DETECTING RARE EVENTS USING KULLBACK-LEIBLER DIVERGENCE

Jingxin Xu, Simon Denman, Clinton Fookes, Sridha Sridharan

Image and Video Research Laboratory, SAIVT Research Group, Queensland University of Technology, Australia
{j15.xu, s.denman, c.fookes, s.sridharan}@qut.edu.au

ABSTRACT

One main challenge in developing a system for visual surveillance event detection is the annotation of target events in the training data. By making use of the assumption that events with security interest are often rare compared to regular behaviours, this paper presents a novel approach by using Kullback-Leibler (KL) divergence for rare event detection in a weakly supervised learning setting, where only clip-level annotation is available. It will be shown that this approach outperforms state-of-the-art methods on a popular real-world dataset, while preserving real time performance.

Index Terms— event detection, video surveillance

1. INTRODUCTION

In recent years, a large number of approaches have been proposed for automatic event detection in video surveillance systems [1–9]. Real world surveillance scenes are often crowded, making approaches based on object tracking [10] unsuitable, and leading to the development of techniques that extract features directly from the image to recognize actions and events [1, 4, 6, 7, 11]. A popular feature representation is an optical flow based descriptor and its extensions [3, 6], which is termed "discrete optical flow" in [13]. However, as optical flow only captures motion between successive frames, the loss of motion characteristics across more than two frames results in some events not being separable. It has been shown in [8] that by using point trajectories as the feature, improvement can be achieved as the trajectories inherently preserve the order of the motion. Regarding the machine learning paradigm, most publications in this field use one-class unsupervised learning [1, 6, 11] based on outlier detection. This strategy is based on the assumption that suspicious and emergency events usually occur at low frequencies. However, such a detection system is limited as it cannot identify what the detected event is. To be able to identify the events, supervised learning approaches have to be used. However, the annotation requirements (individual bounding boxes for all events) make such an approach highly impractical. An alternative is frame level annotation, however this is ambiguous, as there are usually a lot of other irrelevant events in the scene. Recently, [3] introduced the concept of "weakly supervised learning", which is a special type of supervised learning in activity perception where only binary labels at the clip level are used to indicate if an event of interest is present or not, but does not identify where and when the event happens. In this paper, based on the concept of "weakly supervised learning", we propose a method that applies Kullback-Leibler (KL) divergence [12] to detect the video clips that contain the event of interest with binary labels at the clip level. Compared to other weakly supervised learning methods [3, 13], the classifier in the proposed approach has a much lower complexity, which results in computational efficiency and being robust to parameter initializations. The additional assumption that the target events are rare matches many real world surveillance applications that focus on security.

2. ALGORITHM

2.1. Feature Representation

We consider two feature representations: the discrete optical flow approach of [6], and a trajectory representation similar to that proposed by [13]. Within the trajectory based approach, we consider MPEG motion vectors as in [13], particle video as used by [8], as well as the KLT (Kanade-Lucas-Tomasi) tracker [14].

To extract trajectory features, the video is divided into uniform clips, from which trajectories are extracted (either using MPEG motion vectors, particle video, or the KLT tracker). Trajectories with short durations are removed, and a set of point trajectories is created (See Figure 1). A trajectory, s, is described by a sequence of locations in time. Since the trajectories that are generated by different instances of the same events vary from each other, it is necessary to conduct a dimension reduction to capture the key characteristics that are shared by different instances of the same event. Motivated by [13, 15, 16], the low Fourier coefficients are used to describe the trajectories. The sequence is separated into a horizontal and a vertical series, which are denoted $X = [x_1, x_2, ..., x_N]$ and $Y = [y_1, y_2, ..., y_N]$. The Fourier Transform is taken sep-

Fig. 1. Trajectory Construction

arately on the two signals,

$$X_f = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} x_i exp(\frac{-j2\pi ft}{N}), \quad f = 0, 1, ..., N-1 \quad (1)$$

$$Y_f = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} y_i exp(\frac{-j2\pi ft}{N}). \quad f = 0, 1, ..., N-1. \quad (2)$$

The number of points N is equal to the number of frames in a video clip. For trajectories with fewer than N points, the last point is repeated to fill the trajectory, and thus ensure that the DFT is performed on a fixed length sequence. The Fourier coefficients of each trajectory are viewed as a sample, and K-means is applied to train a codebook. Finally, a video clip is represented by a histogram of codewords. To select the number of clusters for K-means, we select the elbow point of the curve formed by plotting number of clusters against the sum of square errors [17].

2.2. Event Detection using KL divergence

In this section, we present a novel approach for event detection using weakly supervised learning, meaning we only have binary labels at the clip level (as in [3, 13]) during training. As it is highly likely that the event of interest will be co-occurring with many others, the irrelavent events at the same clip are expected to result in the failure of a supervised learning method.

In real world deployment, events with security interest often occur in lower frequencies than regular behaviours, indicating the number of video clips labeled "1" (containing the event) is much smaller than the number of video clips labeled "0" (without the event). By making use of this assumption, we propose to apply Kullback-Leibler (KL) divergence [12] to efficiently and effectively separate the video clips into two classes based on whether the event of interest is present or not. The structure of the learning model is similar to the naive Bayes model, with the difference being that the log-likelihood used in naive Bayes is replaced with KL divergence. However, this replacement brings significant benefits in our application.

Let $\mathscr{V} = \{v_0, v_1, v_2, \cdots, v_{K-1}\}$ be the vocabulary with a cardinality of K. Let c denote the label of a video clip, and p represent the probability. Let a video clip be represented as a histogram of code words, $X = [x_0, x_1, \cdots, x_{K-1}]$, where x_i is the frequency of v_i in the present video clip. Suppose a video clip X is considered to be generated by sampling a distribution H, which is a probability distribution of the K codewords over $\mathscr{V}$. Let $G \subset \mathscr{V}$ be a subset of $\mathscr{V}$ which contains the codewords specified for the event of interest. The motion patterns for the background events are represented by codewords other than those in G. Let M be the cardinality of G. As G is a non-empty subset of $\mathscr{V}$, we have $K > G \geq 1$. To facilitate the analysis, we can order the code words in $\mathscr{V}$ so that the last M codewords are elements of G, as

$$v_{K-M}, v_{K-M-1}, \cdots, v_{K-1} \in G. \quad (3)$$

Let $H = [h_0, h_1, \cdots, h_{K-1}]$, where h_i is the probability of the codeword v_i. Clearly, $\sum_{i=0}^{K-1} h_i = 1$, and $h_i \geqq 0$.

Suppose the difference between class "1" and class "0" is only whether there is the presence of the event of interest or not. We partition the video clips generated by H into two classes. Those containing the codewords from G are labelled "1"; others are labelled "0".

Now let us consider the conditional probabilities under the known labels. For video clips labelled "1", all codewords are possible. Therefore,

$$p(v_i|c = 1) = h_i, \quad (4)$$

where $0 \leq i < K$. For video clips labelled "0", the occurrence of codewords from G is impossible, which indicates

$$p(v_i|c = 0) = \begin{cases} \frac{h_i}{S} & (i < K - M) \\ 0 & (K > i \geq K - M) \end{cases}, \quad (5)$$

where $S = \sum_{j=0}^{K-M+1} h_j$ is a constant to ensure $\sum_{i=0}^{K-1} p(v_i|c = 0) = 1$. Since S is the summation of a partial probability distribution, then $S < 1$. If $i < K - M$, we will have

$$\frac{p(v_i|c = 1)}{p(v_i|c = 0)} = \frac{h_i}{\frac{h_i}{S}} = S. \quad (6)$$

Eq. 6 shows that ratio of the probabilities of a codeword associated with background events conditioned on the two labels is a constant.

The goal of the learning process is to estimate $p(v_i|c = j)$, where $i \in \{x|x \in N, 0 \leqq x < K\}$ and $j \in \{0, 1\}$. If we partition the training samples into two groups by their labels, we can obtain the histogram of words for each class. Given sufficient training data, these histograms can be used to estimate $p(v_i|c = j)$ by adding a normilisation step,

$$p(v_i|c = j) \approx \frac{n_i^j}{N^j}, \quad (7)$$

where n_i^j denotes the frequencies of v_i when $c = j$; and N^j is the total number of words when $c = j$. From Eq. 6 and Eq. 7, we have

$$\begin{aligned}\frac{n_i^1}{n_i^0} &\approx \frac{p(v_i|c=1)\times N^1}{p(v_i|c=0)\times N^0}\\ &= S \times \frac{N^1}{N^0},\end{aligned} \tag{8}$$

under the condition of $i < K-M$ (this indicates $v_i \in G$). In the previous discussion, it was shown that $S < 1$. Suppose that the event of interest is a rare event, such that the number of video clips labelled "0" is much larger than the number of video clips labelled "1". Thus the number of codewords for the video clips labelled "0" is much larger than the number of codewords for video clips labelled "1", i.e. $N^1 < N^0$. Then $\frac{N^1}{N^0} < 1$. Based on the above discussion, we have the following expression:

$$\forall i,j \quad i \neq j; \quad v_i, v_j \notin G \Rightarrow \frac{n_i^1}{n_i^0} = \frac{n_j^1}{n_j^0} = r, \tag{9}$$

where $r < 1$ is a constant. If the training dataset is sufficiently large, then $n_i^0 \gg 1$ and $n_i^1 \gg 1$. We further have

$$\frac{1+n_i^1}{1+n_i^0} \approx \frac{n_i^1}{n_i^0} = r \quad v_i, v_j \notin G \quad \&\& \; i \neq j. \tag{10}$$

The term $\frac{1+n_i^1}{1+n_i^0}$ refers to the Laplace smoothing approximation in the training process. In the case where $n_i^j = 0$, $P(v_i|c = j) = 0$. The possibility of 0 for the probabilities will causes problems such as division by 0 in the following process. As such, additive smoothing (Laplace smoothing) is applied[18]

$$p(v_i|c=j) \approx \frac{1+n_i^j}{N^j + K}. \tag{11}$$

The addition of 1 in the numerator avoids the encounter of 0 probabilities, and the addition of K in the denominator ensures the marginal probabilities sum to one.

After training, there is a learned distribution of codewords for each class: P_0 for class "0"; and P_1 for class "1".

Given a video clip $X = [x_0, x_1, \cdots, x_{K-1}]$, we can normalise it into a distribution $Q = [q_0, q_1, q_2, \cdots, q_{K-1}]$, where $q_i = x_i / \sum_{j=0}^{K} x_j$. Intuitively, one simple criterion of recognition can be that if the distribution Q is closer to P_0, then the video clip is recognized as class "0"; otherwise, it is recognized as class "1".

The KL divergence is used to measure the similarity of two distributions. Let $D_{KL}(Q||P_0)$ be the KL divergence for Q and P_0; and let $D_{KL}(Q||P_1)$ be the KL divergence for Q and P_1. Then we have

$$D_{KL}(Q||P_c) = \sum_{i=0}^{K-1} q_i ln \frac{q_i}{p(v_i|c)}, \tag{12}$$

where $c \in \{0, 1\}$ is the class label. In binary classification applications using traditional supervised learning, the result $D_{KL}(Q||P_0) > D_{KL}(Q||P_1)$ indicates that the codeword distribution in the present video clip Q is closer to the codeword distribution of class "1". Thus we classify the video clip as class "1". Equivalently, we can write the criterion as $D_{KL}(Q||P_0) - D_{KL}(Q||P_1) > 0$ for the classification of labelled "1" video clips. In the following discussion, we show that this method can be extended to support weakly supervised learning by modifying the threshold. In reality the ability to support weakly supervised learning depends on the assumption that the number of video clips labelled "0" is much larger than those labelled "1", which mathematically derives Eq. 10.

More precisely, we have

$$\begin{aligned} & D_{KL}(Q||P_0) - D_{KL}(Q||P_1) \\ = \;& \textstyle\sum_{i=0}^{K-1} q_i ln \frac{q_i}{p(v_i|c=0)} - \sum_{i=0}^{K-1} q_i ln \frac{q_i}{p(v_i|c=1)} \qquad (13)\\ = \;& \textstyle\sum_{i=0}^{K-1} q_i (ln \frac{q_i}{p(v_i|c=0)} - ln \frac{q_i}{p(v_i|c=1)})\\ = \;& \textstyle\sum_{i=0}^{K-1} q_i ln \frac{p(v_i|c=1)}{p(v_i|c=0)}\\ = \;& ln \textstyle\prod_{i=0}^{K-1} (\frac{p(v_i|c=1)}{p(v_i|c=0)})^{q_i}. \qquad (14)\end{aligned}$$

Suppose that the input video clip is labelled "0" . This indicates that $q_i = 0$ for any $v_i \in G$. Then we have

$$\begin{aligned} & D_{KL}(Q||P_0) - D_{KL}(Q||P_1) \\ = \;& ln \textstyle\prod_{i \notin G} (\frac{p(v_i|c=1)}{p(v_i|c=0)})^{q_i}\\ = \;& ln \textstyle\prod_{i=0}^{K-1} (\frac{\frac{1+n_i^1}{N^1+K}}{\frac{1+n_i^j}{N^0+K}})^{q_i}\\ = \;& ln \textstyle\prod_{i=0}^{K-1} (\frac{1+n_i^1}{1+n_i^0} \times \frac{N^0+K}{N^1+K})^{q_i}\\ \approx \;& ln \textstyle\prod_{i=0}^{K-1} (r \times \frac{N^0+K}{N^1+K})^{q_i}\\ = \;& ln (r \times \frac{N^0+K}{N^1+K})^{\sum_{i \notin G} q_i}\\ = \;& ln (r \times \frac{N^0+K}{N^1+K}). \qquad (15)\end{aligned}$$

Now suppose the input video clip is labelled "1". In this case, at least for some $v_i \in G$, $q_i \neq 0$. Since code words from G only occur in the video clip labelled "1", then for any $v_i \in G$, we have

$$\begin{aligned} p(v_i|c &= 0) = \frac{1}{N^0 + K};\\ p(v_i|c &= 1) > \frac{1}{N^1 + K}. \end{aligned} \tag{16}$$

Eq. 16 leads to the following result as

$$\begin{aligned}\frac{p(v_i|c=1)}{p(v_i|c=0)} &> \frac{\frac{1}{N^1+K}}{\frac{1}{N^0+K}} \\ &= \frac{N^0+K}{N^1+K} \\ &> r \times \frac{N^0+K}{N^1+K} \qquad s.t \quad v_i \in G. \quad (17)\end{aligned}$$

Applying Eq. 17 to Eq. 14, we have

$$\begin{aligned}& D_{KL}(Q||P_0) - D_{KL}(Q||P_1) \\ = \;& ln \prod_{i \notin G} (\tfrac{p(v_i|c=1)}{p(v_i|c=0)})^{q_i} \prod_{i \in G} (\tfrac{p(v_i|c=1)}{p(v_i|c=0)})^{q_i} \\ = \;& ln \prod_{i \notin G} (r \times \tfrac{N^0+K}{N^1+K})^{q_i} \prod_{i \in G} (\tfrac{p(v_i|c=1)}{p(v_i|c=0)})^{q_i} \\ > \;& ln \prod_{i \notin G} (r \times \tfrac{N^0+K}{N^1+K})^{q_i} \prod_{i \in G} (r \times \tfrac{N^0+K}{N^1+K})^{q_i} \\ = \;& ln (r \times \tfrac{N^0+K}{N^1+K})^{\sum_{i=0}^{K} q_i} \\ = \;& ln (r \times \tfrac{N^0+K}{N^1+K}). \qquad (18)\end{aligned}$$

From Eq. 15 and Eq. 18, using the distance of KL divergence ($D_{KL}(Q||P_0) - D_{KL}(Q||P_1)$) as the criterion, theoretically there is a separation boundary $ln(r \times \frac{N^0+K}{N^1+K})$ which can be used to separate the video clips into the two classes.

3. EVALUATION

The MIT Traffic Database [6] is a 90-minute real world traffic surveillance video and is used here. This dataset (see Figure 2) is very challeging as it contains time varying levels of occlusion; a mix of vehicles, pedestrians and bicycles; waving trees and shadows. The events to detect are defined in Figure 2, and we follow experimental protocol and ground-truth of [13]. The video is cut uniformly into video clips of length 96 frames. In total there are 1728 video clips. Video clips are partitioned into five equal groups for a 5-fold cross validation, and performance is evaluated using the mean of the AUCs (Area Under the ROC curves) on the five folds. Each experiment is conducted with a different combination of classifier, feature descriptor and parameters. Experiments are conducted using four different features (discrete optical flow and three trajectory based approaches), and for each trajectory approach we vary the number of AC Fourier coefficients (1 and 5); and the number of clusters for K-means (30, a middle value automatically detected by the method Section 2.1, and 300).

Table 1 shows the means of AUCs over all feature and parameter settings. The KL-divergence approach reports the best mean of AUCs on the detection of Right Turn, Left Turn, and Jay Walking 1. It is ranked No. 2 in the detection of Jay Walking 2 event. It is important to note that, each entry in Table 1 (except the average column) is the mean of $19 \times 5 \times 4 = 380$ AUCs (19 feature and parameter settings, 5 folds, and 4 events). As a result, a small increment in Table 1 generally reflects an improvement. The computational time of the algorithm depends on the feature extraction approach used. When the KLT tracker and trajectory based features are used with 500 keypoints tracked, a one-hour video can be processed in less than 35 minutes [1]. The detection process using KL divergence runs very fast (similar to the naive Bayes model), and the time can typically be ignored in practice. Therefore, our approach supports real time detection.

Fig. 2. The Events of Interest in this paper

Table 1. The mean AUCs over all feature configurations for each event and each learning model

	Right Turn	Left Turn	JW 1	JW 2	mean
LDA	0.6807	0.5727	0.5895	0.5687	0.6029
LDA BG	0.6727	0.5840	0.5853	0.5583	0.6001
SVM	0.7160	0.6490	0.5942	0.6120	0.6428
Naive Bayes	0.7655	0.6887	0.6335	**0.6460**	0.6834
MIL-CS	0.7720	0.6943	0.6195	0.6218	0.6769
KL	**0.7880**	**0.7000**	**0.6402**	0.6418	**0.6925**

4. CONCLUSION

We propose a novel method for classifying video clips based on the presence of events using KL divergence. The proposed method outperforms the state-of-the-art methods in a real world surveillance dataset while preserving real-time detection.

[1] Running on a single core of an Intel Xeon 2.66 GHz Processor for a video file with the resolution of 704×576

5. REFERENCES

[1] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on (2008)*, 2008.

[2] S. Ali and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," in *IEEE International Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–6.

[3] T.M. Hospedales, Jian Li, Shaogang Gong, and Tao Xiang, "Identifying rare and subtle behaviors: A weakly supervised joint topic model," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 12, pp. 2451 –2464, dec. 2011.

[4] V. Mahadevan, Weixin Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[5] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[6] Xiaogang Wang, Xiaoxu Ma, and W.E.L. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 3, pp. 539 –555, march 2009.

[7] Tao Xiang and Shaogang Gong, "Beyond tracking: Modelling activity and understanding behaviour," *International Journal of Computer Vision*, vol. 67, no. 1, pp. 21–51, 2006.

[8] J. Xu, S. Denman, S. Sridharan, and C. Fookes, "Activity analysis in complicated scenes using dft coefficients of particle trajectories," in *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on*, sept. 2012, pp. 82 –87.

[9] Ji Liu Yang Cong, Junsong Yuan, "Sparse reconstruction cost for abnormal event detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[10] Xiaogang Wang, KengTeck Ma, Gee-Wah Ng, and W.EricL. Grimson, "Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models," *International Journal of Computer Vision*, vol. 95, pp. 287–312, 2011.

[11] Jingxin Xu, Simon Denman, Sridha Sridharan, Clinton B. Fookes, and Rajib Rana, "Dynamic texture reconstruction from sparse codes for unusual event detection in crowded scenes," in *Joint ACM Workshop on Modeling and Representing Events (J-MRE'11)*, 2011.

[12] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. pp. 79–86, 1951.

[13] J. Xu, S. Denman, V. Reddy, C. Fookes, and S. Sridharan, "Real-time video event detection in crowded scenes using mpeg derived features: a multiple instance learning approach," *Pattern Recognition Letters*, 2013.

[14] Jianbo Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, jun 1994, pp. 593 –600.

[15] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient similarity search in sequence databases," *Foundations of Data Organization and Algorithms (1993)*, 1993.

[16] A. Naftel and S. Khalid, "Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space," *Multimedia Systems (2006)*, 2006.

[17] Ethem Alpaydin, *Introduction to machine learning*, MIT press, 2004.

[18] Stanley F. Chen and Joshua Goodman, "An empirical study of smoothing techniques for language modeling," in *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, Stroudsburg, PA, USA, 1996, ACL '96, pp. 310–318, Association for Computational Linguistics.