Adaptive Pooling in Multi-Instance Learning for Web Video Annotation

Yizhou Zhou[1,2] Xiaoyan Sun[2] Dong Liu[1] Zhengjun Zha[1] Wenjun Zeng[2]
[1]University of Science and Technology of China [2]Microsoft Research Asia
{dongeliu,zhazj}@ustc.edu.cn {v-yizzh,xysun,wezeng}@microsoft.com

Abstract

Web videos are usually weakly annotated, i.e.*, a tag is associated to a video once the corresponding concept appears in a frame of this video without indicating when and where it occurs. These weakly annotated tags pose big troubles to many Web video applications,* e.g. *search and recommendation. In this paper, we present a new Web video annotation approach based on multi-instance learning (MIL) with a learnable pooling function. By formulating the Web video annotation as a MIL problem, we present an end-to-end deep network framework to solve this problem in which the frame (instance) level annotation is estimated from tags given at the video (bag of instances) level via a convolutional neural network (CNN). A learnable pooling function is proposed to adaptively fuse the outputs of the CNN to determine tags at the video level. We further propose a new loss function that consists of both bag-level and instance-level losses, which enables the penalty term to be aware of the internal state of network rather than only an overall loss, thus makes the pooling function learned better and faster. Experimental results demonstrate that our proposed framework is able to not only enhance the accuracy of Web video annotation by outperforming the state-of-the-art Web video annotation methods on the large-scale video dataset FCVID, but also help to infer the most relevant frames in Web videos.*

1. Introduction

Video annotation, also known as video tagging, is an essential but challenging problem especially for Web videos. It plays a crucial role in orgnizing and accessing large-scale video collections [6]. However, Web videos are usually weakly annotated. They are often contributed by end users and thus lack regular metadata and/or descriptive text. Compared with many carefully annotated video datasets in which only a few representative tags are associated to each short video, a tag can be assigned to a Web video as long as a frame in this video reveals the related concept but where and when the concept occurs is not provided.

Existing work on video annotation can be roughly classified into two categories, temporal-aware and frame-based methods. Temporal-aware methods [14, 20, 37] exploit temporal correlations among video frames to recognize actions and events identifiable with motion information. Frame-based methods, on the other hand, focus on contents in individual frames since many important concepts can be well determined by looking at individual frames of a video without using temporal correlations. Examples include objects, scenes, and lots of actions like eat, sit, and so on. Thus frame-based methods are proposed to annotate videos based on concepts in individual frames (usually keyframes) and achieve promising results at relatively lower computational cost [47]. In this paper, we focus on the frame-based approach to annotate Web videos for both accuracy and efficiency.

Annotation related work has been investigated with regard to the characteristics of Web videos. The vast quantity of near-duplicated videos are explored for both video tagging [41, 5] and retagging [6]. Tagging of foreground moving object is designed in [44] for the uncontrolled Web video. Another group of methods manages to take advantage of resourceful complementary information to help Web video annotation [54, 55, 5]. Also, semantic events besides visual concepts are studied for detection [53] and summarization [47] of long Web videos. However, to the best of our knowledge, there is only a few work focusing on the annotation problem directly from the weakly annotated Web videos, especially based on deep network frameworks.

In this paper, we present a new Web video annotation method by formulating the annotation as a multi-instance learning (MIL) problem since Web video is less temporal-aware and its tags are coarsely grained. Inspired by the recent advances of deep network models, we propose a deep network framework to solve the MIL problem of frame-based video annotation. The network accepts frames of a video as inputs and processes each frame individually and

This work was supported by the Natural Science Foundation of China (NSFC) under Grants 61331017 and 61390512, and by the Fundamental Research Funds for the Central Universities under Grant WK3490000001. It was done when Yizhou Zhou was an intern at Microsoft Research.

identically by a convolutional neural networks(CNN)[24]. Later on, outputs at frame level are fused to determine the relevant tags at the video level and further refine the frame level prediction. The fusion is framed as pooling in deep network models which essentially bridges the gap between instance-level learning and bag-level supervision. The refinement is executed internally through our proposed bag+instance loss function.

We address the fusion problem in Web video annotation by proposing a learnable pooling function that is introduced as a new learnable pooling layer in deep networks for end-to-end training. Moreover, a new loss function is presented that consists of both bag-level and instance-level losses, and the latter is estimated from instance-level outputs of the network. The new loss function contributes not only to the learning of pooling function, but also to the quick convergence of the training. We show by experiments that our proposed new pooling layer and new loss function together help to improve the accuracy of Web video annotation and meanwhile are able to identify the most relevant frames. Our proposed method outperforms the state-of-the-art annotation methods (even with much more complex networks that use temporal correlation and/or multimodal including audio information on the large scale Web video dataset FCVID [19].

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 presents the MIL deep network for Web video annotation followed by the learnable pooling and new bag+instance loss function. Section 4 shows experimental results. At last, Section 5 concludes this paper.

2. Related Work

In this section, related work on video annotation, MIL deep networks, and pooling functions is discussed.

Video Annotation/Classification. Much progress has been made on video annotation recently. Most of the work follows a standard pipeline, *i.e*. kinds of features are extracted from videos and then fed into classifiers to generate tags. Depending on whether the extracted features embed temporal correlations, existing methods can be roughly categorized into temporal-aware and frame-based ones. Temporal-aware methods often focus on the design of temporal features, such as spatial-temporal interest points (STIP) [25] and trajectory-based descriptors [45], whereas frame-based methods explore visual features developed for still images [47]. Recent work focuses more on deep network models [20, 37] rather than the classic support vector machine (SVM) based solutions [39, 34, 48]. Karpathy *et al*. extend the connectivity of a CNN in time domain to take advantage of local spatio-temporal information and suggest a multi-resolution architecture for video classification [20]. Simonyan and Zisserman propose a two-stream CNN approach that extracts features from both static frames and motion optical flow separately [37].

For Web video annotation, Sun *et al*. present consensus foreground object templates for videos captured by freely-moving cameras at low resolutionn [44]. Chen *et al*. introduce a video retagging approach through both visual and textual information [6]. Near-duplicated videos on the Web are efficiently exploited in [41, 5]. Complementary information, *e.g*. web images [54], user search behavior [55], or knowledge engine [5], is also investigated to help annotate Web videos. Yang *et al*. formulate face labeling in broadcasting news video as a MIL problem and propose the exclusive density method as the solution [52]. Different from the previous methods, we propose a new video annotation method that formulates the Web video annotation as a MIL problem and solve it with a deep network framework.

MIL deep networks. MIL is first formulated by Dietterich *et al*. [7]. It then becomes a widely adopted paradigm as many tasks can be cast as MIL problems. In earlier work, MIL is combined with SVM [4], traditional back-propagation (BP) network [35], or the AdaBoost method [13]. Recently, MIL deep networks become increasingly popular for weakly supervised tasks such as image classification [31] and object detection [40]. However, few studies investigate MIL deep networks for video annotation.

Max pooling has been widely adopted in existing MIL deep networks to fuse instance-level outputs by applying a maximum function [30, 31, 32]. Prior to the boom of deep learning, several kinds of alternative pooling functions had been presented, *e.g*. the generalized means, noisy-OR [57], log-sum-exponention (LSE) [35], and the integrated segmentation and recognition (ISR) model [21]. These functions are then adopted in MIL deep networks as pooling strategies [9, 33]. In [23], a comprehensive study is provided to evaluate several pooling strategies including noisy-OR, LSE, ISR, and adaptive noisy-AND. In all of these efforts, pooling functions are predefined before training rather than learnable.

Pooling functions in Deep Networks. Pooling functions also play important rules in deep networks. Mean, max, average, and stochastic pooling functions are the most well-known and widely used ones [1, 2, 56]. Moreover, Gulcehre *et al*. propose the Learned-Norm Pooling that uses the generalized mean, also known as the L_p norm, to learn the order p in deep feedforward and recurrent neural networks [11]. They provide interesting interpretations of the L_p norm by relating it to the maxout pooling function [10] and showing a geometrical insight. Generalized pooling strategies proposed by Lee *et al*. bring learning and "responsiveness" in the pooling operation [26] and use a tree-structured fusion of learned pooling operations. Malinowski and Fritz present a learnable spatial pooling function for visual recognition, where the function is fixed to

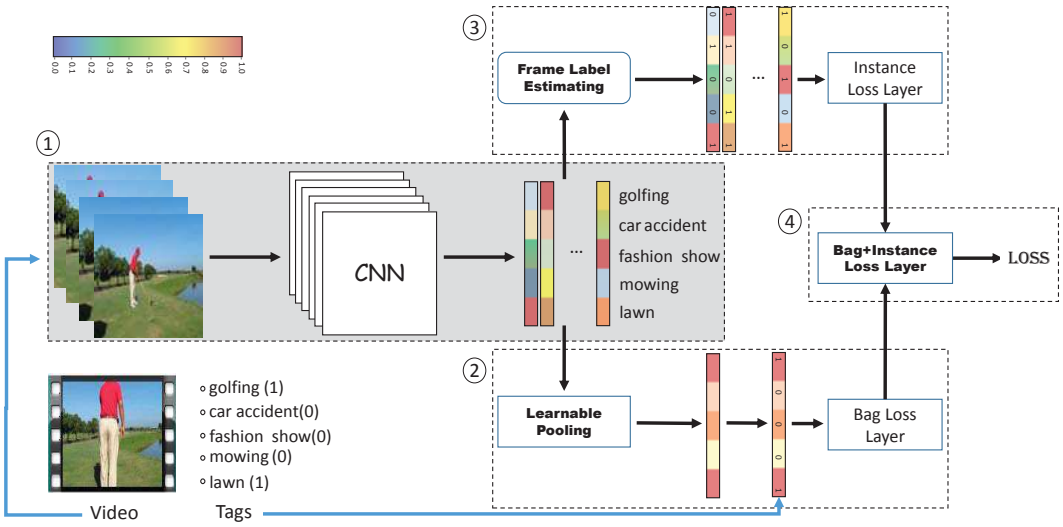

Figure 1. Illustration of the deep network used for frame-based video annotation. The entire network is used for training, whereas only the first two parts are used for online prediction. The colorful columns shown in Parts 1, 2, and 3 denote the predicted relevance values corresponding to different tags at frame level or video level, respectively. Color encodes the relevance value. Given an input video, the network processes all the frames separately and outputs predictions of each frame as in Part 1. Then those frame predictions are fed into Part 2 and Part 3 to derive instance-level and bag-level losses, respectively. Finally the two losses are combined to output the LOSS in Part 4. Please refer to Section 3.1 for more details.

average or max pooling but the pooling regions are adaptive [28]. Spatial pyramid pooling, which organizes pooling of different sizes into a pyramid-like hierarchical architecture, is proposed for images of varying size [12]. Jia *et al*. introduce the receptive fields learning in the pooling operation for image classification [15]. Different from the existing pooling functions, our proposed learnable pooling is dedicated for the MIL deep network that inherently has a semantic gap between instance-level learning and bag-level supervision. To the best of our knowledge, it is the first work investigating how to incorporate adaptive pooling function into the MIL deep network for video annotation.

3. The Proposed MIL Deep Network

In this section, we first present the overview of our proposed MIL deep network. Then we introduce the learnable pooling with bag+instance loss function and show how they enable end-to-end training of our MIL deep network.

3.1. Network Overview

Fig. 1 illustrates the overview of the proposed MIL deep network for Web video annotation. It consists of four parts as numbered in the figure. The first part contains a CNN that processes each input frame independently and outputs predicted relevance values (denoted by colors in columns) of tags (denoted as the colored columns) of each frame. The second part collects the outputs of all the frames of the same video and fuses them by the learnable pooling layer, resulting in video-level relevance values that are then compared with those of the ground-truth tags of the video. Meanwhile, the third part uses frame-level predictions to calculate the instance-level loss. Finally, the fourth part jointly minimizes the bag-level and instance-level losses.

Note that the network shown in Fig. 1 is used for training. In the online prediction, only the first two parts are needed. We use the trained CNN to process each frame inside an input video and achieve the frame-level predictions. Then these frame-level predictions are fused by the learned pooling function to generate the predicted tags. The frame-level predictions further enable us to localize the most relevant frames to a specific tag in videos.

In the following sections, we will introduce the learnable pooling function and the joint loss function.

3.2. Learnable Pooling

Formulation. We formulate the Web video annotation as a MIL problem based on the assumption that tags (classes) that can be inferred from individual frames (instances) are labeled to a video (bag of instances). In other words, a tag is associated with a video as long as it is identifiable from at least one frame inside the video. This is also a basic assumption of MIL.

Mathmatically, let $\mathbf{V}_i$ denote a bag of instances and $V_{ij} \in \mathbf{V}_i, j = 1, \ldots, N_i$, denote j-th instance where N_i is the number of instances of the bag. Let $c \in \{1, \ldots, C\}$ denote a specific class where C is the number of possible classes. Assuming the classes are independent of each other to simplify the discussion, the "ground-truth" of instance V_{ij} for class c is denoted by $y_{ij}^c \in \{0, 1\}$ and the ground-truth of the entire bag $\mathbf{V}_i$ for class c is denoted by $y_i^c \in \{0, 1\}$. We then have

$$y_i^c = 1 - \prod_{j=1}^{N_i}(1 - y_{ij}^c), \quad (1)$$

as a result of the MIL assumption. Note that y_i^c is available in the training data. Once $y_i^c = 0$, we have $y_{ij}^c = 0$ for sure; but if $y_i^c = 1$, the "ground-truth" of y_{ij}^c is actually not available in the training data. Thus, this is a weakly supervised problem.

We aim to learn a classifier for the instances, *e.g*. a CNN for static frames, and to predict the likelihood of V_{ij} relevant to class c, denoted by $q_{ij}^c \in [0, 1]$ hereafter. However, as long as the instance-level labels y_{ij}^c are not available, we have to rely on the bag-level label y_i^c. We then define a fusing/pooling function Φ_c to predict the likelihood of $\mathbf{V}_i$ being relevant to c as $p_i^c = \Phi_c(q_{i1}^c, q_{i2}^c, \ldots, q_{iN_i}^c)$.

Instantiation. The pooling function Φ_c shall mimic the MIL basic assumption Eq. (1). Indeed, replacing y_{ij}^c by q_{ij}^c in Eq. (1) has been used as a pooling function and termed as noisy-OR [9, 57], which is however mathematically intractable. We present a flexible yet simple pooling function based on the generalized means [11], which can adapt itself to different classes during training at a low computational cost,

$$p_i^c = \left(\frac{1}{N_i} \sum_{j=1}^{N_i} (q_{ij}^c)^{r_c^{(t)}} \right)^{\frac{1}{r_c^{(t)}}}, \quad (2)$$

where the exponent $r_c^{(t)}$ (also denoted by r for short hereafter) is the learnable parameter. Note the subscript c stands for different classes and the superscript t stands for different training stages. On the one hand, the corresponding r value can be small if a class is detectable in many instances; otherwise the corresponding r value should be large. On the other hand, the pooling strategy should be adaptive to the evolving network. At earlier stages of training, the network has little discriminative ability and its output is nearly random. In this case, a smaller r value *e.g*. $r = 1$ (average pooling) is suitable to suppress random errors. As the training goes on, the network is more and more tuned to the given samples and thus the predictions become increasingly reliable. A larger r value is then necessary to give more confidence to more relevant (larger q_{ij}^c) instances. Finally, if the network is perfect in the sense that $q_{ij}^c = y_{ij}^c$, then $r = \infty$ (max pooling) would be used so that Eq. (2) is equivalent to Eq. (1).

We would like to point out that the generalized means have been studied as pooling functions for MIL as well as for other deep networks. However, to the best of our knowledge, we are the first to propose dynamically adjusting the parameters of pooling to suit for the evolving network for video annotation.

3.3. New Bag+Instance Loss Function

Formulation. We now turn to the learning of our proposed pooling function, specifically, the learning of $r_c^{(t)}$ in Eq. (2). Please note that this parameter is learned for *training*, albeit useful for prediction, whereas almost all the other parameters in deep networks are learned for *prediction*. This difference encourages us to develop a new loss function to simultaneously learn the pooling function as well as the instance-level classifier.

As in classic MIL problems, the bag-level supervision should be taken into account in the loss function. For example, we can adopt the well known cross entropy as the bag-level loss function,

$$\ell_i^c = -y_i^c \times \log(p_i^c) - (1 - y_i^c) \times \log(1 - p_i^c). \quad (3)$$

Moreover, as the pooling function is designed to collect instance-level predictions to achieve bag-level prediction, the optimal pooling parameter should be dependent on the instance-level performance, which is evolving during the training process. The instance-level loss can be estimated by

$$\mathfrak{R}_{ij}^c = -u_{ij}^c \times \log(q_{ij}^c) - (1 - u_{ij}^c) \times \log(1 - q_{ij}^c), \quad (4)$$

where $u_{ij}^c = \mathbf{1}(q_{ij}^c \geq 0.5)$ with $\mathbf{1}(\cdot)$ being an indicator function. Indeed, the instance-level loss is a measure of the uncertainty of q_{ij}^c, which also represents the discriminative ability of the instance-level classifier. We then propose to minimize the difference between bag-level and instance-level losses, *i.e*.

$$d_i^c = \|\ell_i^c - \frac{1}{N_i} \sum_{j=1}^{N_i} \mathfrak{R}_{ij}^c\|. \quad (5)$$

Minimizing Eq. (5) requires the pooling, as the bridge between instance and bag, to be suitable for the current status of the instance-level classifier. The pooling layer effectively transfers the instance-level discriminative ability into the bag-level classification ability. Within the context of error back propagation in network training, the bag-level loss is the most faithfully back propagated to the instance-level classifier through the pooling layer. In this sense, we regard the learned pooling as optimal for *training*.

In summary, we propose the following loss function to jointly minimize the bag-level loss and the difference between bag-level and instance-level losses,

$$L_i^c = \ell_i^c + \lambda (d_i^c)^2 = \ell_i^c + \lambda \left(\ell_i^c - \frac{1}{N_i} \sum_{j=1}^{N_i} \mathfrak{R}_{ij}^c \right)^2, \quad (6)$$

where λ is the Lagrangian multiplier ($\lambda = 1$ in all experiments). We use the square of difference Eq. (5) for methematical simplicity.

Calculation. We follow the stochastic gradient descent (SGD) approach to minimize the loss function for training the deep networks. In a mini-batch, the losses of different videos and different tags are calculated individually and later summed up. Here losses of positive and negative videos are calculated differently. For positive videos where $y_i^c = 1$, we adopt the loss function defined in Eq. (6). But for negative videos where $y_i^c = 0$, we have $y_{ij}^c = 0$ surely for any frame j, and thus directly use the instance-level loss,

$$L_i^c = \frac{1}{N_i} \sum_{j=1}^{N_i} \mathfrak{R}_{ij}^c, \tag{7}$$

where $\mathfrak{R}_{ij}^c$ is calculated by Eq. (4) with u_{ij}^c set to 0. In other words, the pooling from frames to video is not applicable to negative videos, which then contribute none to the learning of the pooling.

4. Experiments

4.1. Settings

Datasets. We evaluate the performance of our proposed method on the large scale video dataset FCVID [19]. FCVID contains 91,223 videos crawled from YouTube, being one of the largest datasets of real-world Web videos. Videos in FCVID are manually labeled with 239 predefined categories, covering a wide range of topics like social events (*e.g*. tailgate party), procedural events (*e.g*. making cake), objects (*e.g*. panda), scenes (*e.g*. beach), and so on. Note that each video can be labeled with multiple annotations, and the manual labeling process had been carefully designed so that a fairly complete label set is achieved for each video. In addition to raw videos and annotations, the FCVID dataset provides several pre-computed features including CNN-based, SIFT, dense trajectories, and audio features, as well as other metadata. Since our research focuses on frame-based video annotation, we utilize only the video frames and annotations in the FCVID dataset. The train/test split follows that in [19], with 45,611 videos used for training and 45,612 videos for test.

We also conduct experiments on another real-world web video dataset CCV. CCV was collected with care to ensure relevance to consumers' interest without post-editing. The dataset contains 9,317 unconstrained youtube videos over 20 semantic categories and each video is annotated manually. CCV dataset also provides extra audio and visual features, however, we only use the raw videos and annotations in our framework. The train/test split is kept the same with the standard partition in CCV dataset.

In addition, we perform tests on the well-known UCF101 dataset [42], which contains 13,320 videos with 101 action categories, to demonstrate how our proposed method handle videos with high temporal correlation. We choose UCF101 as it is one of the largest human action datasets in terms of variety of categories, number of samples, and number of baselines. Note that the categories in UCF101 are restricted to human actions and not as comprehensive as those in FCVID or CCV. Thus, we train a different network for each dataset. The train/test split follows that in [42].

Training Data. For each training video, we use only a random subset of its frames to reduce the training workload. The sample size is proportional to the number of frames of each video. In our experiments, the samples vary from 5 to 41 frames as the video durations are between several seconds to several minutes. We organize the sampled frames into mini-batches, and carefully ensure that all the sampled frames of the same video are included in the same mini-batch. Moreover, each mini-batch contains video frames of the same sample size, so that the pooling function is relatively easy to implement. Then, the effective numbers of videos in mini-batches are different since the mini-batch size is fixed to a multiple of sample size not exceeding 60. We perform random mirroring and cropping on the video frames before CNN to help avoid over-fitting. The data are shuffled only once before training and all the experiments use the same shuffled data.

Training Process. We implement the proposed learnable pooling and bag+instance loss function in the framework of Caffe [16] and modify the well-known VGG-19 model [38] as the basic CNN to process video frames. We change the last full-connection layer of VGG-19 to 239-dim for FCVID, 101-dim for UCF101 and 20-dim for CCV. We also replace the softmax by sigmoid for FCVID because in FCVID the annotations are not mutually exclusive. Except for the modified full-connection layers that are randomly initialized, the other layers of VGG-19 are initialized with those parameters pretrained on ImageNet [38]. The pooling parameters are all initialized with 1 (average pooling). As for the parameters for training, the learning rate is fixed to 10^{-6}, the momentum is 0.9, and the weight decay parameter is 0.0005 (but 0 for pooling parameters).

Evaluation. For each test video, we also select only a subset of its frames and use the trained CNN to predict the frame-level annotations. Then we test two approaches to predict the video annotations, using the learned pooling parameters or simply using max pooling. Finally, we report the mean average precision (mAP) achieved on all the test videos.

4.2. Performance Evaluation

FCVID. In order to verify the efficiency of our proposed learnable pooling, we perform several comparative experiments on FCVID with regard to the network with learnable pooling (LearnP) and the network with fixed average pooling (AveP) or max pooling (MaxP). We also evaluate the performance of the new bag+instance loss (BIL) in comparison with that of the traditional bag-level loss (BL), where

Method		mAP
SVM-MKL [22]		75.2%
M-DBM [43]		74.4%
rDNN-F [19]		75.4%
DASD [17]		72.8%
rDNN-C [19]		74.4%
rDNN [19]		76.0%
OSF Network [50]		76.5%
MIL (MaxP in test)	MaxP+BL	66.8%
	AveP+BL	69.6%
	LearnP+BL	76.1%
	AveP+BIL	75.3%
	LearnP+BIL	76.4%
LearnP+BIL (LearnP in test)		**78.5%**

Table 1. Performances of different methods on FCVID.

Method		mAP
Nagel *et al*. [29]		71.7%
Jiang [19]		73.5%
Wu *et al*. [51](RGB-stream)		75.0%
Jiang *et al*. [18](RGB-stream)		77.9%
Chang *et al*. [3]		78.3%
Ours	MaxP+BL	72.8%
	AveP+BL	74.3%
	LearnP+BL	79.7%
	AveP+BIL	79.3%
	LearnP+BIL	**80.2%**

Table 2. Performances of different methods on CCV.

BL is equivalent to setting $\lambda = 0$ in Eq. (6).

Moreover, the annotation accuracy of our proposed method is compared with the state-of-the-arts on the FCVID dataset, including SVM plus multiple kernel learning (SVM-MKL) [22], multimodal deep Boltzmann machines (M-DBM) [43], domain adaptive semantic diffusion (DASD) [17], a series of regularized deep network methods (rDNN, rDNN-F, rDNN-C) [19], and object-scene semantic fusion (OSF) network [50]. Please note that our method uses only individual frames to perform video annotation. But the other state-of-the-art results are achieved with multimodal features, including frame-based features, motion trajectory features, and audio features, and/or much more complex networks that consider feature relations, class relations, or both.

Table 1 summarizes the mAP results of different methods on FCVID. It demonstrates that our proposed learnable pooling with the new bag+instance loss achieves the best result. Though our method is frame-based, it outperforms much more complex networks that use temporal correlation and/or multimodal information. Therefore, the results demonstrate the high potential of our proposed method in Web video annotation.

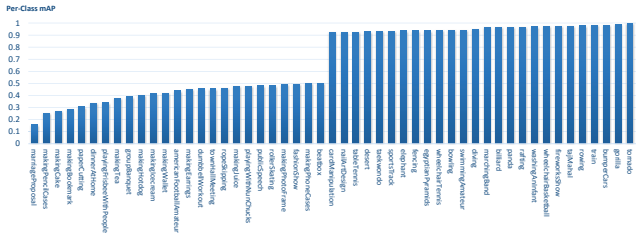

Figure 2. The highest 10% and lowest 10% per-tag mAP results together with the corresponding tags in FCVID.

The canonical setting of MIL is to learn an instance-level classifier given only bag-level supervision. For that purpose, we perform experiments that adopt different training strategies[1] but always use max pooling in the test, as shown in Table 1. Since max pooling is fixed, the final mAP is solely dependent on the instance-level classification accuracy, but which is hard to measure due to lack of ground-truth at frame level. We can observe that both the learnable pooling and the new bag+instance loss (BIL) lead to better training of the instance-level classifier. When using the traditional bag-level loss (BL), learnable pooling provides more than 6 (9) percents gain in mAP over average (max) pooling. When using the proposed BIL, the gain provided by learnable pooling is lower but still around 1 percent. The gain in mAP provided by BIL is more than 5 percents for average pooling and 0.3 percent for learnable pooling. It is worth noting that even 0.5 percent improvement on mAP could be notable according to previous state-of-the-art work [50, 19]. These results verify the effectiveness of our proposed learnable pooling and bag+instance loss function for Web video annotation in the context of MIL.

We further calculate the mAP results separately for each tag and plot the highest 10% and lowest 10% mAP results together with the corresponding tags in Fig. 2. It can be observed that the tags with higher mAP are often concepts of scene (*e.g*. tornado, fireworks show), object (*e.g*. gorilla, train), or action with specific scene or object (*e.g*. rowing, diving). Tags with lower mAP are often actions, especially of subtle motions (*e.g*. making pencil cases, making bookmark) or events that with diverse behaviors (*e.g*. marriage proposal, dinner at home). Note that our method is built upon frame-level CNN without exploration of temporal correlation in videos, and thus has difficulty in distinguishing action-related concepts.

CCV. Table 2 presents the mAP of different methods in comparison on CCV. It can be observed that our method (learnable pooling + bag+instance loss + using the learned pooling in the test) significantly boosts the performance on the baselines and achieves the best results compared to other work when only spatial information is used as input. Espe-

[1]We did not test the combination of max pooling and bag+instance loss because if using max pooling, the bag-level loss is determined by the most relevant instance and thus the second term of Eq. (6) is not meaningful.

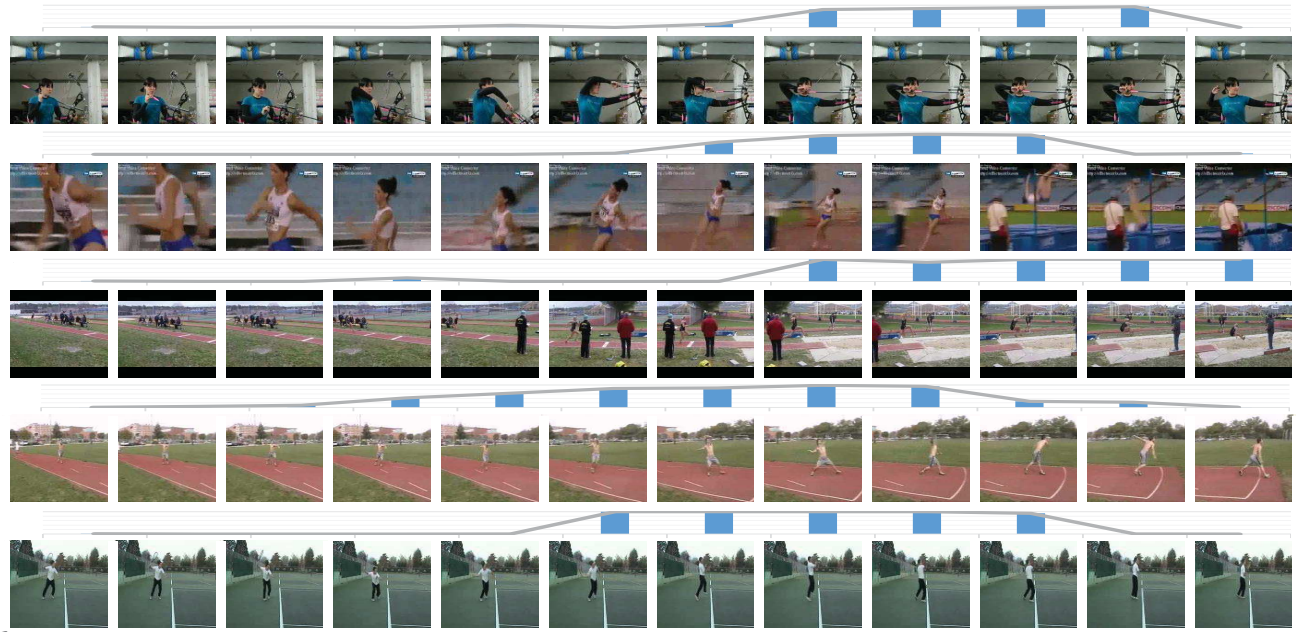

Figure 3. Example results of frame-level predictions of relevance values shown above the video frames. Each row shows frames selected from one video. The tags from top to bottom are: Archery, High Jump, Long Jump, Javelin Throw, and Tennis Swing.

cially, our method outperforms the complex framework in [3] that also focuses on pooling, which proposes the semantic pooling by defining semantic salience for video event analysis. Apparently, our method is more efficient in terms of both performance and complexity.

UCF101. Table 3 presents the accuracy of different methods on UCF101. It shows that our proposed method significantly enhances the accuracy of annotation on actions such as playing musical instruments (PM in Table 3) as these actions are highly related to surrounding objects that are identifiable from still frames. We can observe that our scheme with simple CNN models outperforms the methods that involve the spatial-temporal information and thus introduce more complex models [8, 27]. Compared with the state-of-the-art action recognition algorithms [36, 46], our scheme achieves comparable accuracy, 83.4% in average, based on the same CNN features. However, our scheme is much more efficient with a quite simple CNN structure while the others are using complicated LSTM with multi-modal inputs [36, 8, 27] or much deeper CNN structures [46].

Method \ Type	HO	BM	HH	PM	SP
A.K *et al.* [20]	55%	57%	68%	65%	79%
Our Method	71%	73%	84%	96%	87%
J.D *et al.* [8]	82.9%				
Li *et al.* [27]	82.1%				
S.S *et al.* [36]	84.5%				
*X.W *et al.* [49]	80.2%				
+L.W *et al.* [46]	84.5%				

Table 3. Performances of different methods on UCF101. The action categories are divided into five types [42]: Human-Object Interaction (HO), Body-Motion Only (BM), Human-Human Interaction (HH), Playing Musical Instruments (PM), and Sports (SP). Some references did not report per-type mAP. * refers to the results that are reported in the settings where only spatial information is used as input. * and + indicate only RGB stream is used.

4.3. Analyses

We first analyze the performance of the proposed learnable pooling along with the joint loss function in training process. Fig. 4 plots the learning curves of different training strategies for MIL. It shows that, at earlier stages of training, the max pooling turns out to be inefficient while the average pooling performs better as it suppresses random errors of network. Our proposed learnable pooling achieves the highest speed of training. Using BIL also provides higher speed of training, and the speedup is quite significant when combining with average pooling. This result demonstrates the feasibility of the learnable pooling in training MIL networks as well as the effectiveness of our proposed bag+instance loss.

We then discuss the learned pooling parameters r for different tags during the training, some of which are plotted in Fig. 4. This figure shows that the parameter r is increasing monotonically as training continues regardless of the types of losses and tags. It exactly matches our conjecture that as the training goes on, the discriminative ability of network is

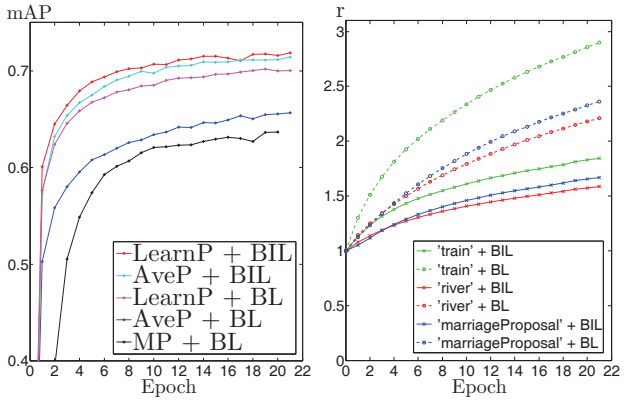

Figure 4. Left: The learning curves of different MIL methods. Right: The learned pooling parameters for several different tags (train, river, marriage proposal) during training.

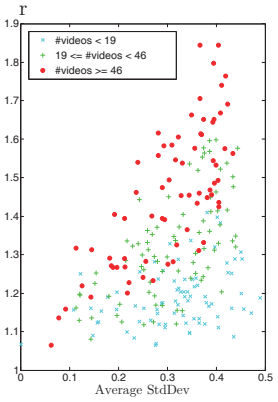

Figure 5. The scatter plot of learned pooling parameters and the average standard deviations of predicted relevances.

increasing and r should be increased to weight more confidence on more relevant instances. Another observation is that the learned r is smaller when using BIL than that when using BL, because the second term of Eq. (6) tends to punish larger r that widens the gap between bag-level and instance-level losses.

As different tags have different learned pooling parameters as shown in Fig. 4, we further analyze the relation between the learned pooling parameter and the video characteristics of each tag. Quantitatively, we calculate the average standard deviation of the predicted relevances for each tag c, namely $\nu^c = \frac{1}{N_c}\sum_{i=1}^{N_c}\nu_i^c$, where N_c is the number of videos annotated with tag c in the training set, and

$$\nu_i^c = \sqrt{\frac{1}{N_i}\sum_{j=1}^{N_i}(q_{ij}^c - \overline{q_i^c})^2}, \text{ where } \overline{q_i^c} = \frac{1}{N_i}\sum_{j=1}^{N_i} q_{ij}^c. \quad (8)$$

The scatter plot of ν^c and r^c is shown in Fig. 5. Considering the average value ν^c is more reliable when N_c is large, the plot distinguishes tags with different N_c's. From Fig. 5, a clear positive correlation between ν^c and r^c can be observed especially when N_c is large. Such result reveals that, if a tag's relevances vary more significantly among different frames of a video, then the corresponding r should be larger so as to put more attention to the more relevant frames, and vice versa. It also matches our intuition and demonstrates the effectiveness of learning different pooling parameters for different classes.

As we formulate the Web video annotation as a MIL problem, we achieve a network which provides not only video-level but also frame-level predictions. The related frame-level predictions are visualized in Fig. 3. The predicted relevance values are consistent with human inference that only a portion of frames have high relevance to a given concept of the video and the frames with high relevance values are statistically typical for representing the semantic concept of the video. For instance, most videos with concept "High Jump" contain some frames in which people run up with his tilted body or cushion object exists around, while videos of "Long Jump" usually contain frames with sand pools. These frames are representative enough for the concept and thus assigned with high relevance values, as shown in Fig. 3. The rest frames, like people running up in an ordinary pose, are given low relevance values as they are ambiguous in terms of concepts. In this way, our proposed video annotation method is inherently able to localize the relevant frames to a concept in a video.

5. Conclusions

In this paper, we present a MIL deep network for tagging the weakly annotated Web videos. Rather than using a predefined pooling function, we introduce a learnable pooling function that is dynamically adjusted to adapt to different classes in the training process. Moreover, we propose a new bag+instance loss function in conjunction with our learnable pooling function to simultaneously learn the instance-level network and the optimal pooling in a much efficient way. Experimental results and further analyses show that the proposed learnable pooling and new loss function both help to improve the final performance in video annotation, leading to the best known results on FCVID. The results on UCF101 indicate the proposed method has some limitations on highly dynamic action-related videos that usually require temporal-aware methods for better annotation, such as "Body-Motion" and "Human-Object Interaction". Nevertheless, our method is still promising on those action videos that are more scene sensitive or contain some representative body poses, *e.g.* "Sports" and "Human-Human Interaction". Our proposed method can also be extended to enable training of deep network models to cope with more kinds of MIL problems, which will be studied in the future.

References

[1] Y.-L. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the locals: multi-way local pooling for image recogni-

tion. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2651–2658. IEEE, 2011.

[2] Y.-L. Boureau, J. Ponce, and Y. LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118, 2010.

[3] X. Chang, Y.-L. Yu, Y. Yang, and E. P. Xing. Semantic pooling for complex event analysis in untrimmed videos. *IEEE transactions on pattern analysis and machine intelligence*, 39(8):1617–1632, 2017.

[4] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *JMLR*, 5:913–939, 2004.

[5] Z. Chen, J. Cao, Y. Song, Y. Zhang, and J. Li. Web video categorization based on wikipedia categories and content-duplicated open resources. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1107–1110. ACM, 2010.

[6] Z. Chen, J. Cao, T. Xia, Y. Song, Y. Zhang, and J. Li. Web video retagging. *Multimedia Tools and Applications*, 55(1):53–82, 2011.

[7] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1):31–71, 1997.

[8] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2625–2634, 2015.

[9] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, et al. From captions to visual concepts and back. In *CVPR*, pages 1473–1482, 2015.

[10] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio. Maxout networks. In *ICML*, volume 28, pages 1319–1327, 2013.

[11] C. Gulcehre, K. Cho, R. Pascanu, and Y. Bengio. Learned-norm pooling for deep feedforward and recurrent neural networks. In *ECML-PKDD*, pages 530–546. Springer, 2014.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *European Conference on Computer Vision*, pages 346–361. Springer, 2014.

[13] R. Hong, M. Wang, Y. Gao, D. Tao, X. Li, and X. Wu. Image annotation by multiple-instance learning with discriminative feature mapping and selection. *IEEE Trans. Cybernetics*, 44(5):669–680, 2014.

[14] S. Ji, W. Xu, M. Yang, and K. Yu. 3D convolutional neural networks for human action recognition. *IEEE Trans. PAMI*, 35(1):221–231, 2013.

[15] Y. Jia, C. Huang, and T. Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3370–3377. IEEE, 2012.

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. arXiv:1408.5093, 2014.

[17] Y.-G. Jiang, Q. Dai, J. Wang, C.-W. Ngo, X. Xue, and S.-F. Chang. Fast semantic diffusion for large-scale context-based image and video annotation. *IEEE TIP*, 21(6):3080–3091, 2012.

[18] Y.-G. Jiang, Z. Wu, J. Tang, Z. Li, X. Xue, and S.-F. Chang. Modeling multimodal clues in a hybrid deep learning framework for video classification. *arXiv preprint arXiv:1706.04508*, 2017.

[19] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang. Exploiting feature and class relationships in video categorization with regularized deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[20] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, pages 1725–1732, 2014.

[21] J. D. Keeler, D. E. Rumelhart, and W.-K. Leow. *Integrated Segmentation and Recognition of Hand-Printed Numerals.* Microelectronics and Computer Technology Corporation, 1991.

[22] M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien. Lp-norm multiple kernel learning. *JMLR*, 12:953–997, 2011.

[23] O. Z. Kraus, L. J. Ba, and B. Frey. Classifying and segmenting microscopy images using convolutional multiple instance learning. *arXiv:1511.05286*, 2015.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[25] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, pages 1–8. IEEE, 2008.

[26] C.-Y. Lee, P. W. Gallagher, and Z. Tu. Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree. In *International conference on artificial intelligence and statistics*, 2016.

[27] Z. Li, E. Gavves, M. Jain, and C. G. Snoek. Videolstm convolves, attends and flows for action recognition. *arXiv preprint arXiv:1607.01794*, 2016.

[28] M. Malinowski and M. Fritz. Learning smooth pooling regions for visual recognition. In *BMVC*, pages 1–11. BMVA Press, 2013.

[29] M. Nagel, T. Mensink, C. G. Snoek, et al. Event fisher vectors: Robust encoding visual diversity of visual streams. In *BMVC*, volume 2, page 6, 2015.

[30] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free?–Weakly-supervised learning with convolutional neural networks. In *CVPR*, pages 685–694, 2015.

[31] G. Papandreou, I. Kokkinos, and P.-A. Savalle. Untangling local and global deformations in deep convolutional networks for image classification and sliding window detection. *arXiv:1412.0296*, 2014.

[32] D. Pathak, E. Shelhamer, J. Long, and T. Darrell. Fully convolutional multi-class multiple instance learning. *arXiv:1412.7144*, 2014.

[33] P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, pages 1713–1721, 2015.

[34] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, and H.-J. Zhang. Correlative multi-label video annotation. In *Proceedings of the 15th international conference on Multimedia*, pages 17–26. ACM, 2007.

[35] J. Ramon and L. De Raedt. Multi instance neural networks. In *ICML*, pages 53–60, 2000.

[36] S. Sharma, R. Kiros, and R. Salakhutdinov. Action recognition using visual attention. *arXiv preprint arXiv:1511.04119*, 2015.

[37] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014.

[38] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.

[39] C. G. Snoek, M. Worring, and A. W. Smeulders. Early versus late fusion in semantic video analysis. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 399–402. ACM, 2005.

[40] H. O. Song, Y. J. Lee, S. Jegelka, and T. Darrell. Weakly-supervised discovery of visual pattern configurations. In *NIPS*, pages 1637–1645, 2014.

[41] J. Song, Y. Yang, Z. Huang, H. T. Shen, and R. Hong. Multiple feature hashing for real-time large scale near-duplicate video retrieval. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 423–432. ACM, 2011.

[42] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human action classes from videos in the wild. Technical Report CRCV-TR-12-01, University of Central Florida, 2012.

[43] N. Srivastava and R. R. Salakhutdinov. Multimodal learning with deep Boltzmann machines. In *NIPS*, pages 2222–2230, 2012.

[44] S.-W. Sun, Y.-C. F. Wang, Y.-L. Hung, C.-L. Chang, K.-C. Chen, S.-S. Cheng, H.-M. Wang, and H.-Y. M. Liao. Automatic annotation of web videos. In *Multimedia and Expo (ICME), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.

[45] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, pages 3551–3558, 2013.

[46] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision*, pages 20–36. Springer, 2016.

[47] M. Wang, R. Hong, G. Li, Z.-J. Zha, S. Yan, and T.-S. Chua. Event driven web video summarization by tag localization and key-shot identification. *IEEE TMM*, 14(4):975–985, 2012.

[48] M. Wang, X.-S. Hua, J. Tang, and R. Hong. Beyond distance measurement: constructing neighborhood similarity for video annotation. *Multimedia, IEEE Transactions on*, 11(3):465–476, 2009.

[49] X. Wang, A. Farhadi, and A. Gupta. Actions ˜ transformations. *arXiv preprint arXiv:1512.00795*, 2015.

[50] Z. Wu, Y. Fu, Y.-G. Jiang, and L. Sigal. Harnessing object and scene semantics for large-scale video understanding. In *CVPR*, pages 3112–3121, 2016.

[51] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 461–470. ACM, 2015.

[52] J. Yang, R. Yan, and A. G. Hauptmann. Multiple instance learning for labeling faces in broadcasting news video. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 31–40. ACM, 2005.

[53] Y. Yang, Z. Ma, Z. Xu, S. Yan, and A. G. Hauptmann. How related exemplars help complex event detection in web videos? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2104–2111, 2013.

[54] Y. Yang, Z.-J. Zha, Y. Gao, X. Zhu, and T.-S. Chua. Exploiting web images for semantic video indexing via robust sample-specific loss. *IEEE Transactions on Multimedia*, 16(6):1677–1689, 2014.

[55] T. Yao, T. Mei, C.-W. Ngo, and S. Li. Annotation for free: Video tagging by mining user search behavior. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 977–986. ACM, 2013.

[56] M. D. Zeiler and R. Fergus. Stochastic pooling for regularization of deep convolutional neural networks. *arXiv preprint arXiv:1301.3557*, 2013.

[57] C. Zhang, J. C. Platt, and P. A. Viola. Multiple instance boosting for object detection. In *NIPS*, pages 1417–1424, 2005.