not unusual and has been shown to lead to better learning. Lanckriet et al. [18] considered a binary classification problem where the mean and covariance matrix of each class are assumed to be known. Xu et al. [29,30] considered the robust classification problem for a class of non-box-typed uncertainty sets, in contrast to [1,18,25], who robustified regularized classification using box-type uncertainty. Finally, in [27], Tzelepis et al. proposed a linear maximum-margin classifier, called SVM with Gaussian Sample Uncertainty, dealing with uncertain input data. The uncertainty in [27] can be modeled either isotropically or anisotropically, arriving at a convex optimization problem that is solved using a gradient descent approach.

To the best of our knowledge, there has been no study dealing with uncertainty in the video event detection problem, except for [27]. However, [27] introduces linear classifiers, which in the event detection problem are not expected to perform in par with traditional kernel SVMs that are typically used (e.g. [11,31]), despite the advantages of considering data uncertainty in the learning process. In this work, we extend the above study and kernelize the LSVM-iGSU of [27], under the assumption of isotropic sample uncertainty. We apply the resulting KSVM-iGSU to the event detection problem when only a few positive samples are available for training. Moreover, we propose a further extension of KSVM-iGSU, namely Relevance Degree KSVM-iGSU (RD-KSVM-iGSU), inspired by [28], such that related samples can also be exploited as weighted positive or negative ones, based on an automatic weighting scheme.

## 3 Kernel SVM-iGSU

### 3.1 Overview of LSVM-iGSU

LSVM-iGSU [27] is a classifier that takes a input training data that are described not solely by a set of feature representations, i.e. a set of vectors $\mathbf{x}_i$ in some $n$-dimensional space, but rather by a set of multivariate isotropic Gaussian distributions which model the uncertainty of each training example. That is, every

training datum is characterized by a mean vector $\mathbf{x}_i \in \mathbb{R}^n$ and an isotropic covariance matrix, i.e. a scalar multiple of the identity matrix, $\Sigma_i = \sigma_i^2 I_n \in \mathbb{S}_{++}^n$[1]. LSVM-iGSU is obtained by minimizing, with respect to $\mathbf{w}$, $b$, the objective function $\mathcal{J}\colon \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ given by

$$\mathcal{J}(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{l} \mathcal{L}(\mathbf{w}, b, \mathbf{x}_i, \sigma_i^2 I_n, y_i), \tag{1}$$

where $l$ is the number of training data, $\mathbf{w}\cdot\mathbf{x} + b = 0$ denotes the separating hyperplane, and the loss $\mathcal{L}\colon (\mathbb{R}^n \times \mathbb{R}) \times (\mathbb{R}^n \times \mathbb{S}_{++}^n \times \{\pm 1\}) \to \mathbb{R}$ is given by

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \mathbf{x}_i, \sigma_i^2 I_n, y_i) = {} & \frac{y_i - \mathbf{w}\cdot\mathbf{x}_i - b}{2}\left(\operatorname{erf}\left(\frac{y_i - \mathbf{w}\cdot\mathbf{x}_i - b}{\sqrt{2\sigma_i^2\|\mathbf{w}\|_2^2}}\right) + y_i\right) \\ & + \frac{\sqrt{\sigma_i^2\|\mathbf{w}\|_2^2}}{\sqrt{2\pi}} \exp\left(-\frac{(y_i - \mathbf{w}\cdot\mathbf{x}_i - b)^2}{2\sigma_i^2\|\mathbf{w}\|_2^2}\right),\end{aligned} \tag{2}$$

where $\mathbf{x}_i$ and $\sigma_i^2 I_n$ denote the mean vector and the covariance matrix of the $i$-th input entity (Gaussian distribution), respectively, $y_i$ denotes its ground-truth label, and $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,\mathrm{d}t$ denotes the error function.

As discussed in [27], (1) is convex and thus a (global) optimal solution $(\mathbf{w}, b)$ can be obtained using a gradient descent algorithm. The resulting (linear) decision function $f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} + b$ is used in the testing phase for classifying an unseen sample similarly to the standard linear SVM algorithm [4]; that is, according to the distance between the testing sample and the separating hyperplane, without taking into account any uncertainty estimates that could be made for the testing sample representation.

### 3.2 Kernelizing LSVM-iGSU (KSVM-iGSU)

The optimization problem discussed in the previous section can be recasted as a variational calculus problem of finding the function $f$ that minimizes the functional $\Phi[f]$:

$$\min_{f\in\mathcal{H}} \Phi[f], \tag{3}$$

where the functional $\Phi[f]$ is given by

$$\begin{aligned}\Phi[f] = {} & \frac{1}{2}\lambda\|f\|_{\mathcal{H}}^2 + \sum_{i=1}^{l}\left[\frac{y_i - f(\mathbf{x}_i) - b}{2}\left(\operatorname{erf}\left(\frac{y_i - f(\mathbf{x}_i) - b}{\sqrt{2\sigma_i^2\|f\|_{\mathcal{H}}^2}}\right) + y_i\right)\right. \\ & \left. + \frac{\sqrt{\sigma_i^2\|f\|_{\mathcal{H}}^2}}{\sqrt{2\pi}} \exp\left(\frac{(y_i - f(\mathbf{x}_i) - b)^2}{2\sigma_i^2\|f\|_{\mathcal{H}}^2}\right)\right],\end{aligned} \tag{4}$$

[1] $\mathbb{S}_{++}^n$ denotes the convex cone of all symmetric positive definite $n \times n$ matrices with entries in $\mathbb{R}$. $I_n$ denotes the identity matrix of order $n$.

where $\lambda = 1/C$ is a regularization parameter and $f$ belongs to a Reproducing Kernel Hilbert Space (RKHS), $\mathcal{H}$, with associated kernel $k$. Using a generalized semi-parametric version [24] of the representer theorem [17], it can be shown that the minimizer of the above functional admits a solution of the form

$$f(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i k(\mathbf{x}, \mathbf{x}_i) - b, \tag{5}$$

where $b \in \mathbb{R}$, $\alpha_i \in \mathbb{R}$, $\forall i$.

Using the reproducing property, we have

$$\|f\|_{\mathcal{H}}^2 = \langle f, f \rangle_{\mathcal{H}} = \left\langle \sum_{i=1}^{l} \alpha_i k(\cdot, \mathbf{x}_i), \sum_{j=1}^{l} \alpha_j k(\cdot, \mathbf{x}_j) \right\rangle_{\mathcal{H}} = \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}, \tag{6}$$

where $K$ is the kernel matrix, i.e. the symmetric positive definite $l \times l$ matrix defined as $K = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^{l}$, and $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_l)^\top$. Moreover, we observe that $f(\mathbf{x}_i) = \sum_{j=1}^{l} \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{K}_i \cdot \boldsymbol{\alpha}$, where $\mathbf{K}_i$ denotes the $i$-th column of the kernel matrix $K$. Then, the objective function $\mathcal{J}_{\mathcal{H}} \colon \mathbb{R}^l \times \mathbb{R} \to \mathbb{R}$ is given by

$$\begin{aligned} \mathcal{J}_{\mathcal{H}}(\boldsymbol{\alpha}, b) = \; & \frac{1}{2}\lambda \boldsymbol{\alpha}^\top K \boldsymbol{\alpha} + \sum_{i=1}^{l} \left[ \frac{y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b}{2} \left( \operatorname{erf}\left( \frac{y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b}{\sqrt{2\sigma_i^2 \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}}} \right) + y_i \right) \right. \\ & \left. + \frac{\sqrt{\sigma_i^2 \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}}}{\sqrt{2\pi}} \exp\left( - \frac{(y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b)^2}{2\sigma_i^2 \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}} \right) \right], \end{aligned} \tag{7}$$

where the above sum gives the total loss. We (jointly) minimize the above convex[2] objective function with respect to $\boldsymbol{\alpha}$, $b$ similarly to [27] using the Limited-memory BFGS (L-BFGS) algorithm [20]. L-BFGS is a quasi-Newton optimization algorithm that approximates the Broyden-Fletcher-Goldfarb-Shanno (BFGS) [3] algorithm using a limited amount of computer memory. It requires the first order derivatives of the objective function with respect to the optimization variables $\boldsymbol{\alpha}$, $b$. They are given[3], respectively, as follows

$$\begin{aligned} \frac{\partial \mathcal{J}_{\mathcal{H}}}{\partial \boldsymbol{\alpha}} = \; & \lambda K \boldsymbol{\alpha} + \sum_{i=1}^{l} \left[ \frac{\sigma_i^2 \exp\left( - \frac{(y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b)^2}{2\sigma_i^2 \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}} \right)}{\sqrt{2\pi\sigma_i^2 \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}}} K \boldsymbol{\alpha} \right. \\ & \left. - \frac{1}{2} \operatorname{erf}\left( \frac{y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b}{\sqrt{2\sigma_i^2 \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}}} \right) \mathbf{K}_i - \frac{y_i}{2} \mathbf{K}_i \right], \end{aligned} \tag{8}$$

and

$$\frac{\partial \mathcal{J}_{\mathcal{H}}}{\partial b} = -\frac{1}{2} \sum_{i=1}^{l} \left[ \operatorname{erf}\left( \frac{y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b}{\sqrt{2\sigma_i^2 \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}}} \right) + y_i \right]. \tag{9}$$

[2] Convexity can be shown using Theorem 2 proved in [27].

[3] Their derivation is omitted, as it is technical but straightforward.

Since $J$ is a convex function on $\mathbb{R}^l \times \mathbb{R}$, L-BFGS leads to a global optimal solution; that is, at a pair $(\boldsymbol{\alpha}, b)$ such that the decision function given in the form of (5) minimizes the functional (4). We call this classifier kernel SVM-iGSU (KSVM-iGSU).

### 3.3 Relevance Degree KSVM-iGSU

Motivated by [28], we reformulate the optimization problem in (3)-(4) such that a different penalty parameter $c_i \in (0, 1]$ (hereafter called as relevance degree) is introduced to each input datum. That is, the functional $\Phi[f]$ of (4) is now given by

$$\Phi[f] = \frac{1}{2}\lambda\|f\|_{\mathcal{H}}^2 + \sum_{i=1}^{l} c_i \left[ \frac{y_i - f(\mathbf{x}_i) - b}{2} \left( \operatorname{erf}\left( \frac{y_i - f(\mathbf{x}_i) - b}{\sqrt{2\sigma_i^2\|f\|_{\mathcal{H}}^2}} \right) + y_i \right) + \frac{\sqrt{\sigma_i^2\|f\|_{\mathcal{H}}^2}}{\sqrt{2\pi}} \exp\left( \frac{(y_i - f(\mathbf{x}_i) - b)^2}{2\sigma_i^2\|f\|_{\mathcal{H}}^2} \right) \right]. \tag{10}$$

To solve $\min_{f\in\mathcal{H}} \Phi[f]$, following a similar path as in the Sect. 3.2, we arrive at the following convex objective function

$$\mathcal{J}_{\mathcal{H}}(\boldsymbol{\alpha}, b) = \frac{1}{2}\lambda\boldsymbol{\alpha}^\top K\boldsymbol{\alpha} + \sum_{i=1}^{l} c_i \left[ \frac{y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b}{2} \left( \operatorname{erf}\left( \frac{y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b}{\sqrt{2\sigma_i^2\boldsymbol{\alpha}^\top K\boldsymbol{\alpha}}} \right) + y_i \right) + \frac{\sqrt{\sigma_i^2\boldsymbol{\alpha}^\top K\boldsymbol{\alpha}}}{\sqrt{2\pi}} \exp\left( -\frac{(y_i - \mathbf{K}_i \cdot \boldsymbol{\alpha} - b)^2}{2\sigma_i^2\boldsymbol{\alpha}^\top K\boldsymbol{\alpha}} \right) \right], \tag{11}$$

which we again minimize using L-BFGS. The (global) optimal solution $(\boldsymbol{\alpha}, b)$ determines the decision function given in the form of (5). The new extension of KSVM-iGSU obtained in this way is hereafter referred to as a Relevance Degree KSVM-iGSU (RD-KSVM-iGSU).

Furthermore, following the approach presented in [28], we solely assign a single relevance degree $c \in (0, 1]$ only to related samples, keeping the relevance degrees for the rest of the training set equal to 1. The above training parameter needs to be optimized, using a cross-validation procedure.

## 4 Experiments and Results

### 4.1 Dataset and Evaluation Measures

The proposed algorithms are applied in the problem of video event detection and are tested on a subset of the large-scale video dataset of the TRECVID Multimedia Event Detection (MED) 2014 benchmarking activity [22]. Similarly to [27], we use only the training portion of the TRECVID MED 2014 task dataset, which provides ground-truth information for 30 complex event classes,

since for the corresponding evaluation set of the original TRECVID task there is no ground-truth data available. Hereafter, we refer to the aforementioned ground-truth-annotated dataset as MED14 and we divide it into a training subset, consisting of 50 positive and 25 related (near-miss) samples per event class, together with 2496 background samples (i.e. videos that are negative examples for all the event classes), and an evaluation subset consisting of approximately 50 positive and 25 related samples per event class, along with another 2496 background samples.

For assessing the detection performance of each trained event detector, the average precision (AP) [23] measure is utilized, while for measuring the detection performance of a classifier across all the event classes we use the mean average precision (MAP), as is typically the case in the video event detection literature, e.g. [8,22,28].

### 4.2 Video Representation and Uncertainty

For video representation, 2 keyframes per second are extracted at regular time intervals from each video. Each keyframe is represented using the last hidden layer of a pre-trained Deep Convolutional Neural Network (DCNN). More specifically, a 16-layer pre-trained deep ConvNet network provided in [26] is used. This network had been trained on the ImageNet data [6], providing scores for 1000 ImageNet concepts; thus, each keyframe has a 1000-element vector representation. Then, the typical procedure followed in state of the art event detection systems includes the computation of a video-level representation for each video by taking the average of the corresponding keyframe-level representations [2,5,11,31].

In contrast to the existing event detection literature, in the case of RD-SVM-iGSU (or also KSVM-iGSU and the original LSVM-iGSU), the aforementioned keyframe-level video representations can be seen as observations of the input Gaussian distributions that describe the training videos. That is, let $\mathcal{X}$ be a set of $l$ annotated random vectors representing the aforementioned video-level model vectors. We assume that each random vector is distributed normally; i.e., for the random vector representing the $i$-th video, $\mathbf{X}_i$, we have $\mathbf{X}_i \sim \mathcal{N}(\mathbf{x}_i, \Sigma_i)$. Also, for each random vector $\mathbf{X}_i$, a number, $N_i$, of observations, $\{\mathbf{x}_i^t \in \mathbb{R}^n : t = 1, \ldots, N_i\}$ is available; these are the keyframe-level model vectors that have been computed. Then, the mean vector and the covariance matrix of $\mathbf{X}_i$ are computed respectively as follows

$$\mathbf{x}_i = \frac{1}{N_i} \sum_{t=1}^{N_i} \mathbf{x}_i^t, \quad \Sigma_i = \sum_{t=1}^{N_i} (\mathbf{x}_i^t - \mathbf{x}_i)(\mathbf{x}_i^t - \mathbf{x}_i)^\top. \tag{12}$$

Now, due to the assumption for isotropic covariance matrices, we approximate the above covariance matrices as multiples of the identity matrix, i.e. $\widehat{\Sigma_i} = \sigma_i^2 I_n$ by minimizing the squared Frobenious norm of the difference $\Sigma_i - \widehat{\Sigma_i}$ with respect to $\sigma_i^2$. It can be shown (by using simple matrix algebra [10]) that for this it suffices to set $\sigma_i^2$ equal to the mean value of the elements of the main diagonal of $\Sigma_i$.

### 4.3 Experimental Results and Discussion

The proposed kernel extensions of LSVM-iGSU [27] (KSVM-iGSU, RD-KSVM-iGSU) are tested on the MED14 dataset, and compared to standard kernel SVM (KSVM), LSVM-iGSU [27] and RD-KSVM [28]. We note here that for the problem of video event detection (and especially when only a few positive training samples are available), kernel SVM is the state-of-the-art approach [2,5], while, when also a few related samples are available, RD-KSVM leads to state-of-the-art detection performance [28]. We experimented on the problem of learning from 10 positive examples per each event class, together with 5 related samples, that are drawn from the set of 25 related samples provided for each event class following the method presented in [28]; i.e., the 5 nearest to the median of all 25 related samples were kept for training both RD-KSVM and RD-SVM-iGSU. Also, we randomly chose 70 negative samples for each event class, while we repeated each experiment 10 times. That is, for each different experimental scenario, the obtained performance of each classifier (KSVM, RD-KSVM, LSVM-iGSU, KSVM-iGSU, and RD-SVM-iGSU) is averaged over 10 iterations, for each of which 10 positive samples have been randomly selected from the pool of 50 positive samples that are available in our training dataset for each target event class.

For all the above experimental scenarios where a kernel classifier is used, the radial basis function (RBF) kernel has been used. Training parameters ($C$ for LSVM-iGSU; $C$, $\gamma$ for KSVM, KSVM-iGSU; and $C$, $\gamma$, and $c$ for RD-KSVM, RD-KSVM-iGSU) are obtained via cross-validation. For $C$, $\gamma$, a 10-fold cross-validation procedure (grid search) is performed with $C$, $\gamma$ being searched in the range $\{2^{-16}, 2^{-15}, \ldots, 2^{2}, 2^{3}\}$. For $c$, an approach similar to that presented in [28] is followed. That is, related samples are initially treated as true positive and true negative ones (in two separate cross-validation processes) and $C$, $\gamma$ are optimized as described above; then, by examining the minimum cross-validation errors of the two above processes, we automatically choose whether to treat the related samples as weighted positive or weighted negative ones, and also fix the value of $C$ to the corresponding optimal value. Using this $C$, we proceed with a new cross-validation process (again grid search) for finding the optimal $\gamma$, $c$ pair (where $c$ is searched in the range $[0.01, 1.00]$ with a step of 0.05).

Table 1 shows the performance of the proposed KSVM-iGSU and RD-KSVM-iGSU, compared to LSVM-iGSU [27], the standard KSVM, and the RD-KSVM [28], respectively, in terms of average precision (AP), for each target event, and mean AP (MAP), across all target events. Bold-faced values indicate the best performance for each event class. We can see that LSVM-iGSU, whose improved performance over the standard linear SVM was studied extensively in [27], cannot outperform the kernel methods that are typically used for the video event detection problem, achieving a MAP of 0.1761. Without using any related samples, KSVM-iGSU that takes into account the input uncertainty, outperformed the standard kernel SVM for 25 out of 30 target event classes, achieving a MAP of 0.2527 in comparison to KSVM's 0.2128 (achieving a relative boost of 18.75 %). Moreover, when related samples were used for training, the proposed RD-KSVM-iGSU


**Fig. 1.** Indicative results (top-5 returned shots) for comparing RD-KSVM-iGSU with RD-KSVM, for four event classes.

outperformed the baseline RD-KSVM for 27 out of 30 target event classes, achieving a MAP of 0.2730, in comparison to RD-KSVM's 0.2218 (i.e. a relative boost of 23.08 %). This RD-KSVM-iGSU result also represents a 8 % relative improvement (MAP of 0.2730 versus 0.2527) in comparison to KSVM-iGSU, which does not take advantage of related video samples during training. The above results suggest that using uncertainty for training video event detectors leads to promising results, while the additional exploitation of related samples can further improve event detection performance.

Finally, in Fig. 1 we present indicative results of the proposed RD-KSVM-iGSU in comparison with the baseline RD-KSVM [28] for four event classes, showing the top-5 videos each classifier retrieved. Green borders around frames indicate correct detection results, while red ones indicate false detection. These

**Table 1.** Evaluation of event detection approaches on the MED14 dataset.

| Event class | LSVM-iGSU [27] | KSVM (e.g. [5,11]) | KSVM-iGSU (proposed) | RD-KSVM [28] | RD-KSVM-iGSU (proposed) |
|---|---|---|---|---|---|
| E021 | 0.1741 | 0.1763 | 0.1923 | 0.1823 | **0.2167** |
| E022 | 0.1847 | 0.1903 | 0.2495 | 0.2009 | **0.2604** |
| E023 | 0.4832 | 0.5665 | 0.6361 | 0.5435 | **0.6432** |
| E024 | 0.0536 | 0.0482 | **0.0667** | 0.0489 | 0.0549 |
| E025 | 0.0117 | 0.0210 | 0.0257 | 0.0200 | **0.0287** |
| E026 | 0.1002 | 0.1388 | 0.1530 | 0.1385 | **0.1701** |
| E027 | 0.1600 | 0.2882 | **0.4162** | 0.2899 | 0.4002 |
| E028 | 0.2030 | 0.2234 | 0.2338 | 0.2250 | **0.2495** |
| E029 | 0.2394 | 0.2321 | 0.2948 | 0.2521 | **0.3106** |
| E030 | 0.1612 | **0.2464** | 0.2220 | 0.2398 | 0.2451 |
| E031 | 0.4911 | 0.4595 | 0.6122 | 0.4762 | **0.6497** |
| E032 | 0.0706 | 0.1278 | 0.1490 | 0.1301 | **0.1729** |
| E033 | 0.2217 | 0.3170 | 0.3731 | 0.3265 | **0.3971** |
| E034 | 0.1658 | 0.2129 | 0.3302 | 0.2231 | **0.6541** |
| E035 | 0.2331 | 0.2650 | 0.3580 | 0.2874 | **0.3771** |
| E036 | 0.1753 | 0.1897 | 0.2139 | 0.1923 | **0.2230** |
| E037 | 0.2454 | 0.2928 | 0.3325 | 0.3133 | **0.3569** |
| E038 | 0.0745 | 0.1127 | 0.1231 | 0.1187 | **0.1259** |
| E039 | 0.2161 | 0.2531 | **0.3990** | 0.3294 | 0.3986 |
| E040 | 0.5809 | **0.3205** | 0.3157 | 0.3095 | 0.3021 |
| E041 | 0.0489 | 0.1589 | 0.2166 | 0.1782 | **0.2254** |
| E042 | 0.1021 | 0.1358 | 0.1787 | 0.1532 | **0.1799** |
| E043 | 0.0967 | 0.1568 | 0.2037 | 0.1890 | **0.2101** |
| E044 | 0.0732 | **0.2697** | 0.2087 | 0.2543 | 0.1968 |
| E045 | 0.1307 | 0.2315 | 0.2517 | 0.2385 | **0.2786** |
| E046 | 0.1952 | 0.2457 | 0.2668 | 0.2412 | **0.2721** |
| E047 | 0.0531 | 0.0837 | 0.1796 | 0.1187 | **0.1865** |
| E048 | 0.0672 | 0.0642 | 0.0672 | 0.0654 | **0.0674** |
| E049 | 0.0641 | 0.1250 | 0.1245 | 0.1189 | **0.1329** |
| E050 | 0.2076 | 0.2321 | 0.1867 | **0.2489** | 0.2039 |
| **MAP** | 0.1761 | 0.2128 | 0.2527 | 0.2218 | **0.2730** |

indicative results illustrate the practical importance of the AP and MAP differences between these two methods that are observed in Table 1.

## 5 Conclusions and Future Work

Two extensions of LSVM-iGSU, which is a linear classifier that takes input uncertainty into consideration, were proposed in this paper. The first one (KSVM-iGSU) results in non-linear decision boundaries, while the second one (RD-KSVM-iGSU), which is proposed especially for the problem of video event detection, exploits related class observations. The applicability of the aforementioned methods was verified using the TRECVID MED 2014 dataset, where solely a limited number of positive and related samples were used during training.

In the future, we plan to extend KSVM-iGSU such that the uncertainty of the input data is taken into consideration anisotropically. Also, we plan to exploit related samples in a more elaborate way; for instance, by clustering them into subclasses and assigning a different relevance degree to each subclass.

**Acknowledgment.** This work was supported by the European Commission under contract FP7-600826 ForgetIT.

## References

1. Bhattacharyya, C., Pannagadatta, K., Smola, A.J.: A second order cone programming formulation for classifying missing data. In: Neural Information Processing Systems (NIPS), pp. 153–160 (2005)
2. Bolles, R., Burns, B., Herson, J., et al.: The 2014 SESAME multimedia event detection and recounting system. In: Proceedings of the TRECVID Workshop (2014)
3. Broyden, C.G.: The convergence of a class of double-rank minimization algorithms 1. general considerations. IMA J. Appl. Math. **6**(1), 76–90 (1970)
4. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**, 27:1–27:27 (2011). http://www.csie.ntu.edu.tw/cjlin/libsvm
5. Cheng, H., Liu, J., Chakraborty, I., Chen, G., Liu, Q., Elhoseiny, M., Gan, G., Divakaran, A., Sawhney, H., Allan, J., Foley, J., Shah, M., Dehghan, A., Witbrock, M., Curtis, J.: SRI-Sarnoff AURORA system at TRECVID 2014 multimedia event detection and recounting. In: Proceedings of the TRECVID Workshop (2014)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition CVPR 2009, pp. 248–255. IEEE (2009)
7. Douze, M., Oneata, D., Paulin, M., Leray, C., Chesneau, N., Potapov, D., Verbeek, J., Alahari, K., Harchaoui, Z., Lamel, L., Gauvain, J.L., Schmidt, C.A., Schmid, C.: The INRIA-LIM-VocR and AXES submissions to TRECVID 2014 multimedia event detection (2014)
8. Gkalelis, N., Markatopoulou, F., Moumtzidou, A., Galanopoulos, D., Avgerinakis, K., Pittaras, N., Vrochidis, S., Mezaris, V., Kompatsiaris, I., Patras, I.: ITI-CERTH participation to TRECVID 2014. In: Proceedings of the TRECVID Workshop (2014)
9. Gkalelis, N., Mezaris, V.: Video event detection using generalized subclass discriminant analysis and linear support vector machines. In: Proceedings of International Conference on Multimedia Retrieval, p. 25. ACM (2014)

10. Golub, G.H., Van Loan, C.F.: Matrix Comput., vol. 3. JHU Press, Baltimore (2012)
11. Guangnan, Y., Dong, L., Shih-Fu, C., Ruslan, S., Vlad, M., Larry, D., Abhinav, G., Ismail, H., Sadiye, G., Ashutosh, M.: BBN VISER TRECVID 2014 multimedia event detection and multimedia event recounting systems. In: Proceedings of the TRECVID Workshop (2014)
12. Habibian, A., van de Sande, K.E., Snoek, C.G.: Recommendations for video event recognition using concept vocabularies. In: Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval, pp. 89–96. ACM (2013)
13. Habibian, A., Mensink, T., Snoek, C.G.: Videostory: A new multimedia embedding for few-example recognition and translation of events. In: Proceedings of the ACM International Conference on Multimedia, pp. 17–26. ACM (2014)
14. Jiang, L., Meng, D., Mitamura, T., Hauptmann, A.G.: Easy samples first: self-paced reranking for zero-example multimedia search. In: Proceedings of the ACM International Conference on Multimedia, pp. 547–556. ACM (2014)
15. Jiang, L., Yu, S.I., Meng, D., Mitamura, T., Hauptmann, A.G.: Bridging the ultimate semantic gap: a semantic search engine for internet videos. In: ACM International Conference on Multimedia Retrieval (2015)
16. Jiang, Y.G., Bhattacharya, S., Chang, S.F., Shah, M.: High-level event recognition in unconstrained videos. Int. J. Multimedia Inf. Retrieval **2**(2), 73–101 (2013)
17. Kimeldorf, G., Wahba, G.: Some results on Tchebycheffian spline functions. J. Math. Anal. Appl. **33**(1), 82–95 (1971)
18. Lanckriet, G.R., Ghaoui, L.E., Bhattacharyya, C., Jordan, M.I.: A robust minimax approach to classification. J. Mach. Learn. Res. **3**, 555–582 (2003)
19. Liang, Z., Inoue, N., Shinoda, K.: Event Detection by Velocity Pyramid. In: Gurrin, C., Hopfgartner, F., Hurst, W., Johansen, H., Lee, H., O'Connor, N. (eds.) MMM 2014, Part I. LNCS, vol. 8325, pp. 353–364. Springer, Heidelberg (2014)
20. Liu, D.C., Nocedal, J.: On the limited memory BFGS method for large scale optimization. Mathematical prog. **45**(1–3), 503–528 (1989)
21. Mazloom, M., Habibian, A., Liu, D., Snoek, C.G., Chang, S.F.: Encoding concept prototypes for video event detection and summarization (2015)
22. Over, P., Awad, G., Michel, M., Fiscus, J., Sanders, G., Kraaij, W., Smeaton, A.F., Quenot, G.: An overview of the goals, tasks, data, evaluation mechanisms and metrics. In: Proceedings of the TRECVID 2014. NIST, USA (2014)
23. Robertson, S.: A new interpretation of average precision. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 689–690. ACM (2008)
24. Schölkopf, B., Herbrich, R., Smola, A.J.: A generalized representer theorem. In: Helmbold, D.P., Williamson, B. (eds.) COLT 2001 and EuroCOLT 2001. LNCS (LNAI), vol. 2111, pp. 416–426. Springer, Heidelberg (2001)
25. Shivaswamy, P.K., Bhattacharyya, C., Smola, A.J.: Second order cone programming approaches for handling missing and uncertain data. J. Mach. Learn. Res. **7**, 1283–1314 (2006)
26. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014). arXiv preprint arXiv:1409.1556
27. Tzelepis, C., Mezaris, V., Patras, I.: Linear maximum margin classifier for learning from uncertain data (2015). arXiv preprint arXiv:1504.03892
28. Tzelepis, C., Gkalelis, N., Mezaris, V., Kompatsiaris, I.: Improving event detection using related videos and relevance degree support vector machines. In: Proceedings of the 21st ACM International Conference on Multimedia, pp. 673–676. ACM (2013)

29. Xu, H., Caramanis, C., Mannor, S.: Robustness and regularization of support vector machines. J. Mach. Learn. Res. **10**, 1485–1510 (2009)
30. Xu, H., Mannor, S.: Robustness and generalization. Mach. Learn. **86**(3), 391–423 (2012)
31. Yu, S.I., Jiang, L., Mao, Z., Chang, X., Du, X., Gan, C., Lan, Z., Xu, Z., Li, X., Cai, Y., et al.: Informedia at TRECVID 2014 MED and MER. In: NIST TRECVID Video Retrieval Evaluation Workshop (2014)