Complex Event Detection by Identifying Reliable Shots from Untrimmed Videos

Hehe Fan[1], Xiaojun Chang[2], De Cheng[3], Yi Yang[1]*, Dong Xu[4] and Alexander G. Hauptmann[2]
[1]Centre for Artificial Intelligence, University of Technology Sydney, Sydney, Australia
[2]School of Computer Science, Carnegie Mellon University, Pittsburgh, USA
[3]Institute of artificial intelligence and robotics, Xi'an Jiaotong University, Xi'an, China
[4]School of Electrical and Information Engineering, The University of Sydney, Sydney, Australia
Hehe.Fan@student.uts.edu.au, cxj273@gmail.com, chengde19881214@stu.xjtu.edu.cn
Yi.Yang@uts.edu.au, dong.xu@sydney.edu.au, alex+@cs.cmu.edu

Abstract

The goal of complex event detection is to automatically detect whether an event of interest happens in temporally untrimmed long videos which usually consist of multiple video shots. Observing some video shots in positive (resp. negative) videos are irrelevant (resp. relevant) to the given event class, we formulate this task as a multi-instance learning (MIL) problem by taking each video as a bag and the video shots in each video as instances. To this end, we propose a new MIL method, which simultaneously learns a linear SVM classifier and infers a binary indicator for each instance in order to select reliable training instances from each positive or negative bag. In our new objective function, we balance the weighted training errors and a l_1-l_2 mixed-norm regularization term which adaptively selects reliable shots as training instances from different videos to have them as diverse as possible. We also develop an alternating optimization approach that can efficiently solve our proposed objective function. Extensive experiments on the challenging real-world Multimedia Event Detection (MED) datasets MEDTest-14, MEDTest-13 and CCV clearly demonstrate the effectiveness of our proposed MIL approach for complex event detection.

1. Introduction

Complex event detection has attracted increasing attention in recent years. It aims to automatically detect an event of interest in temporally untrimmed long videos. Unlike human action recognition [27, 23, 22, 26] where actions are well-defined and videos are constrained, complex event detection is more challenging where the target event is complex and may only occur within a short period of time.

The existing works [25, 4] proposed to use different strategies to exploit temporal information for complex event detection. By decomposing the complex event like "grooming a dog" as a set of actions like "prepare water", "walk to dog", "groom dog", "leave dog" and "clean up", Tang *et al*. [25] adopted a Hidden Markov Model based approach to model the complex event. Cheng *et al*. [4] used a sequence memorizer approach [28] to exploit temporal information by treating each video shot as an individual visual word. However, the performance is still not quite satisfactory because it is a non-trivial task to effectively exploit temporal information in long videos like those from MEDTest-13 and MEDTest-14. Meanwhile, researchers also explored information from Wikipedia to provide more detailed explanation and definition for each complex event. Chang *et al*. [3] proposed to first prioritize the shots according to the salience scores from some pre-trained concept detectors, and then used a SVM based method to exploit ordering information of semantic concepts. Yan *et al*. [32] proposed a dictionary learning approach for complex event detection, in which the dictionary consists of a set of selected meaningful concepts for all events. However, these approaches heavily depend on human knowledge in order to design the elementary concepts for each complex event. It remains unclear what and how many concept detectors should be pre-trained for arbitrary events.

Recently, Li *et al*. [17] proposed a dynamic pooling method based on sliding windows. However, this sliding window based approach is computationally expensive. Moreover, the work in [13] partitioned each untrimmed long video as a set of video shots and used the multi-instance learning method $\propto$-SVM [35] for complex event detection. The major limitation of this work is the assumption that almost all video shots in positive bags are positive and all

*Corresponding author. This work is in part supported by Data to Decisions Cooperative Research Centre (www.d2dcrc.com.au), in part supported by the Google Faculty Research Award and in part supported by the award 60NANB17D156 from U.S. Department of Commerce, National Institute of Standards and Technology.

video shots in negative bags are negative, which is overstrict in the real-world complex event detection applications.

In this work, we first partition each untrimmed long video as a set of video shots. For each event class, we observe that some video shots in positive (resp. negative) videos are irrelevant (resp. relevant) to this event. This problem becomes even more severe for long videos. To address the unreliability issue, we formulate the complex event detection task as a multi-instance learning (MIL) problem by treating each video as a bag and the video shots in this video as instances. In contrast to the existing MIL methods that directly infer the labels of instances, we develop a new multi-instance learning method for complex event detection by identifying reliable instances from positive and negative bags. We propose a new objective function to simultaneously learn a linear SVM classifier and infer a binary indicator for each instance. The indicator indicates whether this instance is selected as a reliable instance.

In this paper, we only consider the training errors from the selected reliable shots because the unreliable shots could be misleading, *e.g.*, some shots may reside in the opposite side of the hyperplane. Thus, we directly discard them when training the SVM classifier. We additionally introduce a l_1-l_2 mixed-norm regularization term, which adaptively selects reliable instances from different bags to have them as diverse as possible. Existing MIL methods usually restrict that all instances in a negative bag are all negative. However, this restriction could be too strict for shots in long videos. For example, when training a detector for the event "birthday party", a negative video may contain a shot of people singing. In this case, the shot of "singing" should not be used as a negative instance because birthday part may also have "singing" shots. Therefore, our algorithm relaxes the hard constraint and allows a certain percentage noisy instances in both positive and negative bags. Specifically, we impose a new constraint by guaranteeing a minimum percentage of reliable instances to be selected for each positive/negative bag, which generalizes the constraint in the conventional MIL methods (see Sec 2 for more details). To solve our non-trivial optimization problem, we develop an efficient alternating optimization approach to iteratively solve a weighted SVM optimization problem and infer the binary indicators for all instances.

2. Related Works

Hand-crafted and deep features for videos. Large progress has been made in generating compact and discriminative representations for video, particularly with the success of deep convolution neural networks. Wang *et al*. [27] proposed Improved Dense Trajectory (IDT) by extracting descriptors along dense tracked trajectories, which achieved good results on action recognition datasets but is computationally expensive. Two-stream convolutional neural networks [23] consist of a static frame stream and an additional optical flow stream that takes stacked optical flow images as inputs. It achieves comparable performance to IDT and has motivated many researchers to use convolution neural networks for video motion modeling. C3D networks [26], on the other hand, directly use 3D convolutional network to learn spatio-temporal representations for short video clips. Motion modeling is important in action recognition where videos usually have 5-20 seconds duration. However, motion information can be noisy in real-world untrimmed long videos. For example, in most user generated YouTube videos, camera motion is not constrained and the quality of optical flow varies. In this work, we focus on complex event detection where videos are untrimmed (usually in 2-3 minutes) and more diverse than videos in action recognition datasets.

Multi-instance learning. Multi-instance learning (MIL) methods can be generally classified as bag-based MIL and instance-based MIL. The instance-based approaches like mi-SVM [1], MIL-CPB [15] and KI-SVM [18] explicitly infer the labels of instances, while bag-based approaches like MI-SVM [1] and Sparse MIL [2] do not infer their labels. In most existing MIL methods, the commonly used assumption is that each positive bag consists of at least one positive instance, while all instances in a negative bag are negative. This constraint was relaxed as a more general constraint in [15] that each positive bag consists of at least a certain percentage of positive instances. A similar constraint was also used in [35]. Recently, Liu *et al*. [20] used key instances that are selected by a clustering algorithm for MIL. However, the key instances voted by the clustering algorithm sometimes can not stand for the reliable instances when most of the instances are around the optimal hyperplane. Li *et al*. [16] trained a classifier by using top-k instances that are most likely to be positive (SMIL-TopK). The problem is that a fixed number of positive instances is not reasonable in many practical applications. Furthermore, this strategy fails to use as many training data as possible and will suffer from over-fitting when k is too small. Zhang *et al*. [37] integrated self-paced learning (SPL)[11] into mi-SVM, but they still flip instances' labels during training. Duchenne *et al*. [5] proposed an MIL-inspired discriminative clustering method to find segment locations of the action of interest. For complex event detection, the modified K-Means method cannot achieve satisfactory clustering results, which will degrade the performance of the SVM classifier. In contrast to these MIL methods, we propose a new MIL method by identifying reliable instances.

3. Multi-instance learning by selecting reliable instances (MIL-SRI)

In this section, we present our proposed new multi-instance learning method for complex event detection. The

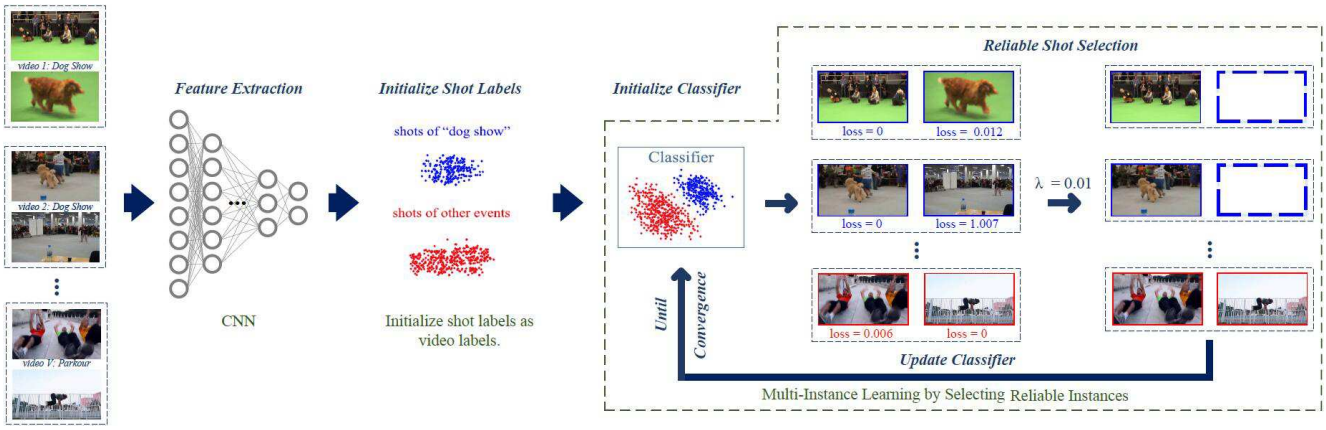


Figure 1: An illustration of the training process of the detector for the event "dog show" by using our MIL-SRI method.

framework for complex event detection is shown in Fig 1. After partitioning each untrimmed long video as a set of video shots, we extract the Convolutional Neural Network (CNN) features from a set of uniformly sampled key-frames in each video shot. The frame features are then merged as a single feature vector to represent the video shot by using pooling methods. Next, the shot labels are initialized as the corresponding video label and the linear SVM classifier is Single Instance Learning SVM (SIL-SVM) [2]. Then the framework selects reliable shots to learn and update the classifier, until no more shots are added into the reliable shot set. We define three types of instances in our model:

- Reliable instances. Positive (resp. negative) instances in positive (resp. negative) bags.
- Ambiguous instances. Instances that are difficult to be defined or classified as positive or negative instances.
- Noisy instances. Positive (resp. negative) instances in negative (resp. positive) bags.

The three types of shots in videos are illustrated in Fig 2 (a).

3.1. Problem Formulation

Suppose we have V videos and $\mathbf{x}_v^i$ is the feature vector of the i-th shot in the v-th video. In our method, each shot is considered as an "instance" and each video is considered as a "bag". A positive or negative bag $\mathcal{B}_v$ is associated with a bag label $Y_v \in \{\pm 1\}$ and the instance-level labels are unknown. We assume that the reliability ratio is known and not less than P_v, which is defined as the proportion of positive instances in the positive bag or the proportion of negative instances in the negative bag. We denote the transpose of a vector/matrix by superscript $'$. The aim of complex event detection is to learn a linear SVM classifier for each event based solely on the video-level label information.

3.2. MIL-SRI model

According to the above settings, complex event detection can be transformed into binary classification and solved by multi-instance learning. In traditional multi-instance learning methods, one step is to infer instances' labels, which means that instance-level labels are updated in training iterations. However, it is non-trivial to infer the instances' labels. We do not infer instance-level labels but train the SVM classifier by inferring a set of reliable instances.

In this work, we assume the decision function is in the form of $f(\mathbf{x}) = \mathbf{w}'\mathbf{x} + b$. Suppose q_v^i denotes the selection indicator of instance $\mathbf{x}_v^i$. If q_v^i equals to 1, $\mathbf{x}_v^i$ is selected as a reliable instance; otherwise, the shot is discarded when training the SVM classier. We further denote $\mathbf{q}_v = [q_v^1, ..., q_v^{|\mathcal{B}_v|}]$ as the selection indication vector for the v-th video and $|\mathcal{B}_v|$ stands for the cardinality of bag $\mathcal{B}_v$. We formulate our idea as the following optimization problem:

$$\min_{\mathbf{w},b,\mathbf{q}_v,\boldsymbol{\varepsilon}_v} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{v=1}^{V}\left(\mathbf{q}_v'\boldsymbol{\varepsilon}_v - \lambda\|\mathbf{q}_v\|_1 - \gamma\|\mathbf{q}_v\|_2\right)$$
$$s.t. \begin{cases} Y_v(\mathbf{w}'\mathbf{x}_v^i + b) \geq 1 - \varepsilon_v^i; \\ \varepsilon_v^i \geq 0; \\ \|\mathbf{q}_v\|_1 \geq P_v|\mathcal{B}_v|; \\ q_v^i \in \{0,1\}. \end{cases} , v = 1, 2, ..., V \tag{1}$$

where $C > 0$ is the parameter that controls the relative importance of detection correctness, $\boldsymbol{\varepsilon}_v = [\varepsilon_v^1, ..., \varepsilon_v^{|\mathcal{B}_v|}]$ is the slack variable vector of the v-th video and λ, γ are two non-negative parameters. The first term $\frac{1}{2}\|\mathbf{w}\|^2$ controls the complexity of our model to avoid overfitting. The second term aims to minimize the training errors from a set of selected confident training instances. To minimize the second term, we need to set all $\mathbf{q}_v$ as zeros, which tends not to select any training instances as confident ones.

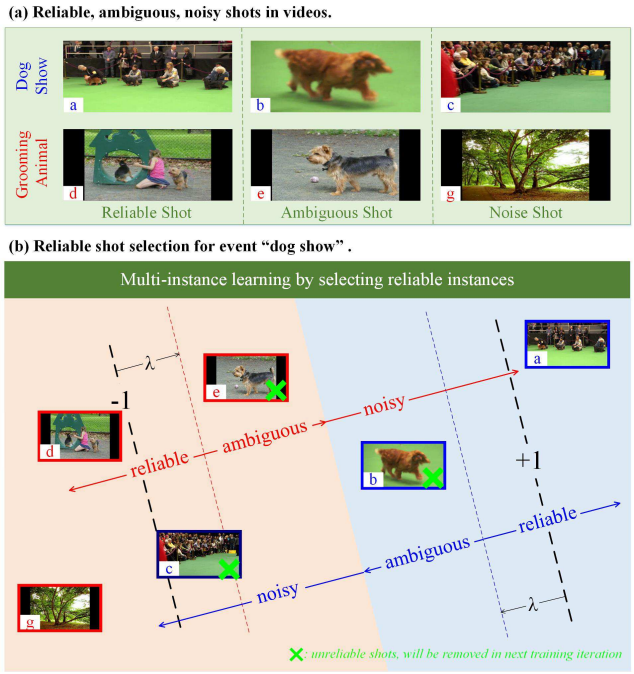


Figure 2: Illustration of different types of shots in videos and the basic principle of our method. Besides dogs and hosts, audience usually appears in the event "dog show" when compared with the event "grooming animal". A shot only with a dog is ambiguous in both events. From the perspective of SVM, shots with $loss \leq \lambda$ are reliable shots, while those with $\lambda < loss \leq 1$ are ambiguous shots, and shots with $1 < loss$ will be treated as noisy shots. These ambiguous and noisy shots should not be used when training the SVM classifier.

By adding the negative sign in front of the l_1-l_2 mixed-norm regularizer $(\lambda\|\mathbf{q_v}\|_1 + \gamma\|\mathbf{q_v}\|_2)$, we need to set all $\mathbf{q}_v$ as ones in order to minimize this -l_1-l_2 mixed-norm regularizer, which encourages all training instances to be selected as training instances. By balancing the training errors and the l_1-l_2 mixed-norm regularizer in Eq. (1), we can select the most reliable training instances (shots) by using our newly proposed approach. Next, we describe how the mixed-norm regularizer and constraints influent multi-instance learning's behaviors.

•$\boldsymbol{\lambda \neq 0, \gamma = 0}$: When $\lambda > 0$, the instance whose loss is less than λ is discarded in the iterative process. The parameter λ is a trade-off to remove unreliable instances. Meanwhile, when the training errors are becoming smaller during the iterative process, more selection indicators $\{q_v^i\}$ are expected to become 1 in order to minimize the negative l_1-l_2 mixed-norm regularization term. As a result, more and more instances are added into the reliable instance set, until no more reliable instances can be added. Fig 2 illustrates how to select shots for event "dog show" according to λ.

•$\boldsymbol{\lambda \neq 0, \gamma \neq 0}$: We have defined P_v as the lower bound for the percentage of reliable shots in each video, which can guarantee that at least $P_v|\mathcal{B}_v|$ shots will be selected from each video when learning the SVM classifier. This constraint can conduct selection diversity to some extent, however, we aim to flexibly select more reliable instances from different videos by utilizing an additional l_2-norm regularization term. Fig 3 illustrates that more reliable instances can be selected from a diverse set of bags after using this l_2-norm regularizer.

•$\|\mathbf{q_v}\|_1 \geq P_v|\mathcal{B}_v|$: With this constraint, we will choose at least a certain percentage of reliable instances from each bag. This can avoid a special case that all instances are inferred as unreliable instances, which may lead to an incorrect hyperplane. In the case of $P_v = 1/|\mathcal{B}_v|$ if $Y_v = +1$ and $P_v = 1$ if $Y_v = -1$, our new constraint will become the constraint in many existing multi-instance learning methods [1], namely, there is at least one positive instance in each positive bag and all instances in a negative bag are negative.

3.3. Optimization Strategy

To solve MIL-SRI, we alternately optimize the involved parameters $(\mathbf{w}, b)$ and $\{\mathbf{q}_v\}_{v=1}^V$. Let $L(\cdot) \geq 0$ be a classic loss function for supervised learning, e.g., hinge loss.

• Optimize $(\mathbf{w}, b)$ when $\{\mathbf{q}_v\}_{v=1}^V$ is fixed: In this step, the optimal problem degenerates to the classic weighted SVM problem [33]:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{v=1}^{V}\sum_{i=1}^{|\mathcal{B}_v|} q_v^i L(Y_v, \mathbf{w}'\mathbf{x}_v^i + b), \quad (2)$$

which can be easily solved. Suppose $|\mathcal{B}|$ is the average number of shots per video, and d is the number of the feature dimension. Since we learn a linear SVM classifier, the computational complexity of this step is $\mathcal{O}(\max(V|\mathcal{B}|, d)\min(V|\mathcal{B}|, d)^2)$.

• Optimize $\{\mathbf{q}_v\}_{v=1}^V$ when $(\mathbf{w}, b)$ is fixed: The goal of this step is to select the instances according to the reliability and diversity. The optimization problem becomes the following:

$$\min_{\mathbf{q}_v} \sum_{v=1}^{V}\Big(\sum_{i=1}^{|\mathcal{B}_v|} q_v^i L(Y_v, \mathbf{w}'\mathbf{x}_v^i + b) - \lambda\|\mathbf{q_v}\|_1 - \gamma\|\mathbf{q_v}\|_2\Big)$$
$$s.t.\ \|\mathbf{q_v}\|_1 \geq P_v|\mathcal{B}_v|;\ q_v^i \in \{0,1\}; v = 1, 2, ..., V \quad (3)$$

We follow [8] to solve this optimal problem and the details are provided in Alg 1 (see step 3 to step 14). Specifically, the video shots with $j \leq P_v|\mathcal{B}_v|$ or $\delta_j' < \lambda + \gamma/(\sqrt{j} + \sqrt{j-1})$ will be selected into the training process, where j is the instance rank w.r.t. its loss value within its video. Since the threshold $\lambda + \gamma/(\sqrt{j} + \sqrt{j-1})$ decreases as

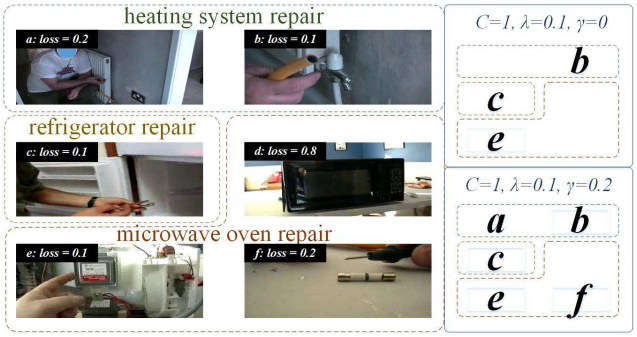


Figure 3: Effect of the l_2 norm for the event "repairing appliance". In this example, the regularizer $\frac{1}{2}\|\mathbf{w}\|^2$ is ignored and we set $C = 1$, $P_v = 1/|\mathcal{B}_v|$. When setting $\lambda = 0.1, \gamma = 0$, the algorithm tends to select shots $\{b, c, e\}$ in order to achieve the minimal loss 0 while the constraint $\|\mathbf{q_v}\|_1 \geq P_v|\mathcal{B}_v|$ is satisfied. When setting $\lambda = 0.1, \gamma = 0.2$, our algorithm selects shots $\{a, b, c, e, f\}$ as it can lead to the minimal loss -0.4, which is smaller than -0.346 when selecting $\{b, c, e\}$. Therefore, our algorithm can select the shots from a diverse set of videos when we additionally consider the l_2-norm based regularizer.

Algorithm 1: Optimization Procedure

Input : Instances $\{\{\mathbf{x}_v^i\}\}$; Video-level labels $\{Y_v\}$; Reliability ratio $\{P_v\}$; Video length $\{|\mathcal{B}_v|\}$; Parameters C, λ, γ.
Output: Classifier $(\mathbf{w}, b)$.
Initialization: $q_v^i \leftarrow 1$.
while *not convergence* **do**
  optimize $(\mathbf{w}, b)$ when $\mathbf{q}_v$ is fixed;
  for $v = 1$ *to* V **do**
    calculate loss $\delta_i = L(Y_v, \mathbf{w}'\mathbf{x}_v^i + b)$;
    sort $\{\delta_i\}$ in ascending order $\rightarrow \{\delta'_j\}$;
    denote $\tau(j)$ as $\delta'_j = \delta_{\tau(j)}$;
    for $j = 1$ *to* $|\mathcal{B}_v|$ **do**
      if $\delta'_j < \lambda + \frac{\gamma}{\sqrt{j}+\sqrt{j-1}}$ ***or*** $j \leq P_v|\mathcal{B}_v|$ **then**
        $q_v^{\tau(j)} \leftarrow 1$;
      else
        $q_v^{\tau(j)} \leftarrow 0$;
      end
    end
  end
end

the rank j grows for each video, this optimization strategy penalizes the shots that are monotonously selected from the same video and thus naturally conducts diversity. The computational complexity of loss calculation is $\mathcal{O}(V|\mathcal{B}|d)$ and the sort operation costs $\mathcal{O}(V|\mathcal{B}|log|\mathcal{B}|)$. Therefore, the computation complexity of this step is $\mathcal{O}(V|\mathcal{B}|d)$ because we usually have $d >> log|\mathcal{B}|$.

3.4. Event Inference

In test time, we use max-pooling to detect whether an event appears in a video. It is defined as:

$$score = \max(\mathbf{w}'\mathbf{x}_i + b), i = 1, 2, ..., |\mathcal{B}|. \quad (4)$$

Therefore, the final video score is determined only by the discriminative shot with the highest score.

4. Experiments

In this section, we conduct thorough experiments to validate our method on 1 synthetic dataset and 3 real-world datasets. The parameter C is set to 1 in all our experiments. We stop training when the difference of $\sum_{v=1}^{V}\|\mathbf{q_v}\|_1$ between two iterations is less than 100.

4.1. Experiments on the synthetic dataset

To provide some intuition on the behavior of the proposed MIL-SRI method, we conduct the experiments on a synthetic dataset with the noisy ratio as 1/11. The noisy ratio is defined as the percentage of negative instances in positive bags or the percentage of positive instances in negative bags. In our experiments, each bag contains 11 instances with each representing a point (x, y) in a two-dimensional space $\mathbb{R}^2$. If $x > 0$, the instance is positive; otherwise it is negative. In the training examples, the normal instances are sampled from the Gaussian distribution $(\mathcal{N}(0, 2), \mathcal{N}(0, 1))$ and the noisy instances are sampled from the Gaussian distribution $(\mathcal{N}(0, 0.01), \mathcal{N}(0, 1))$. Since the training points in the dataset are symmetric with respect to the y-axis, the ideal hyperplane is $x = 0$. For test examples, we rescale the sample space in order to accurately evaluate the learned hyperplane, in which all instances are sampled from $(\mathcal{N}(0, 0.01), \mathcal{N}(0, 1))$. The training dataset contains 2,000 bags and the test dataset contains 1,000 bags. We evaluate the following 4 methods on this dataset:

• **Sparse MIL**: Calculate the central point for each bag by the operation of average-pooling on instances, which aims to train a bag-level classifier.

• **SIL-SVM**: Assign instances' labels as the corresponding bags' label and then train an instance-level classifier.

• **SMIL-TopK**: Choose the most confident k instances in each bag to train an instance-level classifier. Since each bag contains 11 instances, the parameter k is set from 1 to 10.

• **MIL-SRI**: Since the bags are sampled from the same distributions without diversity, we set $\gamma = 0$ and λ is chosen from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$ in our experiments.

From the results illustrated in Fig 4, we can see that MIL-SRI learns the closest hyperplane to the ideal hyper-

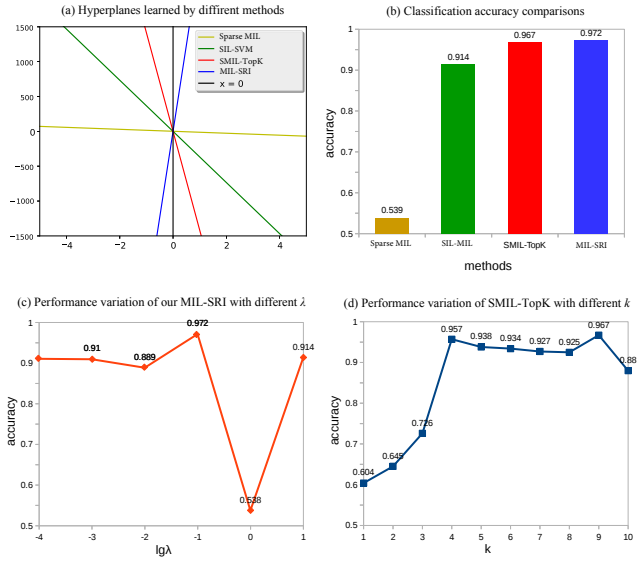


Figure 4: Classification accuracy comparisons on the synthetic dataset.

plane $x = 0$ and therefore achieves the best accuracy 0.972. Sparse MIL obtains the lowest accuracy because the learned hyperplane is severely shifted. We also illustrate the effects of different λs on MIL-SRI and the effects of different ks on SMIL-TopK. Our MIL-SRI outperforms the SMIL-TopK with 0.5% improvement and much larger performance improvement is achieved on the real-words datasets.

4.2. Experiments on the real datasets

In this section, we conduct thorough experiments on the real datasets to evaluate our proposed framework.

Table 1: mAP(%) comparison of baselines on the TRECVID MEDTest-14 and MEDTest-13 datasets.

	MEDTest-14		MEDTest-13	
	100Ex	10Ex	100Ex	10Ex
Sparse MIL	19.8	14.3	23.3	16.4
SIL-SVM	22.2	18.5	24.0	19.8
RNN-LSTM	31.1	20.4	33.8	21.7
SMIL-TopK	36.9	26.2	40.5	26.9
MIL-SRI (Ours)	**38.6**	**28.4**	**43.1**	**28.7**

Datasets: Following recent works on Multimedia Event Detection (MED) [3, 32], we test on three real-world event detection datasets. To our best knowledge, these are the largest public datasets for complex event detection.

- MEDTest-14: The TRECVID MEDTest 2014 dataset contains approximately 100 positive training examples per event, and all events share the same (∼5000) negative training exemplars. The testing set has approximately 23,000 videos. There are 20 events in total, whose detailed descriptions can be found at [1].

- MEDTest-13: Similar to MEDTest 2014 dataset. Note that 10 of 20 events overlap with those in MEDTest-14. The detailed event description can be found at [2].

- CCV: The Columbia Consumer Video dataset contains 9,317 videos in total from 20 semantic categories, including events like "baseball" and "parade" [9].

For the two MED datasets, we detect each event separately according to the NIST standard. We consider both 100Ex and 10Ex settings provided by NIST, which have 100 and 10 positive training exemplars respectively.

Feature extraction: We first segment each video into multiple video shots using the color histogram difference as the indication of the shot boundary. For simplicity, we choose the center frame from each shot, resize it to 224×224 and extract the frame feature from the fc6 layer of VGG16 [24].

Evaluation metric: According to the NIST standard, we evaluate the event detection performance by mean Average Precision (mAP). Average precision (AP) is a single-valued metric approximating the area under the precision-recall curve, which is widely used in information retrieval tasks. Mean Average Precision is the mean of AP over all event classes.

4.2.1 Comparison with the baseline algorithms

We first evaluate the performances of different baselines. Except the baseline methods in Sec 4.1, we additionally evaluate the method based on Recurrent Neural Networks (RNN).

In the RNN-LSTM baseline, we sample one frame per second from the video and then extract the same CNN feature for each frame. Afterwards, the Long Short-Term Memory (LSTM) [7] network takes CNN features as inputs and generates an output at each step. We average the output activations and conduct a 21-way classification (20 categories and 1 background category). The LSTM cell size is set to 1,024.

In the SMIL-TopK method, the parameter k is decided by using cross-validation from the range $\{0.1, 0.2, ..., 0.9\}$. In our method, both λ and γ are decided by using cross-validation from the range of $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^{2}\}$. We set the percentage P_v to 0.1 and 0.5 for positive and negative videos, respectively. We follow the TRECVID protocol

[1] http://nist.gov/itl/iad/mig/med14.cfm
[2] http://nist.gov/itl/iad/mig/med13.cfm

Table 2: mAP(%) comparison between our MIL-SRI method and the state-of-the-art baselines using a **single** type of feature on the TRECVID MEDTest-14, MEDTest-13 and CCV datasets.

(a) MEDTest-14

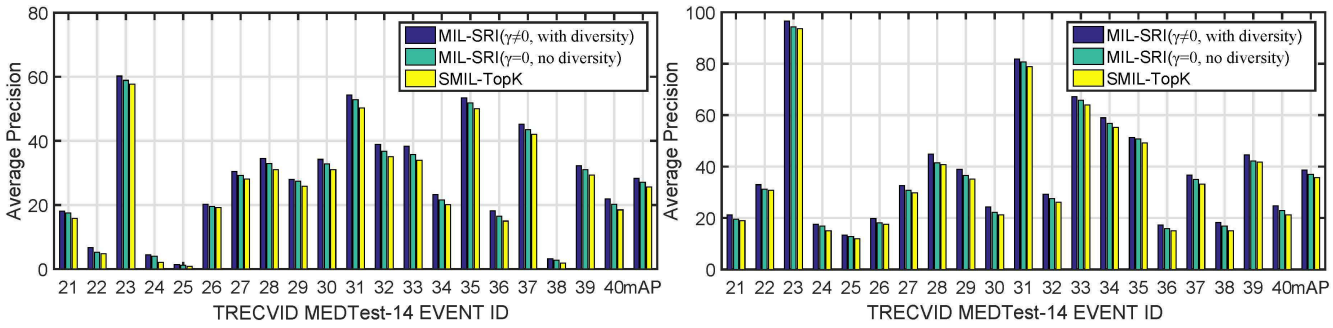
Method	100Ex	10Ex
DP [17]	28.8	17.6
$\propto$-SVM [13]	28.6	17.4
ESR [12]	29.6	18.4
CNN-Exp [36]	29.7	–
STN [10]	30.4	19.8
C3D [26]	31.4	20.5
MIFS [14]	29.0	14.9
CNN + VLAD [30]	35.7	23.2
NI-SVM [3]	34.4	26.1
MIL-SRI (Ours)	**38.6**	**28.4**

(b) MEDTest-13

Method	100Ex	10Ex
Video Story [6]	32.0	19.6
$\propto$-SVM [13]	33.4	21.6
ESR [12]	36.2	20.1
C3D [26]	36.9	22.2
MIFS [14]	36.9	19.3
STN [10]	37.1	20.4
StMM + TP [21]	38.6	21.8
CNN + VLAD [30]	40.3	25.6
NI-SVM [3]	39.2	26.8
MIL-SRI (Ours)	**43.1**	**28.7**

(c) CCV

Method	CCV
Benchmark [9]	59.5
$\propto$-SVM [13]	64.3
FWOT [31]	60.3
GRLF [34]	64.0
SSLF [19]	69.5
RDNN [29]	70.6
StMM + TP [21]	71.7
C3D [26]	77.2
NI-SVM [3]	**78.3**
MIL-SRI (Ours)	77.9



(a) MEDTest-14 100Ex (b) MEDTest-14 10Ex

Figure 5: Per event AP (%) comparison on the MEDTest-14 dataset. The figures are best viewed in color.

and train one detector for each event. The experimental results are reported in Table 1.

From the experimental results, we can see that sparse MIL and SIL-SVM methods achieve lower performances. The RNN-LSTM method can be treated as one type of pooling operation and it achieves better performance. Both SMIL-TopK and MIL-SRI attain higher performances, which demonstrates the importance of using reliable shots.

4.2.2 Comparison with models using a single feature

In this section, we compare our method with a few recent state-of-the-art methods that use a **single** type of feature. The experimental results are reported in Table 2. Note that we list the numbers directly from the papers. When the results are not available, we use the code from the respective authors to obtain the results by ourselves. Following [10], we pre-train Spatial Temporal Network (STN) by using the Sports-1M dataset [10]. Then we fine-tune the top three layers using the tested datasets. We also use the pre-trained C3D network [26] and fine-tune with the tested dataset.

From the experimental results, we can observe that the proposed method compares favorably against the other methods. Note that previous methods can only achieve the best performance on some datasets, *e.g.*, Xu *et al.* [30] achieved the state-of-the-art results on MEDTest-13 100Ex while Chang *et al.* [3] obtained the best performance on MEDTest-13 10Ex. However, our method consistently achieves the state-of-the-art performance on both MEDTest-13 and MEDTest-14 datasets. Specifically, we outperform the state-of-the-art methods by 2.9% (2.3%) on MEDTest-14 100Ex (10Ex) and 2.8% (1.9%) on MEDTest-13 100Ex (10Ex), which is a large improvement for both datasets.

Next, we study the influence of the norms in the regularizer on MEDTest-14 dataset. Note that λ and γ are two non-negative parameters to balance the reliability and diversity, respectively. Specially, we report the results of MIL-SRI, MIL-SRI without diversity by setting $\gamma = 0$ and SMIL-TopK. The results are shown in Fig 5. From the experimental results, we can observe that our MIL-SRI method consis-

Figure 6: Visualization of shot selection results. There are three types of shots, *i.e.*, reliable shots, ambiguous shots and noisy shots. In ambiguous shots, we can observe **semantic ambiguity** and **visual ambiguity**. Specifically, semantically ambiguous shots have similar semantic meaning to the target event or contain a few of elementary concepts about it. In visually ambiguous shots, the event actually happens but can not be detected evidently due to dim weather, camera angles or distances. These ambiguous shots lead to non-discriminative event features and our MIL-SRL removes them from the training set. The figures are best viewed in color.

tently outperforms the special case (MIL-SRI with $\gamma = 0$) and SMIL-TopK. It indicates the effectiveness of l_2 norm in the regularizer. To be specific, for the event "rock climbing" (E027), the MIL-SRI method significantly outperforms the other two methods on MEDTest-14 100Ex (32.64% vs 30.76% and 29.78%).

Table 3: mAP (%) comparison against state-of-the-art systems that fuse **multiple** types of features on the TRECVID MEDTest-14 and MEDTest-13 datasets.

	MEDTest-14		MEDTest-13	
	100Ex	10Ex	100Ex	10Ex
C3D [26] + IDT	33.6	22.1	39.5	26.7
CNN-Exp [36]	38.7	–	–	–
CNN + VLAD [30]	36.8	24.5	44.6	29.8
NI-SVM + IDT [3]	38.1	27.2	46.3	31.5
MIL-SRI (Ours) + IDT	**41.5**	**29.6**	**49.7**	**34.6**

To demonstrate how our method works in complex event detection, we show some results of shot selection after the second training iteration in Fig 6. As can be seen, our method is able to distinguish the three types of shots, *i.e.*, reliable shots, ambiguous shots and noisy shots.

4.2.3 Comparison with the state-of-the-art systems

We also compare our method with some state-of-the-art systems that combine **multiple** types of features. Improved Dense Trajectories (IDT) [27] have dominated complex event detection in the past few years due to their excellent performance over other features. We fuse the prediction results from VGG16 and IDT by averaging classification scores.

The experimental results are reported in Table 3, from which we observe that the proposed method performs competitively against all the existing methods. After fusing the IDT feature, mAP can be improved from 28.4% to 29.6% on MEDTest-14 10Ex and even more improvement is obtained on MEDTest-13 100Ex (from 43.1% to 49.7%). It shows the benefit of fusing motion features in complex event detection.

Xu *et al.* [30] fused multiple features from three different layers of VGG16, *i.e.*, pool5, fc6 and fc7. Our method consistently outperforms [30] in both MEDTest-13 and MEDTest-14 datasets with about 5% absolute improvement. We also outperforms the previous state-of-the-art system [3] by more than 3% absolute improvement on MEDTest-13 dataset. Our method thus achieves the new state-of-the-art result on both MEDTest-14 and MEDTest-13 datasets.

5. Conclusion

To avoid the possibility that irrelevant shots may mislead detection, we aim to distinguish reliable and unreliable shots based solely on video-level labels. In this paper, we make a more reasonable assumption that both reliable and unreliable shots can occur in all videos and aim to learn SVM classifiers only by reliable shots. To solve the problem, we formulate this task as a multi-instance learning problem. Further, we propose the MIL-SRI method which can adaptively select reliable instances and needn't infer instances' label. By this method, we attain the state-of-the-art performance in complex event detection on two real-world datasets.

References

[1] S. A. andIoannis Tsochantaridis and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, 2002. 2, 4

[2] R. C. Bunescu and R. J. Mooney. Multiple instance learning for sparse positive bags. In *ICML*, 2007. 2, 3

[3] X. Chang, Y.-L. Yu, Y. Yang, and E. P. Xing. Semantic pooling for complex event analysis in untrimmed videos. *Trans. Pattern Anal. Mach. Intell.*, 2016. 1, 6, 7, 8

[4] Y. Cheng, Q. Fan, S. Pankanti, and A. N. Choudhary. Temporal sequence modeling for video event detection. In *CVPR*, 2014. 1

[5] O. Duchenne, I. Laptev, J. Sivic, F. R. Bach, and J. Ponce. Automatic annotation of human actions in video. In *ICCV*, 2009. 2

[6] A. Habibian, T. Mensink, and C. G. M. Snoek. Videostory: A new multimedia embedding for few-example recognition and translation of events. In *ACM MM*, 2014. 7

[7] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 1997. 6

[8] L. Jiang, D. Meng, S. Yu, Z. Lan, S. Shan, and A. G. Hauptmann. Self-paced learning with diversity. In *NIPS*, 2014. 4

[9] Y. Jiang, G. Ye, S. Chang, D. P. W. Ellis, and A. C. Loui. Consumer video understanding: a benchmark database and an evaluation of human and machine performance. In *ICMR*, 2011. 6, 7

[10] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F. Li. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 7

[11] M. P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *NIPS*, 2010. 2

[12] K. Lai, D. Liu, M. Chen, and S. Chang. Recognizing complex events in videos by learning key static-dynamic evidences. In *ECCV*, 2014. 7

[13] K. Lai, F. X. Yu, M. Chen, and S. Chang. Video event detection by inferring temporal instance labels. In *CVPR*, 2014. 1, 7

[14] Z. Lan, M. Lin, X. Li, A. G. Hauptmann, and B. Raj. Beyond gaussian pyramid: Multi-skip feature stacking for action recognition. In *CVPR*, 2015. 7

[15] W. Li, L. Duan, D. Xu, and I. W. Tsang. Text-based image retrieval using progressive multi-instance learning. In *ICCV*, 2011. 2

[16] W. Li and N. Vasconcelos. Multiple instance learning for soft bags via top instances. In *CVPR*, 2015. 2

[17] W. Li, Q. Yu, A. Divakaran, and N. Vasconcelos. Dynamic pooling for complex event recognition. In *ICCV*, 2013. 1, 7

[18] Y. Li, J. T. Kwok, I. W. Tsang, and Z. Zhou. A convex method for locating regions of interest with multi-instance learning. In *ECML*, 2009. 2

[19] D. Liu, K. Lai, G. Ye, M. Chen, and S. Chang. Sample-specific late fusion for visual category recognition. In *CVPR*, 2013. 7

[20] G. Liu, J. Wu, and Z. Zhou. Key instance detection in multi-instance learning. In *ACML*, 2012. 2

[21] M. Nagel, T. Mensink, and C. G. M. Snoek. Event fisher vectors: Robust encoding visual diversity of visual streams. In *BMVC*, 2015. 7

[22] J. Y. Ng, M. J. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, 2015. 1

[23] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, 2014. 1, 2

[24] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2014. 6

[25] K. D. Tang, F. Li, and D. Koller. Learning latent temporal structure for complex event detection. In *CVPR*, 2012. 1

[26] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015. 1, 2, 7, 8

[27] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013. 1, 2, 8

[28] F. Wood, C. Archambeau, J. Gasthaus, L. James, and Y. W. Teh. A stochastic memoizer for sequence data. In *ICML*, 2009. 1

[29] Z. Wu, Y. Jiang, J. Wang, J. Pu, and X. Xue. Exploring inter-feature and inter-class relationships with deep neural networks for video classification. In *ACM MM*, 2014. 7

[30] Z. Xu, Y. Yang, and A. G. Hauptmann. A discriminative CNN video representation for event detection. In *CVPR*, 2015. 7, 8

[31] Z. Xu, Y. Yang, I. W. Tsang, N. Sebe, and A. G. Hauptmann. Feature weighting via optimal thresholding for video analysis. In *ICCV*, 2013. 7

[32] Y. Yan, Y. Yang, D. Meng, G. Liu, W. Tong, A. G. Hauptmann, and N. Sebe. Event oriented dictionary learning for complex event detection. *IEEE Trans. Image Processing*, 2015. 1, 6

[33] X. Yang, Q. Song, and Y. Wang. A weighted support vector machine for data classification. *IJPRAI*, 2007. 4

[34] G. Ye, D. Liu, I. Jhuo, and S. Chang. Robust late fusion with rank minimization. In *CVPR*, 2012. 7

[35] F. X. Yu, D. Liu, S. Kumar, T. Jebara, and S. Chang. $\propto$svm for learning with label proportions. In *ICML*, 2013. 1, 2

[36] S. Zha, F. Luisier, W. Andrews, N. Srivastava, and R. Salakhutdinov. Exploiting image-trained CNN architectures for unconstrained video classification. In *BMVC*, 2015. 7, 8

[37] D. Zhang, D. Meng, C. Li, L. Jiang, Q. Zhao, and J. Han. A self-paced multiple-instance learning framework for co-saliency detection. In *ICCV*, 2015. 2