Searching Persuasively: Joint Event Detection and Evidence Recounting with Limited Supervision

Xiaojun Chang[1,2],Yao-Liang Yu[3],Yi Yang[1] and Alexander G. Hauptmann[2]
[1]Centre for Quantum Computation and Intelligent Systems, University of Technology Sydney
[2]Language Technologies Institute, Carnegie Mellon University
[3]Machine Learning Department, Carnegie Mellon University
{cxj273, yee.i.yang}@gmail.com,{yaoliang,alex}@cs.cmu.edu

ABSTRACT

Multimedia event detection (MED) and multimedia event recounting (MER) are fundamental tasks in managing large amounts of unconstrained web videos, and have attracted a lot of attention in recent years. Most existing systems perform MER as a postprocessing step on top of the MED results. In order to leverage the mutual benefits of the two tasks, we propose a joint framework that simultaneously detects high-level events and localizes the indicative concepts of the events. Our premise is that a good recounting algorithm should not only explain the detection result, but should also be able to assist detection in the first place. Coupled in a joint optimization framework, recounting improves detection by pruning irrelevant noisy concepts while detection directs recounting to the most discriminative evidences. To better utilize the powerful and interpretable semantic video representation, we segment each video into several shots and exploit the rich temporal structures at shot level. The consequent computational challenge is carefully addressed through a significant improvement of the current ADMM algorithm, which, after eliminating all inner loops and equipping novel closed-form solutions for all intermediate steps, enables us to efficiently process extremely large video corpora. We test the proposed method on the large scale TRECVID MEDTest 2014 and MEDTest 2013 datasets, and obtain very promising results for both MED and MER.

Categories and Subject Descriptors

H.3.1 [**Information Search and Retrieval**]: Content Analysis and Indexing; I.2.10 [**Vision and Scene Understanding**]: Video analysis

Keywords

Video Analysis; Multimedia Event Detection; Multimedia Event Recounting

1. INTRODUCTION

To deal with the rapidly growing amount of internet videos, this work is focused on two basic tasks in video analysis, event detection and event recounting, and we aim at addressing the two *simultaneously* in a joint framework so that they will greatly benefit from each other. As we will see, the joint framework is particularly advantageous when training data is scarce, because of the information sharing between the two tasks.

In ***event detection***, a sequence of unseen videos is presented and the algorithm must rank them according to their likelihood of belonging to, say the *birthday party* event. Obvious applications of event detection include, but are not limited to, video categorization and retrieval. Here, we are interested in complex events, which may be characterized by several scenes, objects, actions and the rich interactions between them. Because complex events can be extremely complicated or even abstract, their detection, despite a lot of recent progress [5, 11, 16, 22, 41], is still in its infancy. Factors that add to the difficulty include: 1). Web videos are usually unstructured and do not follow any particular distribution; 2). Only a few positive exemplars are available for a certain event during training; 3). The evidences of a certain event can scatter anywhere in a video, and each can be hard to reliably detect.

The most commonly used technique for complex event detection is to aggregate low-level visual features and then feed them to sophisticated statistical classification machines. A recent trend is to employ, instead, semantic *concept* representations [11, 12, 23, 24, 34, 36], which are found to be more discriminative and interpretable. Since not all concepts are related to a certain event, and their representation could be imprecise, our first goal in this paper is to be able to automatically identify the few relevant semantic concepts for detecting a particular complex event.

Event recounting refers to the task of providing *comprehensible* evidences to justify a detection result, *e.g.*, why is this video classified as a "birthday party" event? Low-level features are less useful for event recounting since they are not interpretable, while high-level concepts can provide very useful cues, *e.g.*, this video is a "birthday party" event because semantic concepts like "birthday cake" and "blowing candle" appeared. While most existing works [18, 39] perform event recounting only as a post-processing step after the detection phase, we believe that a good recounting algorithm should not only explain the detection, but also be able to assist detection in the first place.

To this end, we propose a joint framework, illustrated in Figure 1, to simultaneously classify high-level events and locate semantic evidences for each complex event. After extracting a semantic, albeit noisy, video representation, we introduce a recounting model based on recent sparse regularizers [6, 32, 45] that can localize key evidences both concept-wise and temporal-wise, and

MM'15, October 26–30, 2015, Brisbane, Australia.
© 2015 ACM. ISBN 978-1-4503-3459-4/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2733373.2806218.

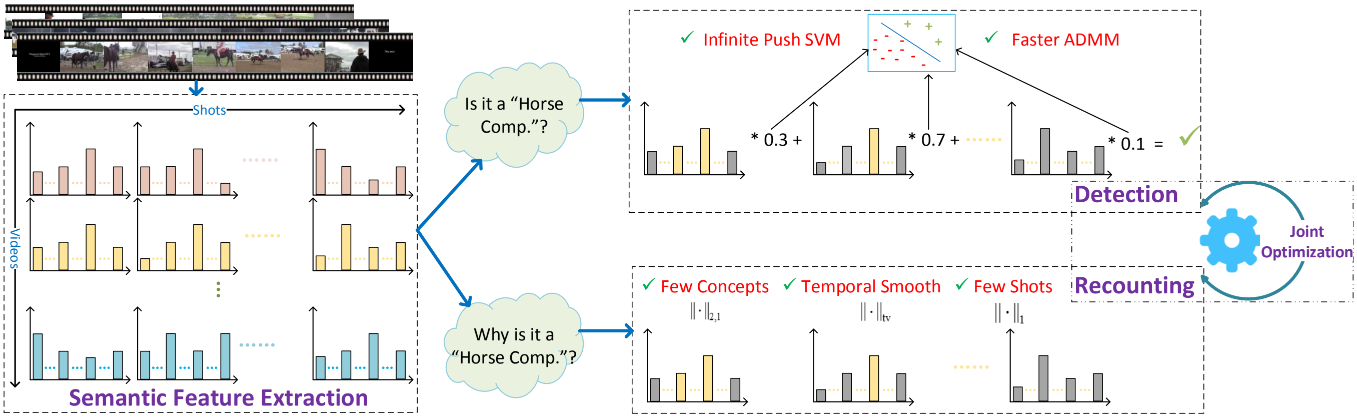

Figure 1: The proposed framework simultaneously conducts event detection and evidence recounting. Illustrated on the particular *Horse Competition* event. We first segment each video into multiple shots upon which we extract *semantic* features. Then we iterate between the detection model and the recounting model. We employ the infinite push SVM [31] for detection and develop a fast ADMM algorithm for it. Sparse regularizers are used to localize indicative concepts for recounting.

a detection model based on the infinite push support vector machine (SVM) [31] that greatly enhances the discriminative power. In a nutshell, our recounting model assists detection by filtering out noisy irrelevant information while simultaneously our detection model guides recounting by directing it to the most discriminative evidences. We further segment each video into multiple shots to exploit the rich temporal information, although this in turn creates a severe computational challenge, especially on large-scale datasets. Therefore, our second contribution is a significantly improved alternating direction method of multiplier (ADMM) algorithm. In contrast to existing works *e.g.* [18, 30], we prove novel closed-form solutions for all intermediate updates that allow us to remove all inner loops, resulting in tremendously improved efficiency. The per-step complexity of our algorithm scales only *linearly* with the problem size. We test our algorithm on the recent TRECVID MEDTest 2013 dataset [25] and MEDTest 2014 dataset [26], and achieve very promising results for both detection and recounting.

We summarize our contributions as follows:
(a). Unlike most previous works that address detection and recounting separately, we integrate event detection and recounting into a joint optimization framework, allowing us to exploit the mutual benefits of the two tasks, particularly when training data is scarce.
(b). We propose a very efficient ADMM algorithm that is key to perform recounting and detection jointly on large scale shot-level video representations.
(c). We conduct extensive experiments on the recent large-scale MEDTest 13 and MEDTest 14 datasets, and obtain very promising results for both detection and recounting.

Paper organization: We first review some related works on event detection and recounting in §2. Then we introduce our semantic representation in §3.1. The joint framework is presented in §3.2, and its optimization scheme is detailed in §4. Experiments are conducted in §5. Finally we conclude in §6.

Notations: We use $\|\cdot\|_F, \|\cdot\|_1, \|\cdot\|_2$ for the Frobenius norm, 1-norm, and 2-norm, respectively. For matrix A, we use the notation $A_{i,j}$, $A_{i,:}$, $A_{:,j}$ for its ij-th entry, i-th row, and j-th column, respectively. We split our m training videos into p positive exemplars $\mathfrak{P}$ and n negative exemplars $\mathfrak{N}$, where of course $m = p + n$. Frequently we will use A_+ to denote the positive part, *i.e.* $\max\{A, 0\}$, understood componentwise for vectors or matrices. The standard inner product $\langle A, B\rangle = \sum_{ij} A_{i,j} B_{i,j}$ is used throughout.

2. RELATED WORK

Complex event detection on unconstrained web videos has attracted wide attention in the field of multimedia and computer vision. A decent video event detection system usually consists of a good feature extraction module and a highly effective classification module (such as large margin support vector machines and kernel methods). Various low-level features (static, audio, *etc.*), *e.g.* the improved dense trajectories [40], already achieve good performances under the bag-of-words representation [35, 37]. Further improvements are obtained by aggregating complementary features in the video level, such as Fisher vector coding [27] and pooling [18, 19]. The latter approaches also try to exploit temporal information by decomposing an event video into short segments and selecting the most informative one for pooling. Unlike these methods, we aim at localizing *semantic* evidences that goes beyond feature pooling, and through a joint detection and recounting framework we achieve better performance.

A recent trend in complex event detection is to employ some high-level semantic representation, which can be advantageous in multiple aspects: 1). It leads to improved generalization capability and allows zero-shot learning (*i.e.* recognizing new events that are never seen in the training phase) [4, 12, 23, 24]. 2). It provides a meaningful way to aggregate low-level features. 3). It yields more interpretable results [11, 34, 36], hence may facilitate other video analysis tasks such as retrieval and recounting. Such semantic representations are usually trained on top of many low-level features, and have roots in object and action recognition [8, 14, 20, 42].

Although much progress has been made for complex event detection and video retrieval, semantic event recounting, which refers to the task of “explaining” a certain detection result, is relatively less addressed. A rule based approach was proposed in [36] to collect evidences, however, it worked on a tailored concept set which heavily relies on human knowledge. In a more controlled setting, [2] adopted object tracks and body-postures to generate textual evidences. By weighing the contributions of the individual concepts to the final detection score and mining the co-occurrence of different concepts, [18, 21, 39] performed event recounting only as a post-processing step after the detection phase. Interestingly, [21] obtained slightly worse detection accuracy when using the recounted key evidences to re-train the detection classifier. We believe this is because the event detection and recounting in [21] are not conducted *jointly* but in two separate stages. The recent work [34], which also considered simultaneous detection and recounting, is the most similar to our work. However, [34] modeled the evidence location as latent variables, which are harder to train when only few unconstrained videos are available. In contrast, our recounting model is much simpler and appears to be more efficient.

We would also like to mention some recent works on video summarization, *e.g.* [7, 10]. Unlike event recounting, video summarization is not oriented towards event detection: It merely aims at summarizing the vast information in a given video, a significant portion of which can be actually irrelevant to detection or recounting, though. How to effectively incorporate summarization information is an interesting direction yet to be explored.

3. JOINT DETECTION AND RECOUNTING

We start by introducing our semantic representation of the video ensemble, which unfortunately is usually noisy and not directly applicable for the recounting task. Then, we motivate our joint training protocol by separately detailing the recounting model and the detection model. The two models are combined into a joint framework to allow *simultaneous* detection and recounting.

3.1 Semantic Concept Representation

Videos, in their raw format, are represented by pixel values of each frame, which are not robust against small variations. Thus, in many video analysis tasks it is a common practice to first extract low level features, such as the improved dense trajectories [40]. To reduce the high dimensionality of low level features and increase their discriminative power, various pooling procedures can be applied on top [14, 35]. However, low level features usually do not have semantic meanings, thus are not suitable for interpretation purposes, such as the recounting task we consider in this work. In recent years, semantic representations based on *concepts, attributes,* and *actions* have been popular in video event detection, recognition and recounting [11, 21, 23, 24, 34, 36]. Usually, these semantic representations are trained on top of low level features.

In this work we represent each video by its confidence scores on c pre-defined concept classes $\mathcal{C} = \{C_1, C_2, \ldots, C_c\}$. The construction of these concepts is detailed in Section 5 below. In order to identify the key *temporal* evidences, we further split each video into s shots/clips. Here, for simplicity, we assume each video has the same number of shots, but the algorithm can be easily extended to the heterogeneous setting. Each shot (in the i-th video) is encoded by a confidence score vector $\mathbf{v}_t \in [0,1]^c, t = 1, \ldots, s$, where for each $k = 1, \ldots, c$, $(\mathbf{v}_t)_k$ indicates the confidence of the k-th concept being present in the t-th shot. Thus, the i-th video is encoded as the matrix $V^i = [\mathbf{v}_1, \ldots, \mathbf{v}_s] \in [0,1]^{c\times s}$, and we use $V = [V^1, \ldots, V^m] \in [0,1]^{c\times sm}$ to represent the entire video ensemble. Here m is the total number of videos. Building on this semantic representation, our goal is to decide whether or not each video V^i belongs to a certain event, and also identify the key evidences to support our classification results, i.e., based on what concepts appeared in which shots our detection algorithm made its judgment.

3.2 The Joint Training Protocol

Motivations: Our joint training protocol for simultaneous event detection and recounting is motivated by the following observations: 1). The concept detectors we use are far from perfect, partly because of the relatively small number of positive exemplars and partly because of the cross-domain training. Therefore, there is a considerable amount of noise in our semantic representation, hence it is only reasonable to treat each semantic representation V^i as a noisy perturbation of some ground truth R^i. 2). Usually, only a few concepts are relevant for detecting a certain event [3]. For instance, in detecting the birthday party event, we would expect concepts like "birthday cake" or "blowing candle" to be highly discriminative while others would be less useful or even misleading (worsened by the imperfections of our semantic representation). Thus it makes sense to find a clean and discriminative representation R^i that has many zero rows, *i.e.* containing only few relevant concepts. Moreover, since each relevant concept likely appears only in very few shots and in a temporally smooth way, we expect each row of R^i to contain only few *consecutive* nonzero entries. 3). Event recounting is different from video summarization (*e.g.* [7, 10]), in the sense that not all concepts appearing in our video are equally useful. In particular, event recounting is detection oriented: We are only interested in identifying the few concepts, along with their temporal positions, that are highly discriminative for our detection module. In other words, event recounting acts more like a recurrent supervised learning task: It aims at both explaining and serving the detection module. Ideally, we would like our "clean" representation R^i to be highly discriminative.

Our recounting model and detection model take the above observations into account. Specifically, our recounting model "denoises" the semantic representation so that it only contains few concepts appearing in few shots in a temporally smooth manner. This denoised representation localizes key evidences and is fed into the infinite push support vector machine (SVM) [1, 30, 31] to gain more discriminative power. Unlike previous recounting works [21, 39], we conduct event detection and event recounting *simultaneously* in a joint framework.

Recounting Model: Our recounting model finds the latent "clean" representation (*i.e.*, ground truth) $R = [R^1, \ldots, R^m]$ by solving the following denoising problem:

$$\min_R \tfrac{1}{2}\|V - R\|_{\mathsf{F}}^2 + \Omega(R), \tag{1}$$

where the regularizer Ω encodes our observation that only few concepts are relevant and they appear in few shots in a temporally smooth manner. Specifically, we use

$$\Omega(R) = \alpha\|R\|_{2,1} + \beta\|R\|_1 + \gamma\sum_{i=1}^m \|R^i\|_{\mathsf{tv}}, \tag{2}$$

where the group norm [45]

$$\|R\|_{2,1} = \sum_{k=1}^c \|R_{k,:}\|_2 \tag{3}$$

encourages many rows of R to be zeroed out (*i.e.* few concepts are relevant); the total variation semi-norm [32]

$$\|R^i\|_{\mathsf{tv}} = \sum_{k=1}^c\sum_{t=2}^s |R^i_{k,t} - R^i_{k,t-1}| \tag{4}$$

encourages piecewise constant entries (i.e. temporally smooth) in each row of R^i; and finally the 1-norm $\|R\|_1$ encourages sparsity [6] (i.e., relevant concepts appear only in few shots). By inspecting the support patterns (*i.e.* nonzero entries) of the "clean" representation R we can localize the key evidences hence perform recounting. However, so far the recounting model (1) is purely unsupervised hence may not be helpful for event detection. To enhance its discriminative power, we will couple it with our supervised detection model below. Note that we detect each event separately (as required in the NIST standard [25, 26]). Therefore, each event will have its own clean representation R, and may choose different indicative concepts. To our best knowledge, the formulation of our recounting model is new.

Detection Model: Our detection model follows the usual approach. It finds a discriminative linear classifier, parameterized by an appropriate matrix W, to distinguish the positive and negative "clean" representations R. From now on we split the m training videos into positive exemplars $\mathfrak{P}$ and negative exemplars $\mathfrak{N}$, with size respectively p and n. For each negative exemplar $j \in \mathfrak{N}$, the quantity $\sum_{i \in \mathfrak{P}} \mathbb{1}(\langle W, R^i \rangle < \langle W, R^j \rangle)$ counts the number of positive exemplars i that are ranked below j by the linear classifier W. These ranking errors with respect to each negative exemplar are combined to yield a loss that we aim to minimize. For computational tractability we upper bound the discrete 0-1 loss $\mathbb{1}(\delta < 0)$ by the *convex* hinge loss $(1-\delta)_+$, where as usual $(\delta)_+ := \max\{\delta, 0\}$ is the positive part. Since we usually pay more attention, if not exclusively, to the *top* of the rank list, we focus on minimizing the maximum ranking error among all negative exemplars $j \in \mathfrak{N}$:

$$\ell(W; R) := \max_{j \in \mathfrak{N}} \frac{1}{p} \sum_{i \in \mathfrak{P}} (1 - \langle W, R^i - R^j \rangle)_+. \tag{5}$$

Adding an appropriate regularizer Φ to control the model complexity, we obtain the infinite push support vector machine [1, 30, 31]:

$$\min_W \ \ell(W; R) + \Phi(W). \tag{6}$$

Usual choices for Φ include the (squared) 2-norm $\Phi(W) = \lambda \|W\|_{\mathsf{F}}^2$ [1] and the sparse 1-norm $\Phi(W) = \lambda \|W\|_1$ [30]. Since our "clean" representation R is already sought to be sparse, we will mostly use the 2-norm in our experiments, due to its better performance. If desired, loss functions other than the worst ranking error ℓ in (5) can also be used.

Joint Framework: So far the recounting model and the detection model are separate hence not helping each other. To integrate them, we propose the following joint optimization framework:

$$\min_{W,R} \ \ell(W; R) + \tfrac{1}{2}\|V - R\|_{\mathsf{F}}^2 + \Phi(W) + \Omega(R), \tag{7}$$

where ℓ and Ω are given respectively in (5) and (2) above. By coupling the two models jointly in the ranking loss ℓ, we expect to exploit the mutual benefits of the two tasks. Indeed, the detection model works on the clean representation R that is supplied by the recounting model, avoiding any noisy or irrelevant concepts. Conversely, the detection model also directs the recounting model to find discriminative evidences that are tailored for detection. More conveniently, the joint optimization problem (7) is biconvex, meaning that fixing either one of W and R results in a convex problem that is immune to local minima. Thus we will use a coordinate descent algorithm to learn W and R, one at a time and iteratively. However, standard convex optimization toolboxes cannot be naively applied here for two reasons: 1). The training ensemble has a large size $c \times s \times m$, putting a stringent time complexity on the numerical algorithm (preferably linear-time); 2). The regularizer Ω is a highly non-smooth function, leading to extremely slow convergence if not properly handled. One obvious remedy for the first issue is to apply a pooling procedure to aggregate the s shot-level information, but it leads to a significant loss of temporal information. Instead, we circumvent the computational difficulties by developing a significantly improved alternating direction method of multipliers (ADMM) algorithm in the next section.

4. THE OPTIMIZATION SCHEME

In this section we develop a much improved linear time ADMM algorithm for our joint model (7). First, we rewrite the ranking loss ℓ by introducing an auxiliary variable A for decoupling:

$$\min_{W,R,A} \ g(A) + \tfrac{1}{2}\|V - R\|_{\mathsf{F}}^2 + \Phi(W) + \Omega(R), \tag{8}$$

$$\text{s.t.} \quad \forall i \in \mathfrak{P}, j \in \mathfrak{N}, \ A_{i,j} = 1 - \langle W, R^i - R^j \rangle, \tag{9}$$

where $g(A) = \frac{1}{p} \max_{j \in \mathfrak{N}} \sum_{i \in \mathfrak{P}} (A_{i,j})_+$. Next we introduce the Lagrangian multiplier matrix Γ and a quadratic penalty term to eliminate the linear constraint (9):

$$\begin{aligned} \min_{W,R,A} \ & \langle \Gamma, A + \mathcal{R}(W) - E \rangle + \tfrac{1}{2}\|A + \mathcal{R}(W) - E\|_{\mathsf{F}}^2 \\ & + \tfrac{1}{2}\|V - R\|_{\mathsf{F}}^2 + \Phi(W) + \Omega(R) + g(A), \end{aligned} \tag{10}$$

where E is the all 1's matrix and $\mathcal{R}(W) = \mathcal{W}(R)$ is the matrix whose ij-th entry is $\langle W, R^i - R^j \rangle$. We use $\mathcal{R}(W)$ to emphasize the linearity in W when R is fixed and use $\mathcal{W}(R)$ to highlight the linearity in R when W is fixed.

The optimization variables in (10) are now uncoupled, and the usual ADMM algorithm iteratively solves them one at a time. At a high level, we switch iteratively between the detection model (parameterized by W, A) and the recounting model (parameterized by R), until they collaboratively reach a consensus. Interestingly, the subproblems w.r.t. W, A and w.r.t. R are completely analogous, thus largely simplifying the subsequent developments. We discuss each subproblem in more details below, and address some severe computational challenges there.

4.1 Optimizing W while fixing R

In this step we fix R as a constant and solve for W and A. The updates from the vanilla ADMM algorithm are as follows:

$$W \leftarrow \arg\min_W \tfrac{1}{2}\|A + \mathcal{R}(W) - E + \Gamma\|_{\mathsf{F}}^2 + \Phi(W) \tag{11}$$

$$A \leftarrow \arg\min_A \tfrac{1}{2}\|A + \mathcal{R}(W) - E + \Gamma\|_{\mathsf{F}}^2 + g(A) \tag{12}$$

$$\Gamma \leftarrow A + \mathcal{R}(W) - E + \Gamma. \tag{13}$$

Unfortunately, executing the above steps is not easy (except for the Lagrangian multiplier Γ). Previous work [30] proposed two dedicated iterative subroutines for the W-step (11) and the A-step (12), respectively. However, the whole algorithm requires three nested loops, significantly slowing down the convergence. Moreover, the numerical errors in the inner loops may also accumulate and affect the outer loop. Essentially the same algorithm (hence same drawback) was adopted in [18]. Here we eliminate all inner loops based on two novel ideas.

Firstly, the difficulty in the W-step (11) is mostly due to the (non-diagonal) quadratic term induced by the linear map $\mathcal{R}(W)$ (recall that R is fixed). Fortunately, we can bypass this difficulty by linearizing the quadratic term at the current iterate W, which is known as the inexact Uzawa step in the ADMM literature [13]. Namely, we replace (11) with two steps:

$$\tilde{W} \leftarrow W - \tfrac{1}{\mu}\mathcal{R}^\top[A + \mathcal{R}(W) - E + \Gamma] \tag{14}$$

$$W \leftarrow \mathsf{P}_\Phi^\mu(\tilde{W}), \tag{15}$$

where the first step is a simple gradient update with step size $1/\mu$ while the second step is a proximal update with respect to the regularizer Φ. The latter, usually referred to as the proximal map [44], is defined for any function f with parameter μ as

$$\mathsf{P}_f^\mu(\mathbf{x}) = \arg\min_{\mathbf{z}} \tfrac{\mu}{2}\|\mathbf{x} - \mathbf{z}\|_2^2 + f(\mathbf{z}). \tag{16}$$

It is a strict and natural generalization of the usual Euclidean projection operator (where f is the indicator function of some constraint set). For "simple" functions, the proximal map admits a closed-form solution, for instance,

$$\mathsf{P}^{\mu}_{\Phi}(W)=\begin{cases}\frac{\mu}{\lambda+\mu}W, & \text{if } \Phi=\frac{\lambda}{2}\|\cdot\|_{\mathsf{F}}^2\\ \operatorname{sign}(W)*(|W|-\frac{\lambda}{\mu})_+, & \text{if } \Phi=\lambda\|\cdot\|_1\end{cases}, \tag{17}$$

where the algebraic operations are componentwise. Thus, the W-step (14) and (15) can now be performed in linear time, without the need of any nested loop at all.

Secondly, using the proximal notation in (16) we note that the A-step in (12) is simply the proximal update $\mathsf{P}^1_g(-\mathcal{R}(W)+E-\Gamma)$. Surprisingly, we prove here that this proximal update in fact admits a closed-form solution, which, to our best knowledge, has not been derived previously. Our result is based on the following new theorem, where we recall that $\mathbf{z}_+$ denotes the positive part, i.e., $(\mathbf{z}_+)_i=\max\{z_i,0\}$.

THEOREM 1. *If the function $f:\mathbb{R}^d\to\mathbb{R}\cup\{\infty\}$ satisfies*

$$\forall i,\ \forall\mathbf{z},\ \ f(\mathbf{z})\geq f(z_1,\ldots,z_{i-1},0,z_{i+1},\ldots,z_d), \tag{18}$$

then for all $\mu>0$,

$$\mathbf{w}-\mathbf{w}_++\mathbf{u}\in\arg\min_{\mathbf{z}}\tfrac{\mu}{2}\|\mathbf{w}-\mathbf{z}\|_2^2+f(\mathbf{z}_+) \tag{19}$$

$$\textit{if and only if}\quad \mathbf{u}\in\arg\min_{\mathbf{z}}\tfrac{\mu}{2}\|\mathbf{w}_+-\mathbf{z}\|_2^2+f(\mathbf{z}). \tag{20}$$

Note that Theorem 1 does not even require the function f to be convex: The condition (18) basically says that on each coordinate i, f attains its minimum at 0. Many familiar functions, for instance p-norms for all $p\geq 0$, satisfy this condition. The significance of the established equivalence in Theorem 1 is that it allows us to swap the positive part (a nonlinear operation) from an *optimization variable* $\mathbf{z}$ in (19) to a *fixed input variable* $\mathbf{w}$ in (20). This simple swapping can lead to enormous computational savings, especially when f is "simple". Indeed, for the A-step in (12), $g=f(A_+)$, where $f(A)=\frac{1}{p}\|A\|_{\infty,1}$ is the max norm of the 1-norms of each column of A. Since the proximal map $\mathsf{P}^{\mu}_{\|\cdot\|_{\infty,1}}$ has been derived in (nearly) closed-form in *e.g.* [29], applying Theorem 1 we immediately have a closed-form for the A-step (12):

$$A\leftarrow\mathsf{P}^1_g(\tilde{A})=\tilde{A}-\tilde{A}_++\mathsf{P}^p_{\|\cdot\|_{\infty,1}}(\tilde{A}_+), \tag{21}$$

where $\tilde{A}=-\mathcal{R}(W)+E-\Gamma$. The computational complexity is linear after sorting [29], without any nested loop at all. In §5.1 below we empirically verify that our closed-form solution (21) significantly improves the nested loops in [18, 30]. We expect Theorem 1 to be useful for other applications involving the infinite push ranking loss.

To summarize, for fixed R we can perform the update w.r.t. W and A in linear time in a single step. The same idea can be recycled to update R, which we discuss next.

4.2 Optimizing R while fixing W

This step is analogous to the previous step where W is optimized. For brevity we only mention the few differences. Again we linearize the quadratic term induced by $\mathcal{W}(R)=\mathcal{R}(W)$ at the current iterate R, and we perform the following updates (for some step size $\mu>1$):

$$\tilde{R}\leftarrow R-\tfrac{1}{\mu}\mathcal{W}^\top[A+\mathcal{W}(R)-E+\Gamma]+\tfrac{1}{\mu}(V-R) \tag{22}$$

$$R\leftarrow\mathsf{P}^{\mu}_{\Omega}(\tilde{R}) \tag{23}$$

$$A\leftarrow\arg\min_A\tfrac{1}{2}\|A+\mathcal{W}(R)-E+\Gamma\|_{\mathsf{F}}^2+g(A) \tag{24}$$

$$\Gamma\leftarrow A+\mathcal{W}(R)-E+\Gamma, \tag{25}$$

where the first step is a simple gradient update; the third A-step is solved using (21) as before; the fourth Γ-step is also straightforward (standard matrix operations). The second proximal update w.r.t. the regularizer Ω in our recounting model is more involved, and has not been considered in previous work. Fortunately, we prove that it can still be easily computed by decomposing the individual regularizers as follows:

Algorithm 1: Faster ADMM for solving JEDaR (7)

```
Input: $V,\alpha,\beta,\gamma,p,\Phi$. Initialize: $W,R,A,\Gamma,\mu>1$.
repeat
    $W\leftarrow W-\frac{1}{\mu}\mathcal{R}^\top[A+\mathcal{R}(W)-E+\Gamma]$;
    $W\leftarrow\mathsf{P}^{\mu}_{\Phi}(W)$ ;                    // Equation (17)
    $A\leftarrow-\mathcal{R}(W)+E-\Gamma$;
    $A\leftarrow A-A_++\mathsf{P}^p_{\|\cdot\|_{\infty,1}}(A_+)$;
    $\Gamma\leftarrow A+\mathcal{R}(W)-E+\Gamma$;
    $R\leftarrow R-\frac{1}{\mu}\mathcal{W}^\top[A+\mathcal{W}(R)-E+\Gamma]+\frac{1}{\mu}(V-R)$;
    $R\leftarrow\mathsf{P}^{\mu}_{\alpha\|\cdot\|_{2,1}}\Big(\mathsf{P}^{\mu}_{\beta\|\cdot\|_1}\big(\mathsf{P}^{\mu}_{\gamma\|\cdot\|_{\mathsf{tv}}}(R)\big)\Big)$;
until convergence;
```

THEOREM 2. $\mathsf{P}^{\mu}_{\Omega}(R)=\mathsf{P}^{\mu}_{\alpha\|\cdot\|_{2,1}}\Big(\mathsf{P}^{\mu}_{\beta\|\cdot\|_1}\big(\mathsf{P}^{\mu}_{\gamma\|\cdot\|_{\mathsf{tv}}}(R)\big)\Big)$.

Very crucially, all three proximal maps on the right-hand side of Theorem 2 have known exact linear time algorithms. By composing them we immediately obtain the proximal map for the sum regularizer Ω, completing the R-step in (23).

To summarize, for fixed W we can again update w.r.t. R in linear time in a single step, without any nested loop at all.

4.3 Combining the W and R steps

For fixed R, iterating (14), (15), (21), (13) by at most $O(\frac{1}{\epsilon})$ steps yields an ϵ-optimal solution for W [13]. Similarly, for fixed W, iterating (22), (23), (24), (25) by at most $O(\frac{1}{\epsilon})$ steps yields an ϵ-optimal solution for R. The two procedures can be alternated until convergence, however, we notice that they share the same A-step and Γ-step. Therefore it seems reasonable to combine the two procedures into one. Another way of thinking about the combination is that since we are alternating the two procedures it would be wasteful to wait until each procedure completely converges. Instead, we simply run each procedure for a *single* iteration and then switch to another immediately. As we observed in the experiments, this "eager" switching strategy works very well.

We summarize the combined and improved ADMM algorithm for solving our joint training protocol (7) in Algorithm 1, and we wish to point out that Algorithm 1 is very general and accommodates various regularizers Φ (*e.g.* those in (17)). Computationally, Algorithm 1 is very appealing since each iteration incurs only a cost linear in the size of the training data. To appreciate the efficiency of our algorithm, we compared it with the full algorithm in [30] (whose model requires $\alpha=\beta=\gamma=0$ and $\Phi=\lambda\|\cdot\|_1$). In general 400x speedups were achieved even on moderate problem sizes (*e.g.* $m=800, cs=100$). This tremendous speedup is the key for us to train our joint framework on large real multimedia datasets.

5. EXPERIMENTS

In this section we conduct thorough experimental evaluations of the proposed **J**oint **E**vent **D**etection **a**nd **R**ecounting framework, abbreviated as **JEDaR**.

Datasets: We test on two real-world datasets: the TRECVID MEDTest 2013 [25] and the TRECVID MEDTest 2014 [26]. Both are collected by the NIST for the TRECVID competition. Each dataset consists of 20 complex events with 10 events in common. Specifically, the MEDTest 2013 dataset has events E006 to E015 and E021 to E030, while the MEDTest 2014 contains events E021 to E040. These events include *changing a vehicle tire*, *grooming an animal*, *etc*. Please refer to [25, 26] for the complete list of event names.

Setup: For all our experiments we strictly follow the ***10Ex evaluation procedure*** outlined by the NIST TRECVID event kit. According to the rules, we ***detect each event separately***, totaling 20 individual detection tasks for each dataset. For each event, we have ***10*** positive videos from the event kit training data, along with approximately 5,000 negative videos from the background training data. The testing data has about 23,000 videos. We report both event detection and recounting results for each event.

Competitors: We compare our method JEDaR against current state-of-the-art alternatives. Support Vector Machine (SVM) and Ridge Regression (RR) are the most widely used classifiers in the TRECVID MED 2014 competition among the top ranked teams and recent research reports. Therefore, the two algorithms are used as the baseline. We also compare to more recent state-of-the-art alternative methods, including dynamic pooling with segment-pairs (DPSP) [19], VD-HMM [37], and ELM [34]. All comparisons follow the same rules of the official TRECVID MEDTest 2013 and MEDTest 2014 data splits.

Concept detectors: 3,135 concept detectors are pre-trained using the TRECVID SIN dataset (346 categories), google sports (478 categories) [15, 17], ucf101 dataset (101 categories) [15, 33], YFCC dataset (609 categories) [15, 38] and DIY dataset (1,601 categories) [15, 43]. The improved dense trajectory features are extracted with the code provided in [40] and encoded them with the fisher vector representation [28]. Then, on top of the extracted low-level features the cascade SVM [9] was trained for each concept detector. We further split each video in the TRECVID MEDTest 2013 and 2014 datasets into s shots, using the color histogram difference as an indication of the boundary. We applied the pre-trained concept detectors to each shot and obtained a 3135-dimensional representation. The average accuracy of the concept detectors is relatively low due to the small number of positive training examples, therefore justifying our recounting model (1) which tries to *denoise* the noisy concept representations.

Parameter Tuning: Our Algorithm 1 has a few parameters and we tune them as follows. We use the recounting model (1) alone, without coupling with the detection model (2), to first tune the regularization constant α so that roughly $5 \sim 10$ rows of the semantic representation R are nonzero. Then we further tune β so that each row contains roughly 10 nonzero entries. These two parameters are fixed in our subsequent experiments. Next we cross-validate the parameter λ from the range $\{0.01, 0.1, 1, 10, 100\}$, and similarly for the parameter γ from the same range. We have conducted a sensitivity analysis and found the results to be relatively robust once the parameters are in a reasonably large region. The parameters for other competing algorithms are set according to their respective implementations.

5.1 Efficiency of our closed-form solution

We first demonstrate the efficiency of our ADMM Algorithm 1, where the key is Theorem 1 that provides a closed-form solution in (21) for the A-step in (12). Previous work [18, 30] dedicated a nested loop iterative subroutine for this step. We randomly generated the input (fixed) square matrix W with varying sizes, and compared the objective values in the A-step (12) (the smaller the better). In this section we set $\alpha = \beta = \gamma = 0$ since this is the setting that [18, 30] can handle, although our algorithm efficiently extends to any α, β, γ, thanks to the new Theorem 2. We used the default setting in [30]: The maximum number of outer and inner iterations are 200 and 50, respectively. Figure 2 confirms the huge computational saving we enjoy thanks to Theorem 1. As can be seen from Figure 2, the running time of [30] increased sharply with the input size, but still failed to converge to the optimum. In comparison, our closed-form solution (21) only took a negligible time and returned a much smaller (in fact optimal) objective value. For the full ADMM algorithm, the difference is even larger, since we chose to linearize the W-step while [18, 30] used another nested loop. This tremendous speedup is the key to train our joint framework on the large TRECVID MEDTest datasets, especially when we segment each video into multiple shots to exploit temporal information and to perform recounting (see below).

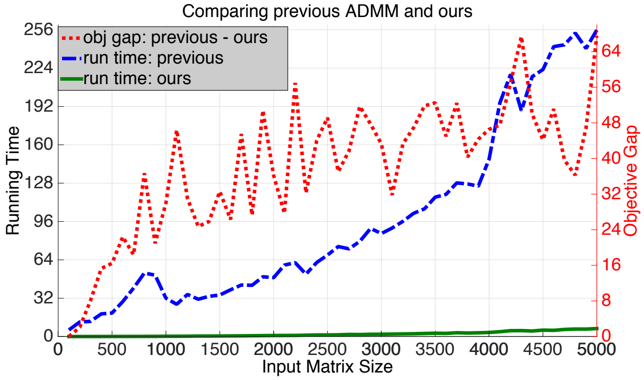

Figure 2: Comparison between our closed-form solution (21) and the previous nested loop iterative subroutine in [30]. Even after spending significantly more time (blue dashed vs. solid green), the latter still has not converged yet (red dot).

5.2 Event Detection Result

As the common practice, we evaluate the event detection performance using the (mean) average precision. The results of our method JEDaR and the competitors (quoted from the respective papers) on both the MEDTest 2013 and MEDTest 2014 datasets are recorded in Table 1. It is clear that the proposed method JEDaR performs the best on both datasets. Comparing against the SVM baseline, we see that JEDaR significantly improves the detection performance for nearly all events, with mAP of 30.95% vs SVM's 20.73% on the MEDTest 2013 dataset and mAP of 25.21% vs 18.32% on the MEDTest 2014 dataset. We attribute the improvement to our joint training framework that integrates recounting with detection. Thanks to our recounting model, many irrelevant noisy concepts are pruned away, which in turn largely improves the detection performance.

Next we discuss the state-of-the-art alternatives that we were able to compare against. From Table 1, we have the following observations: 1). RR performs similar to SVM for most events, achieving for instance mAP of 18.78% vs 18.32% on the MEDTest 2014 dataset. This is in accordance with past experiences of several research groups in the TRECVID MED competition. 2). DPSP [19] improves RR (and SVM), with mAP 20.47% vs 18.78% on the MEDTest 2014 dataset. This is probably because DPSP identifies the segments that are most informative for detecting a given event and also dynamically determines the pooling operator most suited for each sequence. 3). VD-HMM [37] further improves the performance by discovering and assigning sequences of states that are most discriminative for a given event. Note that none of the above methods considered detection and recounting in a joint

Table 1: Experimental comparisons for 10Ex event detection on TRECVID MEDTest 2013 and TRECVID MEDTest 2014 datasets. Mean average precision (mAP) is used as the evaluation metric. Results are presented in percentages. Larger mAP indicates better performance. From the results we observe that the proposed algorithm outperforms the state-of-the-art competitors on both datasets, indicating the superiority of the proposed JEDaR.

	MEDTest 2013					
Event ID	SVM	RR	DPSP	VDHMM	ELM	JEDaR
E006	26.54	27.68	31.29	35.74	38.25	44.66
E007	38.75	39.85	44.48	50.86	54.39	57.63
E008	47.29	48.78	54.82	61.62	65.17	64.22
E009	37.76	36.82	38.75	41.88	45.54	47.94
E010	17.12	17.47	20.64	23.56	26.38	28.49
E011	8.88	9.53	11.28	13.34	14.29	16.77
E012	25.34	25.57	30.66	34.82	38.14	39.38
E013	56.82	57.24	62.53	67.44	66.58	68.25
E014	37.76	38.11	42.48	47.26	47.41	52.33
E015	17.68	18.72	22.69	27.85	31.82	35.34
E021	11.96	12.58	11.24	15.57	14.75	19.53
E022	2.76	2.87	2.72	4.84	7.54	8.77
E023	27.47	27.72	30.75	38.32	41.29	46.26
E024	3.17	2.98	3.12	2.86	3.16	3.98
E025	0.81	0.92	0.87	1.05	1.12	1.27
E026	4.16	4.48	4.85	5.42	5.88	6.23
E027	10.58	10.29	12.56	11.25	13.96	15.62
E028	16.91	18.64	20.82	22.43	25.25	27.41
E029	11.78	12.47	14.28	15.12	17.84	19.63
E030	10.96	11.26	13.65	14.88	15.92	15.26
mean	20.73	21.19	23.72	26.81	28.73	30.95

	MEDTest 2014					
Event ID	SVM	RR	DPSP	VDHMM	ELM	JEDaR
E021	11.96	12.58	11.24	15.57	14.75	19.53
E022	2.76	2.87	2.72	4.84	7.54	8.77
E023	27.47	27.72	30.75	38.32	41.29	46.26
E024	3.17	2.98	3.12	2.86	3.16	3.98
E025	0.81	0.92	0.87	1.05	1.12	1.27
E026	4.16	4.48	4.85	5.42	5.88	6.23
E027	10.58	10.29	12.56	11.25	13.96	15.62
E028	16.91	18.64	20.82	22.43	25.25	27.41
E029	11.78	12.47	14.28	15.12	17.84	19.63
E030	10.96	11.26	13.65	14.88	15.92	15.26
E031	55.98	56.74	61.82	64.24	67.85	69.41
E032	22.89	23.57	24.64	26.48	27.43	28.28
E033	38.74	39.85	41.47	44.28	46.54	46.27
E034	25.88	26.59	27.83	28.74	29.78	31.63
E035	34.85	35.28	38.86	40.63	42.86	45.32
E036	13.28	13.75	15.28	16.45	16.52	19.27
E037	33.15	33.74	35.87	36.12	47.81	49.25
E038	0.92	0.85	0.78	0.91	1.14	1.43
E039	26.94	27.36	31.43	23.22	28.72	30.58
E040	13.25	13.67	16.54	21.41	15.83	18.62
mean	18.32	18.78	20.47	21.71	23.56	25.21

framework. 4). Similar to our method, ELM [34] also considers detection and recounting jointly, and it achieves the second best performance, *e.g.*, with mAP 23.56% on the MEDTest 2014. However, it explicitly models the evidence locations which involve a large number of latent variables, making learning less efficient especially when training data is scarce. Overall, the advantage of DPSP, VD-HMM and ELM over RR and SVM suggests that finding important segments generally leads to better performance. 5). Lastly, the proposed method JEDaR achieves the best performance on both datasets. This confirms the benefits of our joint training framework, in which recounting improves detection by pruning irrelevant noisy concepts while detection directs recounting to the most discriminative evidences.

5.3 Event Recounting Result

In the TRECVID Multimedia Event Recounting (MER) task, a video description is defined as a video shot with a starting frame, an ending frame and a textual description. We use the proposed method to obtain such results as follows: We average the SVM weight W along the shot dimension and pick few top concepts for each event. These concepts are highly discriminative. Then, for each shot of the test video, we examine its top concept scores and decide if it is useful for recounting. The evaluation of event recounting is not easy, since a). there is no ground truth information, and b). few previous works can be compared with. Here we compare to the natural baseline which conducts detection and recounting in separate stages. Following the NIST evaluation pipeline [25, 26], we invited 10 volunteers to serve as judges. Before evaluation, we showed each judge the event category descriptions in text, in addition to five positive videos in the training set. Then, we randomly picked 10 positive videos from the test set, and presented informative shots generated by the baseline and the proposed method. The judges were asked to determine which shots are more discriminative. To make a fair comparison, we do not tell the judges which shot is generated by which method during evaluation. Two evaluation metrics are employed: 1). average accuracy, which measures the percentage of correctly labeled shots; and 2). relative performance, which counts judges' preferences of the baseline or the proposed method.

Table 2: Average video length, average shot length generated by the proposed method, and average accuracy derived from the judges' labels.

	MEDTest 2013	MEDTest 2014
Average Video Length	164.3 seconds	188.4 seconds
Average Shot Length	6.4 seconds	8.6 seconds
Average Accuracy	89.7 %	83.2 %

Table 2 summarizes the average length of the test videos, the average length of shots generated by the proposed method, and the average accuracy derived from the judges' labels. We observe that the proposed method achieves 89.7% and 83.2% average accuracy by selecting only 3.8% and 4.5% of shots in the original videos, respectively. This clearly demonstrates that our method is capable of localizing reasonable evidences that are accountable to the (positive) detection outcome and are comprehensible to humans. The results seem to indicate that the MEDTest 2014 dataset is more challenging than the MEDTest 2013 dataset.

The judges' preferences between the proposed method and the baseline are averaged and recorded in Table 3. It is clear that the proposed method is subjectively better for most of the events on both datasets. To verify, we show some of the recounting results in Figure 3 for the MEDTest 2014 dataset. These results again confirm that the proposed method successfully localizes the key evidences that also match human's intuition, for instance, as one would expect, "horse", "show jumping" and "animal" are all very indicative evidences for the *horse riding competition* event.

5.4 Sensitivity Analysis

We conduct some sensitivity analysis in this section, to draw further insights of the proposed joint training framework.

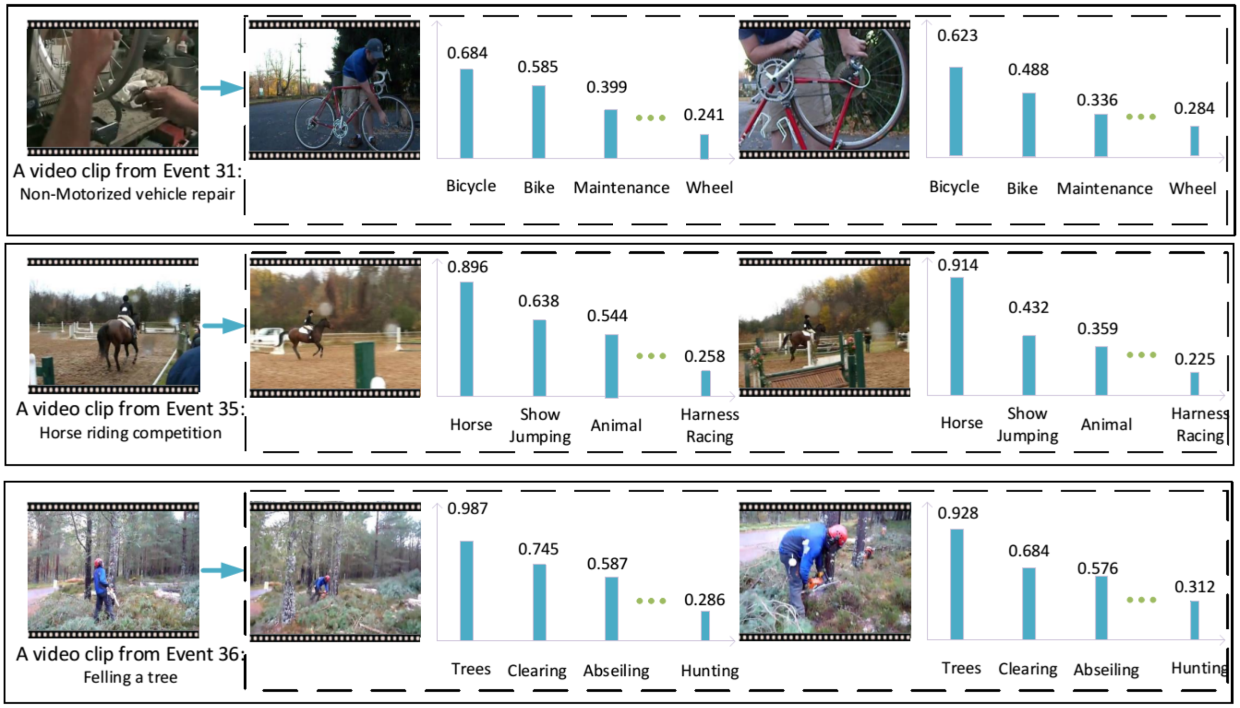

Figure 3: Example recounting results generated by the proposed method on the TRECVID MEDTest 2014 dataset. The video events are *non-motorized vehicle repair*, *horse riding competition* and *felling a tree*. The first two relevant shots are displayed.

Table 3: Event recounting results on the TRECVID MEDTest 2013 and 2014 datasets. For each event, we randomly pick 10 positive test videos and ask 10 judges to rate *better, similar, or worse* between the proposed method and the baseline. The results among judges are aggregated via averaging.

MEDTest 2013				MEDTest 2014			
ID	Better	Worse	Similar	ID	Better	Worse	Similar
E006	6	2	2	E021	8	1	1
E007	8	1	1	E022	2	5	3
E008	7	2	1	E023	4	4	2
E009	5	2	3	E024	7	0	3
E010	9	1	0	E025	5	2	3
E011	7	1	2	E026	6	2	2
E012	5	2	3	E027	7	2	1
E013	3	6	1	E028	5	3	2
E014	7	1	2	E029	4	2	4
E015	3	1	6	E030	3	1	6
E021	8	1	1	E031	8	0	2
E022	2	5	3	E032	6	2	2
E023	4	4	2	E033	6	1	3
E024	7	0	3	E034	5	0	5
E025	5	2	3	E035	3	1	6
E026	6	2	2	E036	4	1	5
E027	7	2	1	E037	6	2	2
E028	5	3	2	E038	3	6	1
E029	4	2	4	E039	3	0	7
E030	3	1	6	E040	7	1	2
Total	111	41	48	Total	102	36	62

Effects of Ω in recounting model: We first study the influence of different norms in the sparse regularizer Ω on MEDTest 2014 dataset. Recall that $\Omega(R) = \alpha\|R\|_{2,1} + \beta\|R\|_1 + \gamma\sum_i \|R^i\|_{\text{tv}}$ is used in our recounting model to enforce different structured sparse information about the evidences. We drop one of the three terms in Ω and record its influence on the detection accuracy. In details, we compare: a). without the group norm, *e.g.* $\alpha = 0$; b). without the ℓ_1 norm, *e.g.* $\beta = 0$; and c). without the total variation norm, *e.g.* $\gamma = 0$. The results are summarized in Figure 4. We see that the full method (*i.e.*, none of α, β, γ is set 0) consistently performs the best, indicating the usefulness of all three sparse regularizers. Dropping the total variation norm (*i.e.* $\gamma = 0$) deteriorates the performance the most, indicating the utter importance of temporal smoothness. Dropping the group norm (*i.e.* $\beta = 0$) affects the performance the least (on average). This is because the ℓ_1 norm also has the effect of zeroing out irrelevant concepts. The comparison against the separate training model, *i.e.* $\alpha = \beta = \gamma = 0$, is performed below. Overall the benefits of incorporating these sparse regularizers in our joint detection and recounting framework are significant.

Effects of Φ in detection model: As discussed in Section 3.2, the proposed algorithm directly applies to different regularizers Φ in the infinite push SVM detection model (6), including the usual (squared) ℓ_2-norm and the sparse ℓ_1-norm. We conduct experiments to compare their performances on MEDTest 2014 dataset. The results are summarized in Figure 5. We observe that $\Phi = \ell_2^2$ outperforms $\Phi = \ell_1$ for all events, leading to a significantly higher mAP. This is because the "clean" representation R is already sought to be sparse in our recounting model, thus imposing further sparsity in the detection model often hurts the detection performance. On the other hand, the learned SVM weight W for $\Phi = \ell_1$ is much sparser than $\Phi = \ell_2^2$, and may be advantageous when testing time is a bigger concern.

Joint vs. Separate: In our last experiment we validate the effectiveness of the proposed joint detection and recounting framework. By setting $\alpha = \beta = \gamma = 0$, we obtain a separate model that first performs event detection then followed by recounting. The results, summarized in Table 4, clearly demonstrate that the proposed joint training method consistently outperforms the separate training method on all events, usually by a large margin. We attribute this significant improvement to the coupling between detection and recounting: recounting helps detection by pruning irrelevant noisy concepts while detection directs recounting to more discriminative concepts.

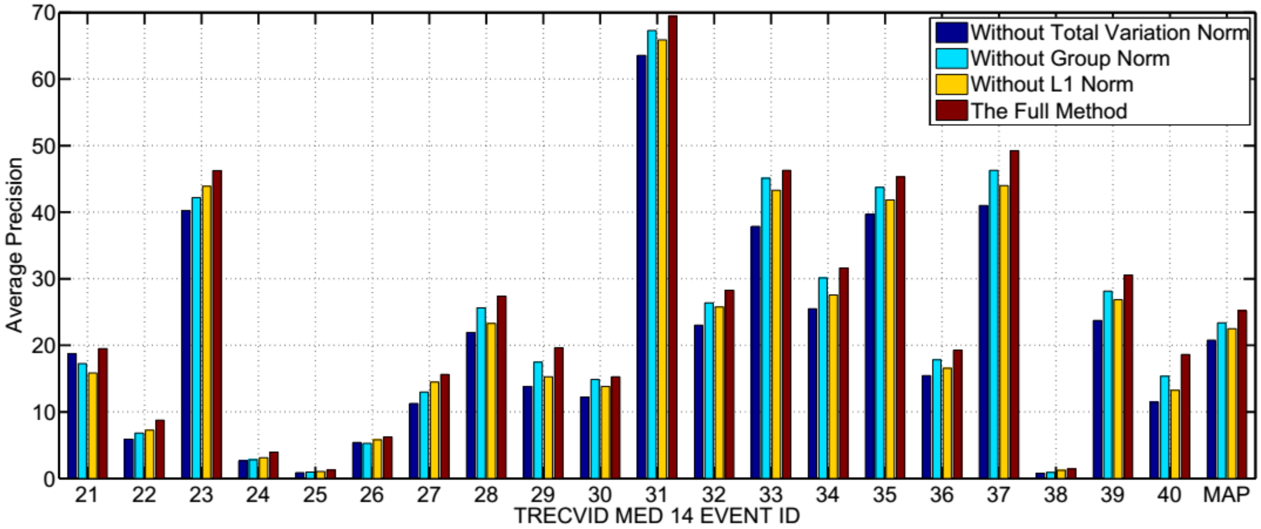

Figure 4: The average precisions of different sparse regularizers Ω on the TRECVID MEDTest 2014 dataset. Comparisons are made among our method by a) dropping group norm ($\alpha = 0$); b) dropping the ℓ_1 norm ($\beta = 0$); c) dropping total variation norm ($\gamma = 0$); and d) the full method.

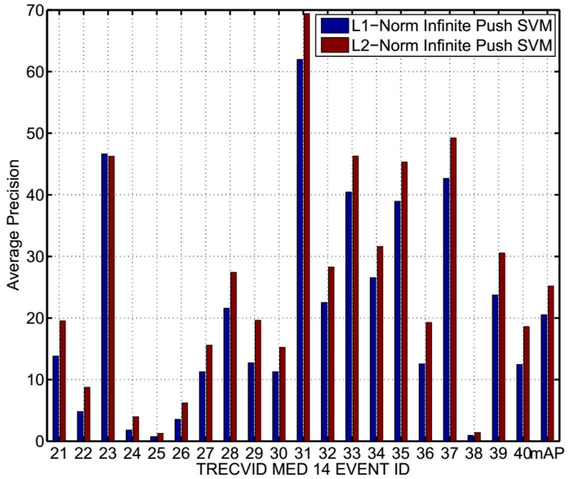

Figure 5: Performance comparison for the infinite-push SVM with $\Phi = \ell_2^2$ and $\Phi = \ell_1$ on the TRECVID MEDTest 2014 dataset. Results are presented in percentages.

6. CONCLUSION

We have proposed a novel joint training protocol to *simultaneously* conduct event detection and recounting. Based on a noisy semantic video representation, we couple a recounting model which aims at localizing the key evidences with a detection model which aims at enhancing the discriminative power. The recounting model assists detection by filtering out noisy irrelevant information while simultaneously the detection model guides recounting by directing it to the most discriminative evidences, conceptual-wise and temporal-wise. The two models are optimized *jointly* hence benefit greatly from each other. To address the computational challenge due to operating on the shot level, we significantly improve the existing ADMM algorithm by proving closed-form solutions for the intermediate proximal updates. Augmented with the Uzawa linearization trick we are able to remove all inner loops in previous ADMM implementations, without affecting the convergence properties at all. The proposed method is tested on the large-scale and challenging TRECVID MEDTest 2013 and 2014 datasets, achieving very promising results in both detection and recounting. In the future we intend to incorporate domain knowledge and supplementary information such as text and audio.

Table 4: Performance comparison of the separate training method (*i.e.* event detection only with $\alpha = \beta = \gamma = 0$) and our joint training method. Mean average precision (mAP) is used as the evaluation metric. Results are presented in percentages. A larger mAP indicates better performance.

	MEDTest 2013			MEDTest 2014	
ID	Separate	Joint	ID	Separate	Joint
E006	38.25	44.66	E021	12.68	19.53
E007	49.26	57.63	E022	5.86	8.77
E008	57.45	64.22	E023	41.35	46.26
E009	41.58	47.94	E024	2.73	3.98
E010	22.69	28.49	E025	0.76	1.27
E011	11.58	16.77	E026	4.46	6.23
E012	31.86	39.38	E027	10.26	15.62
E013	61.59	68.25	E028	21.48	27.41
E014	43.87	52.33	E029	11.64	19.63
E015	27.43	35.34	E030	9.58	15.26
E021	12.68	19.53	E031	61.58	69.41
E022	5.86	8.77	E032	22.57	28.28
E023	41.35	46.26	E033	39.58	46.27
E024	2.73	3.98	E034	25.57	31.63
E025	0.76	1.27	E035	38.58	45.32
E026	4.46	6.23	E036	12.74	19.27
E027	10.26	15.62	E037	41.84	49.25
E028	21.48	27.41	E038	0.98	1.43
E029	11.64	19.63	E039	24.86	30.58
E030	9.58	15.26	E040	13.98	18.62
mean	25.32	30.95	mean	21.28	25.21

Acknowledgment

We thank the reviewers for their valuable comments. This work was partially supported by the US Department of Defense, the U.S. Army Research Office (W911NF-13-1-0277), partially supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20068, and partially supported by the ARC DECRA project. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, ARO, or the U.S. Government.

References

[1] S. Agarwal. The infinite push: A new support vector ranking algorithm that directly optimizes accuracy at the absolute top of the list. In *SDM*, 2011.

[2] A. Barbu et al. Video in sentences out. In *UAI*, 2012.

[3] S. Bhattacharya, F. X. Yu, and S.-F. Chang. Minimally needed evidence for complex event recognition in unconstrained videos. In *ICMR*, 2014.

[4] X. Chang, Y. Yang, A. G. Hauptmann, E. P. Xing, and Y.-L. Yu. Semantic concept discovery for large-scale zero-shot event detection. In *IJCAI*, 2015.

[5] X. Chang, Y. Yang, E. P. Xing, and Y.-L. Yu. Complex event detection using semantic saliency and nearly-isotonic SVM. In *ICML*, 2015.

[6] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, 2001.

[7] P. Das, C. Xu, R. Doell, and J. Corso. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *CVPR*, 2013.

[8] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *CVPR*, 2009.

[9] H. P. Graf, E. Cosatto, L. Bottou, I. Durdanovic, and V. Vapnik. Parallel support vector machines: The cascade SVM. In *NIPS*, 2004.

[10] A. Gupta, P. Srinivasan, J. Shi, and L. S. Davis. Understanding videos, constructing plots: Learning a visually grounded storyline model from annotated videos. In *CVPR*, 2009.

[11] A. Habibian, K. E. van de Sande, and C. G. Snoek. Recommendations for video event recognition using concept vocabularies. In *ICMR*, 2013.

[12] A. Hauptmann, R. Yan, W.-H. Lin, M. Christel, and H. Wactlar. Can high-level concepts fill the semantic gap in video retrieval? a case study with broadcast news. *IEEE Transactions on Multimedia*, 9(5):958–966, 2007.

[13] B. He and X. Yuan. On the $O(1/n)$ convergence rate of Douglas-Rachford alternating direction method. *SIAM Journal on Numerical Analysis*, 50(2):700–709, 2012.

[14] N. Ikizler-Cinbis and S. Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. In *ECCV*, 2010.

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

[16] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah. High-level event recognition in unconstrained videos. *International Journal of Multimedia Information Retrieval*, 2(2):73–101, 2013.

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

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

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

[20] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[21] J. Liu, Q. Yu, O. Javed, S. Ali, A. Tamrakar, A. Divakaran, H. Cheng, and H. Sawhney. Video event recognition using concept attributes. In *WACV*, 2013.

[22] Z. Ma, Y. Yang, Y. Cai, N. Sebe, and A. G. Hauptmann. Knowledge adaptation for ad hoc multimedia event detection with few exemplars. In *ACM MM*, 2012.

[23] M. Mazloom, A. Habibian, and C. G. Snoek. Querying for video events by semantic signatures from few examples. In *ACM MM*, 2013.

[24] M. Merler, B. Huang, L. Xie, G. Hua, and A. Natsev. Semantic model vectors for complex video event recognition. *IEEE Transactions on Multimedia*, 14(1):88–101, 2012.

[25] NIST. The TRECVID MED 2013 evaluation plan. http://nist.gov/itl/iad/mig/med13.cfm, 2013.

[26] NIST. The TRECVID MED 2014 evaluation plan. http://nist.gov/itl/iad/mig/med14.cfm, 2014.

[27] D. Oneaţă, J. Verbeek, and C. Schmid. Action and event recognition with fisher vectors on a compact feature set. In *ICCV*, 2013.

[28] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010.

[29] A. Quattoni, X. Carreras, M. Collins, and T. Darrell. An efficient projection for $l_{1,\infty}$ regularization. In *ICML*, 2009.

[30] A. Rakotomamonjy. Sparse support vector infinite push. In *ICML*, 2012.

[31] C. Rudin. The p-norm push: A simple convex ranking algorithm that concentrates at the top of the list. *Journal of Machine Learning Research*, 10:2233–2271, 2009.

[32] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[33] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild, 2012.

[34] C. Sun and R. Nevatia. DISCOVER: Discovering important segments for classification of video events and recounting. In *CVPR*, 2014.

[35] A. Tamrakar, S. Ali, Q. Yu, J. Liu, O. Javed, A. Divakaran, H. Cheng, and H. Sawhney. Evaluation of low-level features and their combinations for complex event detection in open source videos. In *CVPR*, 2012.

[36] C. C. Tan, Y.-G. Jiang, and C.-W. Ngo. Towards textually describing complex video contents with audio-visual concept classifiers. In *ACM MM*, 2011.

[37] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. In *CVPR*, 2012.

[38] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L. Li. The new data and new challenges in multimedia research. *CoRR*, abs/1503.01817, 2015.

[39] C. Tsai, M. L. Alexander, N. Okwara, and J. R. Kender. Highly efficient multimedia event recounting from user semantic preferences. In *ICMR*, 2014.

[40] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013.

[41] Y. Yang, Z. Ma, Z. Xu, S. Yan, and A. G. Hauptmann. How related exemplars help complex event detection in web videos? In *ICCV*, 2013.

[42] B. Yao, X. Jiang, A. Khosla, A. Lin, L. Guibas, and L. Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *ICCV*, pages 1331–1338, 2011.

[43] S. Yu, L. Jiang, and A. G. Hauptmann. Instructional videos for unsupervised harvesting and learning of action examples. In *ACM MM*, 2014.

[44] Y. Yu. On decomposing the proximal map. In *NIPS*, 2013.

[45] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68:49–67, 2006.