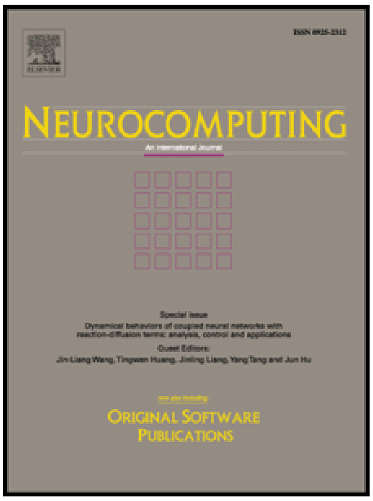

ISSN 0925-2312
ELSEVIER
NEUROCOMPUTING
An International Journal
Special issue
Dynamical behaviors of coupled neural networks with reaction-diffusion terms: analysis, control and applications
Guest Editors:
Jin-Liang Wang, Tingwen Huang, Jinling Liang, Yang Tang and Jun Hu
now also including
ORIGINAL SOFTWARE PUBLICATIONS

A unified framework with a benchmark dataset for surveillance event detection

Zhicheng Zhao, Xuanchong Li, Xingzhong Du, Qi Chen,
Yanyun Zhao, Fei Su, Xiaojun Chang, Alexander G. Hauptmann

PII: S0925-2312(17)31444-3
DOI: 10.1016/j.neucom.2017.04.079
Reference: NEUCOM 18815

To appear in: *Neurocomputing*

Received date: 8 December 2016
Revised date: 5 March 2017
Accepted date: 21 April 2017

Please cite this article as: Zhicheng Zhao, Xuanchong Li, Xingzhong Du, Qi Chen, Yanyun Zhao, Fei Su, Xiaojun Chang, Alexander G. Hauptmann, A unified framework with a benchmark dataset for surveillance event detection, *Neurocomputing* (2017), doi: 10.1016/j.neucom.2017.04.079

A unified framework with a benchmark dataset for surveillance event detection

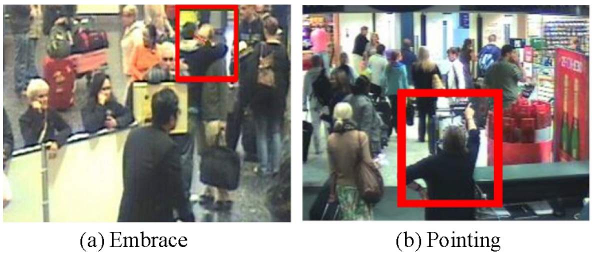
Zhicheng Zhao[a,b,*], Xuanchong Li[c], Xingzhong Du[d], Qi Chen[a], Yanyun Zhao[a], Fei Su[a], Xiaojun Chang[c], Alexander G. Hauptmann[c]

[a]*School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing*
[b]*Beijing Key Laboratory of Network System and Network Culture, Beijing*
[c]*School of Computer Science, Carnegie Mellon University, Pittsburgh*
[d]*School of Information Technology and Electrical Engineering, University of Queensland, Queensland*

Abstract

As an important branch of multimedia content analysis, Surveillance Event Detection (SED) is still a quite challenging task due to high abstraction and complexity such as occlusions, cluttered backgrounds and viewpoint changes etc. To address the problem, we propose a unified SED detection framework which divides events into two categories, i.e., short-term events and long-duration events. The former can be represented as a kind of snapshots of static key-poses and embodies an inner-dependencies, while the latter contains complex interactions between pedestrians, and shows obvious inter-dependencies and temporal context. For short-term event, a novel cascade Convolutional Neural Network (CNN)−HsNet is first constructed to detect the pedestrian, and then the corresponding events are classified. For long-duration event, Dense Trajectory (DT) and Improved Dense Trajectory (IDT) are first applied to explore the temporal features of the events respectively, and subsequently, Fisher Vector (FV) coding is adopted to encode raw features and linear SVM classifiers are learned to predict. Finally, a heuristic fusion scheme is used to obtain the results. In addition,

*Corresponding author
Email addresses: zhaozc@bupt.edu.cn (Zhicheng Zhao), me@xuanchong.li (Xuanchong Li), domainxz@gmail.com (Xingzhong Du), chenqi1990@163.com (Qi Chen), zyy@bupt.edu.cn (Yanyun Zhao), sufei@bupt.edu.cn (Fei Su), cxj273@gmail.com (Xiaojun Chang), alex@cs.cmu.edu (Alexander G. Hauptmann)


(a) Embrace

(b) Pointing

Figure 1: Examples of two pre-defined surveillance events from different camera views.

a new large-scale pedestrian dataset, named SED-PD, is built for evaluation. Comprehensive experiments on TRECVID SEDtest datasets demonstrate the effectiveness of proposed framework.

Keywords: surveillance event detection, pedestrian dataset, pedestrian detection, cascade CNN

1. Introduction

Video monitoring systems provide a necessary support for handling various incidents and emergencies. Rapidly-growing amount of surveillance videos boom the developments of content-based multimedia understanding, and SED [1, 2] has become one of the most challenging tasks. As illustrated in Fig. 1, the goal of SED is to localize events of interest (e.g., "Embrace" and "Pointing") from a large-scale monitoring video dataset. These events are easily related to potential menaces of the public security. Therefore, detecting specific events quickly and accurately becomes an important and indispensable task.

SED contains three key elements: whether, when and where does the event happen. It concentrates on human behaviors and group activities, thus the detection complexity is much higher than action recognition from simple KTH dataset [3], in which actions are simulated individually under a controlled setting. Additionally, unlike datasets Hollywood2 [4], UCF-101 [5] and Sports-1M [6] etc, only actions are concerned, SED also emphasizes the interactions of individuals. Moreover, videos are often collected from busy environments such as the airport and subway, where monitoring conditions (viewpoints, scales and illumination etc.) are varied, and the people stream is bustling and serious occlusion is common.

Therefore, some events become extremely complicated and abstract, SED in multi-camera views, despite recent progresses [7, 8, 9, 10] were made, is still in its infancy. Compared with other unitary surveillance tasks such as pedestrian detection [11, 12, 13], action recognition [6, 14] and person re-identification [15], a great number of literatures exist, and the work of SED is quite few. As an integrated project of above tasks, SED not only relies on the capturing of spatial information at the semantic level, but also requires to fuse motion, temporal relationships and contexts, which intensify the difficulties of SED: 1) data imbalance is quite serious. For most of events, only a few positive samples are available for training, and meanwhile, there are no bounding boxes annotated. 2) surveillance videos are unstructured and do not follow any particular distribution, thus a host of statistical learning-based methods are intractable. 3) SED has to suffer from cluttered surrounding and serious occlusion, and large inner-variances of events weaken the capability of the detector.

We observe that: 1) surveillance events are centered at human actions and activities. 2) they have strong dependencies: short-term events characterize unique actions (inner-dependencies), and could be represented by a series of snapshots which are composed of key-poses, while long-duration events consist of multiple interactions (inter-dependencies). As a result, we address SED problem from two aspects. First, to detect the pedestrian in realistic monitoring scenes, we build a new large-scale pedestrian dataset−SED-PD, and then propose a cascade CNN−HsNet to identify the human. Second, a unified SED framework is presented to detect two kinds of events respectively: for short-term events, multiple CNN models are trained to classify, and for events with long-interactions, the temporal contexts are modeled by fusing DT and IDT. Our contributions can be summarized as follows:

1. Based on a systemically analysis on recent TRECVID SED practices, we

propose a unified SED framework, which provides a practical solution by combining the advantages of conventional video representations and deep CNN models.

2. SED-PD: a large-scale pedestrian dataset is built to deal with the challenge of serious occlusion. Meanwhile, a fast cascade CNN−HsNet is constructed to detect the pedestrian effectively.
3. Available surveillance events are divided into two categories: short-term events and long-interaction activities, and then CNN-based and trajectory-based methods are proposed to detect them respectively.
4. Comprehensive experiments on TRECVID SEDtest datasets demonstrate the effectiveness of proposed framework.

The rest of the paper is organized as follows: in Section 2, we review related works of SED. Section 3 describes our SED-PD dataset, and Section 4 introduces the proposed framework and algorithm. Section 5 discusses the experimental results. Finally, we conclude our work in Section 6.

2. Related work

SED can be roughly divided into two phases. Before 2008, SED typically followed a hierarchical pipeline: object detection and tracking were leveraged at the low-level, and predefined actions were recognized at the middle-level, and finally at the high-level, activities were encoded based on the action primitives. In this period, successful action recognition mainly benefited from the combination of Bag-of-Words (BoW) framework, local spatio-temporal features and SVM classifier. Space-Time Interest Points (STIP) [16] and HOG [17] and Histogram of Optical Flow (HOF) [18] were representative feature detectors and descriptors respectively. In order to explore the temporal relationships between primitive actions of an event, graph-based models were proposed. Hidden Markov Model (HMM) [19], neural networks [20, 21, 22], Bayes Network [23, 24, 25, 26, 27] were used. However, due to the lacking of public large-scale datasets, most experiments were conducted on small-scale self-built datasets.

In 2008, SED was introduced to the TRECVID serial and thus attracted increasing attention [2]. Correspondingly, annual evaluation results highlight the progress of action recognition and event detection, which undergoes the evolution from modeling events based on hand-crafted features to representing videos with deep features [6]. Instead of local spatio-temporal cuboids, trajectory-based methods became a crucial branch of action recognition. With Motion Boundary Histogram (MBH) [18] and FV encoding [28], methods based on DT [29] and IDT [30] achieved the best performance among conventional models.

Recently, encouraged by the results of CNNs achieved in image classification [31], researchers began to mine temporal correlation and then built novel CNN models for action recognition and video analysis. For example, 3D CNN [6], two-stream CNN [14] and C3D network [32]. [33] employed Long Short-Term Memory (LSTM) network to seek temporal relationships and obtained the state-of-the-art results on the UCF-101 and Sports-1M datasets. [34] also used LSTM to explore the temporal context of events and achieved a promising performance.

Although great successes have been achieved by motion-based CNN models, very limited applications [35] are reported in recent SED practices. One reason is that the SED dataset is quite different with above datasets, thus existing models are hard to be transfered to SED. Moreover, due to high complexity, not all events in SED dataset can be modeled by a deep network. Actually, after taking a panoramic view of the recent TRECVID SED evaluations, we find that the conventional representation methods still hold a dominant place. For instance, [9] used MoSIFT feature with SVM detector, and achieved outstanding results in SED2012 dataset. The deformable part-based model (DPM) [36] also shown a high generalization ability.

However, existing SED methods share three main limitations: 1) different dependences of the events were neglected and all events were treated in the same way; 2) probabilistic frameworks such as HMM and CRF were difficult to represent complex interactions between individuals; 3) the introduction of the deep model was quite limited and unsuccessful. As a result, comprehensive

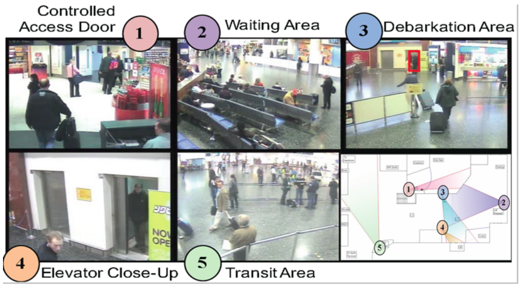


Figure 2: SED videos captured from 5 locations in Gatwick airport[2].

schemes need to be considered. In this paper, we combine the merits of the conventional model and the deep network and propose a unified SED framework.

3. Pedestrian Dataset: SED-PD

To provide a true benchmark for pedestrian detection in highly occluded scenes, we label a large-scale surveillance video dataset, i.e., TRECVID SED-train2008 [2]. It contains 50 full-length videos, which were collected from five cameras in London Gatwick airport, where serious occlusions and frequent interactions are common. Fig. 2 shows the camera views, displaying (from left to right, top to bottom) a controlled access door, a waiting area with benches, a waiting area with kiosks, an elevator close-up view and a transit area.

Table 1: The SED-PD dataset.

SED-PD	training set	validating set	test set
Person	124,000	63,000	210,000
Frame	15,000	7,500	22,500

We semi-automatically annotate 9 videos and collect total 397,000 pedestrian instances. Table 1 shows the information. The resolution of the video frame is 720x576 pixels. The dataset is diviede into three subsets: training set, validating set and test set. The training set contains 124,000 persons and covers 15,000 frames, and set includes 63,000 pedestrains and 7,500 frames, and the test set

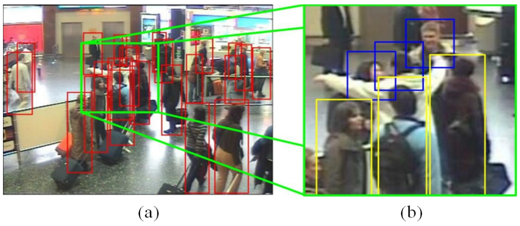


Figure 3: (a) An annotation example from SED-PD. Red bounding boxes denote the pedestrians. (b) Partial details. Pedestrians in blue boxes are heavily occluded by pedestrians in yellow boxes.

has 210,000 individuals and covers 22,500 frmes. Each frame averagely contains 9 persons. Fig. 3 shows an example.

We provide the bounding box of the head-shoulder of each pedestrian. The labeling precedure is: first, based on [37], the pedestrians of each frame are detected, and then all candidates are manually inspected by the annotator. Meanwhile, we generate more than 7 millions negative samples by a random sampling scheme: 1) suppose an image has N pedestrians, $60N$ bounding boxes with the same size of pedestrians are randomly yielded; 2) delete the negative samples, whose overlaps with pedestrians exceed 0.5. We name the dataset the SED-PD [1], which is as large as Caltech dataset [38]. The biggest difference of them is that our dataset contains more occlusion, while the backgrounds of the latter are more various.

4. The Proposed Framework

As shown in Fig. 4, the proposed framework is mainly composed of 4 parts: pedestrian detection, short-term event classification, long-duration activities identification and results fusion. Firstly, based on SED-PD, we train a cascade CNN (HsNet) to detect the pedestrian. Secondly, according to intra-dependencies and inter-dependencies of events, we classify events into two categories: short-term events and long-duration activities. The former is charac-

[1] http://www.bupt-mcprl.net/datadownload.php

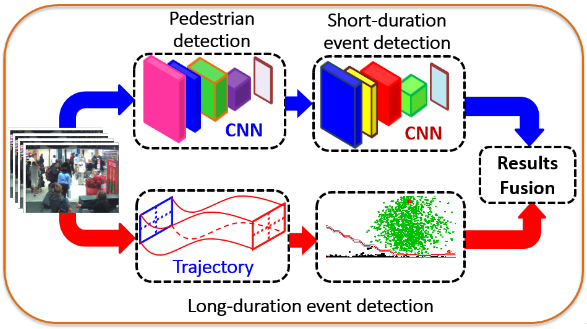


Figure 4: The unified framework of SED. It mainly includes 4 parts: pedestrian detection, short-term event detection, long-duration event detection and results fusion.

terized with unique actions, and can be treated as a kind of static snapshots of key-poses. Hence, using the results of pedestrian detection, we re-train new CNN models to classify them. Regarding the long-duration events, in order to narrow inherent semantic gap between motion patterns and complex events, we model temporal context of events by DT and IDT, and apply FV to encode raw features. Afterwards, fast and effective linear SVM classifiers are learned to detect them. Finally, a score-based late fusion scheme is applied to obtain the final results.

4.1. Pedestrian Detection by Cascade CNN (HsNet)

Employing deep models to detect the pedestrian become a trend, and a series of experiments indicate that they can generate more interpretable results than traditional approaches based on hand-crafted features such as HOG and DPM. For example, JointDeep [11] improved the generalization ability by combining feature learning, a DPM and an occlusion model. Hosang et.al [13] applied AlexNet to obtain impressive performance on Caltech dataset. TA-CNN [39] reported state-of-the-art results on the Caltech and KTH datasets by joining semantic tasks, pedestrian and scene attributes.

Mentioned deep models can effectively handle partial occlusion in ordinary scenes such as Caltech and Hollywood2 datasets. For SED task, however, due to frequent interactions and critical occlusion, pedestrian detection is still a

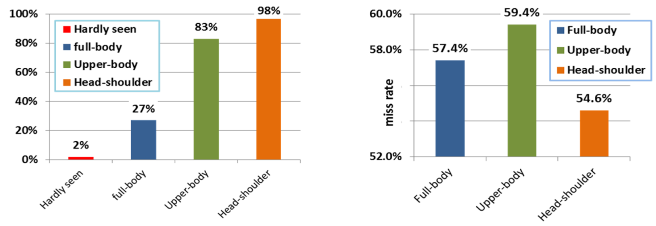


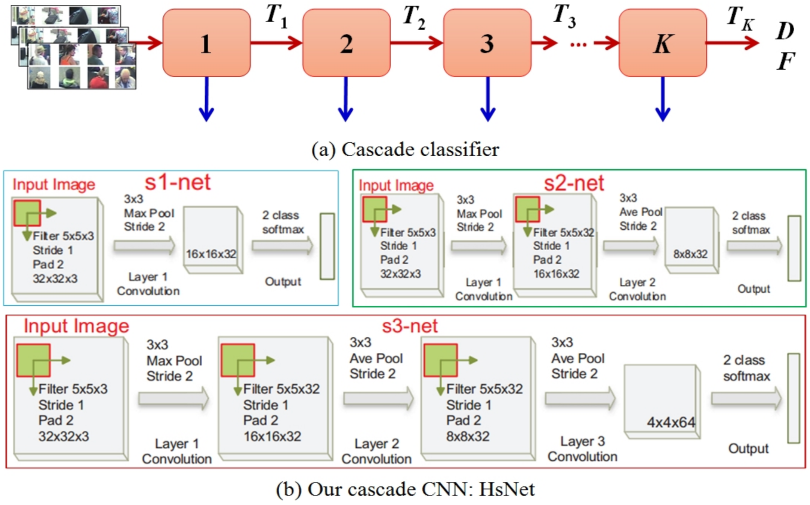
Figure 5: Two evidences for head-shoulder-based pedestrian detection. (a)Occurrence frequency of body parts on SED-PD. (b)The miss rate of three parts test by the Cifar-10 model.

challenging problem. Moreover, the network architectures of existing models such as JointDeep [11] and SDN [12] are too complicated to apply to a real-time system. Hence, we address this problem from three aspects: 1) only head-shoulders are used. 2) CNN is leveraged to capture the most discriminative information. 3) a cascade detection structure is constructed to speed up the detection.

The first practice derives from two key evidences: the statistical analysis and the experimental result. Firstly, we evaluate the occlusion distribution of SED-PD by counting the average occurrences of three kinds of body parts, i.e., head-shoulder, upper-body and the whole body. The result is shown in Fig. 5(a). We can see that most pedestrian instances (73%) show incomplete body parts, while 98% head-shoulders are kept. Secondly, we conduct a preliminary human detection experiment on SED-PD with the Cifar-10 network [40], and the comparison is shown in Fig. 5(b). The result is consistent with Fig. 5(a), i.e., the head-shoulder is the most distinctive part in SED.

Afterwards, we propsoe a cascade CNN (HsNet) to detect the pedestrain, Fig. 6(a) gives the flowchart. Let K denotes the number of cascade of CNN models, f_k and d_k represent the false positive and the detection rates of the k-th CNN model respectively. T_k is threshold for tuning f_k and d_k. The final false positive and detection rates are defined as

$$F = \prod_{k=0}^{K} f_k \tag{1}$$



(a) Cascade classifier

(b) Our cascade CNN: HsNet

Figure 6: Schematic depiction for the person detection. (a)cascade classifiers. (b)the architectures of three subnets.

$$D = \prod_{k=0}^{K} d_k \tag{2}$$

The learning goal of the cascade network is to achievie a satisfactory detection rate D, meanwhile, decreases the false positive rate F as far as possible. Once wanted D and F are predefined, based on the performance of the k-th model, parameters K, T_k, f_k and d_k can be determined. In our implementation, in order to balance the training time and the performance, we empirically set K=3. Furthermore, unlike popular cascade network, the same classifier for all cascades are chosen, our cascade network (HsNet) concatenates different CNN models.

Fig. 6(b) gives the network architecture, which is composed of 3 subnets: s1-net, s2-net and s3-net. They are built from simple to complex so as to improve the discriminative representation for pedestrians step-by-step:

- s1-net: only contains a convolutional, a pooling and a two-class softmax layer, and is designed to quickly get rid of majority obviously negative candidates.

- s2-net: includes 2 convolution layers and is used to further reduce the number of false detections.

- s3-net: is the most complex subnet. It follows the similar architecture of Cifar-10 network to strengthen the learning of hard samples and can achieve the most discriminative power.

Since the number of cascade K is fixed to 3, the training objective can be simplified as: obtaining the optimal parameters to guarantee a high detection rate d_k. During the course of HsNet training, all positive samples are input into s1-net, s2-net and s3-net. For negative samples in k-the subnet, when the output scores are higher than the threshold T_k, they will be fed into the $(k{+}1)$-th subnet. The implementation of HsNet will be introduced in Section 5.

The detection flowchart is: for a video frame I_i, a multi-scale sliding window scheme is first used to yield input patches P_j. Afterwards, HsNet is applied to determine the candidate head-shoulders. Fianlly, the Non-Maximum Suppression (NMS) algorithm is used to eliminate redundant detection boxes and obtain the final results H_k.

4.2. Event Detection

Existing SED approaches treat with different events in the same way. Actually, events are featured with different dependencies: a kind of events often shortly occur on a single pedestrian and can be represented by simple actions, showing strong inner-dependencies. Another kind of events are complex and contain long interactions between individuals. In these cases, the inter dependencies become crucial. Therefore, our SED goes through two parallell detection pipelines: short-term event detection and long-duration one detection.

4.2.1. Short-term Event Detection

Key-poses classification. For an arbitrary event, if it is dominated by a pure and transitory action, and meanwhile, the subject of the event usually keeps a static gesture for a while, the event is defined as a short-term event.

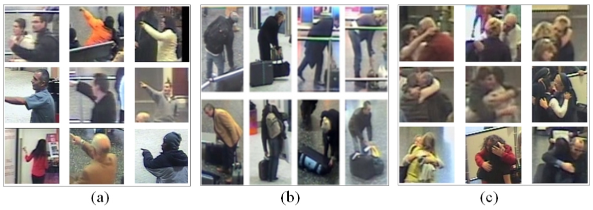


Figure 7: Short-term events: (a) “Pointing”. (b) “ObjectPut”. (c) “Embrace”. Note that we normalize the direction of “Pointing” to left direction.

In this paper, we regard short event as a series of key-poses, as shown in Fig. 7, and then apply CNN models to detect them. However, fairly insufficient positive events highlight the data imbalance problem. To decrease the impact and enhance the generalization ability of models, we take 2 measures: 1) augmenting the positive samples by conducting multiple transformations such as mirror, randomly rotation and crop etc. Finally, we increase the size of the positive data set by a factor of 4; 2) employing the same architecture of HsNet, we train novel CNN models to classify this type of events. Given a video, the detection of the short-term events can be summarized as 3 steps:

1. Pedestrain detection. For each video frame, the HsNet is first employed to detect the pedestrains, and then the sizes of all detected bounding boxes are expanded by 1.5 times.
2. Key-pose classification. The pedestrain is resized into 32x32 pixels, and then is input into the key-pose models to classify respectively.
3. Event localization. Based on the nearest-neighbor (NN) search, a fast object trackig algorithm (see Algorithm 1)is used to associate consecutive key-poses into an individual event.

key-pose tracking for event localization. According to spatial position and temporal consistency, we adopt a simple tracking algorithm based on NN search to link key-poses into events. For key-pose p_i, NN search traverses all poses q_j in next ΔT frames to find the best matching pair (p_i, q_{j*}), and meanwhile, follow a spatial constraint. i.e., if their Intersection-Over-Union (IoU) ratio exceeds

Algorithm 1 The NN search for event localization

```
Require: The key-poses coordinate set of the first frame: $P = \{p_0^i\}_{i=1}^{N_0}$; The coordinate set of all key-poses: $Q = \{q_t^j\}$, $j = 1, 2, ..., N_t$, $t = 1, 2, ..., T$. $T$: the length of tracking window; $N_0$ and $N_t$: the number of detected key-poses in the first and the $t$-th frame respectively.
Ensure: Tracking sequence of $S = \{s_k\}$.
Initialize $N_0$ tracking sequences,
for $i = 1$ to $N_0$ do
    for $t = 1$ to $T$ do
        for $j = 1$ to $N_t$ do
            Compute the distance: $d_{ij} = ||p_i\text{-}q_j||^2$;
        end for
    end for
    Find the nearest neighbour of $p_i$ : $d_{ij*} \leftarrow \arg \min_j d_{ij}$;
    Compute the ratio of IoU between $p_i$ and $q_j*$;
    if IoU>0.5 then
        Record $q_j*$ and find next NN within next $T$ frames;
    else
        Create a new tracking sequence $s_{j*}$ for $q_{j*}$;
        Add a new tracking for $q_{j*}$;
    end if
end for
```

0.5, this pair is recorded. Otherwise, a new event will be labeled from q_{j*}, and correspondingly, a new tracking sequence will be created. Here, we apply l_2-norm to calculate the distance of two poses p_i and q_j. The tracking algorithm can be described in Algorithm 1. Here, $q_t^j = (x_t^j, y_t^j)$ denotes the j-th key-pose in the t-th frame, and (x_t^j, y_t^j) is the central coordinate of the key-pose. In experiments, we set ΔT=25.

4.2.2. Long-duration Event Detection

If an event includes multiple human interactions with different actions and holds a relative long time, the event is defined as the long-duration event.

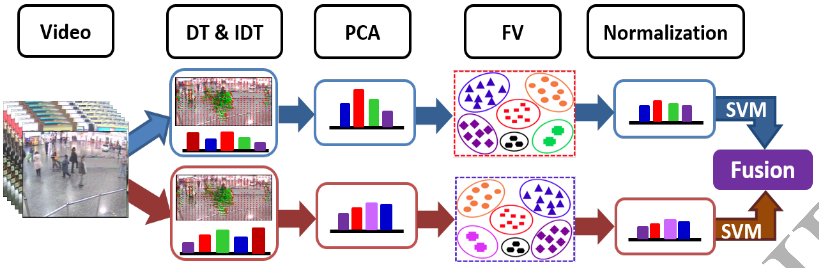


Figure 8: The pipeline of video representation and long-duration event detection.

For this kind of events, static postures are hard to represent. In addition, our preliminary experiments show that temporal deep networks such as LSTM and two-stream CNNs can also not play positive role. [35, 41] use CNN models, but the results are not satisfactory. Therefore, we adopt a trajectory-based method to extract the contexts. Fig. 8 shows the pipeline, which mainly consists of 5 parts: 1) local-level feature extraction; 2) dimension reduction by PCA and whiten PCA; 3) video representation based on FV encoding and normalization; 4) linear SVM classification for events; 5) results fusion.

Low-Level feature extraction. In order to accelerate the feature extraction, for an input video V, we firstly resize it to 320x240 pixels, and then split it into N_v temporal segments with a fixed length L_v and 50% overlap, i.e., $V = \{V_1, ..., V_{N_v}\}$. For a complex event, local motion is distinctive information. Compared with some sparse detectors such as STIP etc, DT and IDT, since integrates much richer visual cues, can more steadily capture special motion patterns in complex scenes.

In DT, five kinds of descriptors, i.e., dense trajectory, HOG, HOF, MBHx, and MBHy are yielded to represent the spatio-temporal relationships of videos. IDT enhances the representation of object motion by removing the “global motion” of videos. In fact, IDT regards the dominant motion between two frames as the global motion to remove. In SED, videos are always recorded from fixed angles and no camera motions happen, thus the dominant motion actually comes from the people stream, where the interference from the irrelevant persons could be decreased, but true local motions of events would also be eliminated. In this situation, DT will be more suitable. In our experiment, DT and IDT show a complementary feature so that we combine them.

Raw feature encoding with Fisher vector. To represent each video clip V_i, we encode the various number of raw features into a fixed dimension. For the IDT and DT feature, the PCA operation is beneficial to speed up the computation and remove the correlation of the data. In recent literature [42], the whiten PCA has been proved to be superior to PCA in action recognition. In our implementation, before raw features are encoded, both PCA and whiten PCA are employed, and their performance comparison will be discussed in Section 5.

Afterwards, we employ FV to encode DT and IDT descriptors. Based on Fisher kernel [28], FVs model the video through optimizing a generative Gaussian mixture model (GMM) $p(\mathbf{x};\theta)$

$$G_\theta^{\mathbf{x}} = \nabla_\theta \mathrm{log} p(\mathbf{x};\theta) \tag{3}$$

$$\begin{aligned} p(\mathbf{x};\theta) &= \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x};\mu_k,\Sigma_k) \\ \forall k &: \pi_k \geq 0, \Sigma_{k=1}^{K}\pi_k = 1 \end{aligned} \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^D$ represents the feature descriptor, and K is the number of Gaussian components. The GMM parameters $\theta = \{\pi_k, \mu_k, \Sigma_k, k = 1, ..., K\}$. π_k, μ_k and Σ_k denote the mixture weight, mean vector and covariance matrix respectively. Σ_k is assumed to be diagonal with variance vector δ_k^2. θ could be estimated by Expectation-Maximization (EM) with Maximum Likelihood (ML) estimation. FV generates a $2DK$-dimension signature $\mathbf{x}$ for D dimensional descriptors and K Gaussian clusterings. Let $\mathbf{x}$ stands for a set of descriptors extracted from a video clip V_i. The final FV of $\mathbf{x}$ is the concatenation of gradient vectors

$$\mathcal{G} = \left\{G_{\mu,1}^{\mathbf{x}}, G_{\delta,1}^{\mathbf{x}}, ..., G_{\mu,k}^{\mathbf{x}}, G_{\delta,k}^{\mathbf{x}}, ..., G_{\mu,K}^{\mathbf{x}}, G_{\delta,K}^{\mathbf{x}}\right\} \tag{5}$$

where $G^{\mathbf{x}}_{\mu,k}$ and $G^{\mathbf{x}}_{\delta,k}$ are D-dimensional gradients with respect to the mean vector μ_k and standard deviation δ_k of the k-th component.

$$G^{\mathbf{x}}_{\mu,k} = \frac{1}{\sqrt{\pi_k}}\omega_k(\frac{x_i - \mu_k}{\delta_k}) \tag{6}$$

$$G^{\mathbf{x}}_{\delta,k} = \frac{1}{\sqrt{2\pi_k}}\omega_k(\frac{(x_i - \mu_k)^2}{\delta_k^2} - 1) \tag{7}$$

where ω_k is the soft assignment of descriptor x_i with respect to the k-th component.

$$\omega_k = \frac{\pi_k \mathcal{N}(\mathbf{x}; \mu_k, \Sigma_k)}{\Sigma_i^K \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \Sigma_i)} \tag{8}$$

Prior to $\mathcal{G}$ is applied to train the classifier, the power normalization and ℓ_2 normalization are successively introduced to adjust the absolute differences of video clips so that FV representation could fit to the change of the numbers of local descriptors and Gaussian components.

We observe that the bigger of the size of Gaussian clustering is, the sparser the FV will become, which results in some dimensions of the FV distribution concentrate to zero. Hence, we firstly apply power normalization to suppress the tendency. And then, ℓ_2 normalization is employed to remove the particularity of specific events so as to enhance the generalization ability of FV representation.

Event classification. Based on normalized video representations, one-against-rest linear SVM classifiers are learned to detect events. Specifically, firstly, we orderly train one SVM classifier for each event and each camera, and then a later-fusion scheme based on the scores of the classifiers is used to get the final results. It consists of 3 steps:

1. Data augmentation. we treat the video clips which have 50% overlap with the ground truth as the positive samples.
2. Detector training. we use LIBLINEAR [43] with twofold cross-validation to train linear SVM detectors. Moreover, via a robust probability function [44], we transform the decision values into probabilities. Finally, multiple detectors are learned for per feature, and each of them only focuses on one event under one camera.

3. Event merge. we first learn an optimal threshold for each detector by cross-validation, and then apply NMS to merge the adjacent positive clips. The merging rule is: filtering the clips whose output probabilities are below predefine threshold firstly, and then attributing the adjacent clips' labels to the clip whose probability is the local maximum.

5. Experimental Results

5.1. Datasets and experimental setup

5.1.1. The dataset

According to the Miss Rate (MR), we first compare our pedestrian detection algorithm with state-of-the-art methods on the SED-PD dataset. Then we evaluate the SED framework on TRECVID SED2015 dataset [1], containing seven events. i.e., "CellToEar", "Embrace", "ObjectPut", "Pointing", "PeopleMeet", "PeopleSplitUp" and "PersonRuns". Table 2 provides a brief description. Note that due to fairly high complexity and the scale of this dataset, the TRECVID SED evaluations over the years always focus on seven ad hoc events.

Table 2: The test events.

Event	Description
CellToEar	someone puts a cell phone to his/her ear.
ObjectPut	someone drops or puts down an object.
Embrace	someone puts one or both arms part way around another person.
PeopleMeet	one or more people walk up to one or more others, stop and the conversation occurs.
PeopleSplitUp	when one or more people separate themselves from a group of two or more people, who are either standing, sitting or moving together communicating, and then leave the frame.
PersonRuns	someone runs.
Pointing	someone points.

The training data of SED2015 consists of about 100 hours videos and the testing set has additional 45 hours. This dataset is extremely challenging due to multiple confounding factors such as dense group activities, serious intra-variances of the same events (e.g., PeopleSplitUp) from different cameras and tiny object (e.g., CellToEar) etc. In addition, the annotations of events only contain category labels and temporal extents without bounding boxes. Furthermore, a rough statistics on training set show that the proportion of all positive events are not more than 5%. The least frequent event, "CellToEar" occupies less than 0.5% indeed, which pop outs the problem of the data imbalance.

In experiments, we partition the training data into two parts: 2/3 of them is used to train and 1/3 is used to validate.

5.1.2. The setting

Following the SED evaluation, we use the Normalized Detection Cost Rate (NDCR) [2] to evaluate the performance. NDCR is a weighted linear combination of the system's two error types: Miss Detections Probability (P_{miss}) and False Alarms Rate (R_{FA}). The lower of the NDCR is, the better of the performance will be. It is defined as:

$$\begin{aligned} NDCR &= P_{miss} + 0.005 \times R_{FA} \\ P_{miss} &= N_{miss}/N_{Ref} \\ R_{FA} &= N_{false}/N_{CamHrs} \end{aligned} \tag{9}$$

where, N_{miss} is the number missed detections, and N_{Ref} is the number of true event observations. N_{false} is he number of incorrect detections. N_{CamHrs} is the number of camera hours of processed material.

In the evaluation, NDCR uses two criteria: the Actual NDCR (ADCR) and the Minimum NDCR (MDCR). The ADCR is the primary metric, which is computed by restricting the putative observations to those with true actual decisions. The MDCR is a diagnostic metric that is found by searching the detection error tradeoff curve for the minimum cost. The difference between the ADCR and MDCR indicates the benefit a system could have gained by selecting a better threshold.

Experiments are conducted on a PC with an Intel Core i7 CPU (3.4 GHz)

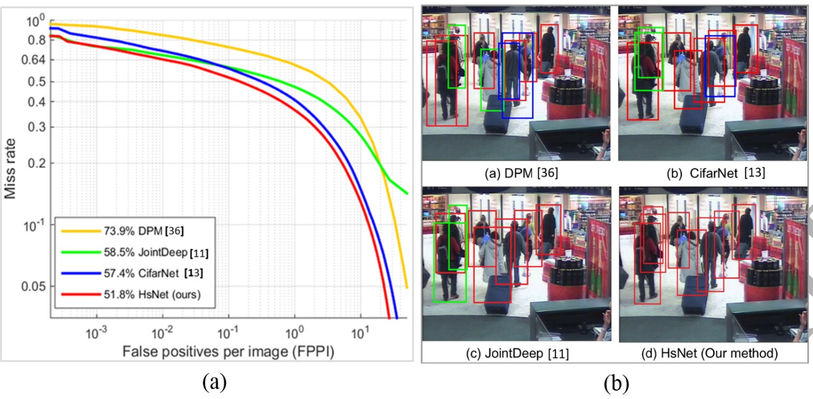


Figure 9: Comparison with state-of-the-art methods. (a) the miss rate. (b) a detection instance: red bounding boxes are correct results, and green and blue ones stand for the missing and flase detection respectively.

and 8GB memory, and an NVIDIA Titan-X GPU.

5.2. Pedestrian Detection

We apply Stochastic Gradient Descent (SGD) with back-propagation to optimize the HsNet. We bias the sampling towards the positive data because it is extremely rare. The deep model is implemented by Caffe toolbox [45], and the parameter configuration is as follows: 32 positive samples and 96 negative ones are uniformly sampled to construct a mini-batch in each iteration. All layers are randomly initialized from a Gaussian distribution with $\delta = 0.01$. The initial learning rate is 0.01 and will be dropped 90% for every 5,000-iteration. The momentum is set to 0.9 to speed up the learning, and the maximum iteration number is 60,000. The detection rate of the HsNet is set to $D = 0.995$.

We compare our model with state-of-the-art methods, i.e., DPM+LSVM [36], JointDeep [11] and CifarNet [13]. Fig. 9(a) shows the comparison. We can see that the DPM-based model performs the worst due to the limitation of the hand-crafted features. Although [11] and [13] leverage the strength of the deep network, they still cannot deal with the highly occluded pedestrians, thus get high MR (58.5% and 57.4% respectively). In contrast, our HsNet can effectively

Table 3: The comparison for runtime efficiency.

Device	Model (second/frame)			
	Our	CifarNet[13]	DPM[36]	JointDeep[11]
CPU	1.02	1.69	3.76	31.52
GPU	0.33	0.56	–	–

restrain the appearance variations of pedestrians. A clear example is shown in Fig. 9(b). Compared with other methods, our model effectively detects most of humans and obtains the lowest false detection rate.

Table 3 compares the runtime efficiency. Our HsNet obtains the lowest computation cost and we owe it to two reasons. First, detecting the head-shoulder instead of the full-body is a much beneficial scheme, which drives us to design a simple deep network to achieve the competitive accuracy with complex models such as JointDeep. Second, the cascade architecture quickly filters out most of the negetive samples in the early stages, thus further reduces the detection time.

5.3. Evaluation for Event Detection

5.3.1. Short-term event classification

According to previous definition, three events in SED2015 dataset, i.e., “Embrace”, “ObjectPut” and “Pointing” are viewed as short-term events. Correspondingly, three cascade CNN models are learned to classify them respectively. In our experiments, in order to demonstrate the effectiveness of cascade CNN, a single CNN model is also trained for comparing. As is shown in Fig. 10, it includes 3 convolution layers, 2 fully-connect layers and a softmax classifier.

Firstly, on the SEDtest2015, we compare cascade CNN, single CNN and the best results of other systems at SED2015 evaluation. Trajectory-based approach (is represented as “DT+IDT” in Table 4) is also tested to illustrate the superiority of CNN-based method. According to this Table, We can see that cascade CNN achieves the best performance, and the single CNN and trajectory-based

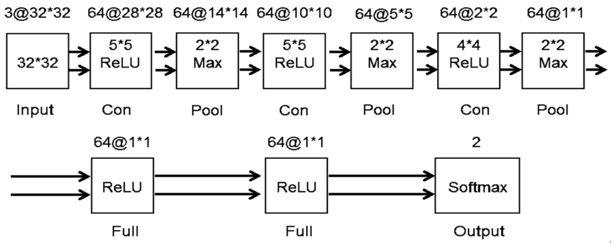


Figure 10: The single CNN architecture for short-term event detection.

methods follow successively, and outperform other methods at SED2015 evaluation. The results indicate that for short events, the CNN model can capture discriminative gesture information. Additionally, the cascade CNN is better than single CNN. The results show that the cascade architecture can not only partly overcome data imbalance by eliminating most of negative samples but also gradually concentrate the learning of hard samples, thus boosts the performance.

Moreover, we referentially compare the best results of SED2013 [10] and SED2014 [46] evaluations, and Chen's method [9] that was conducted on SED-2012. Note that the test data of annual SED evaluation is different and the annotations are also not available, thus we qualitatively compare these numbers. We notice that our "Embrace" obtains a new record in recent years' SED competitions, which indicates cascade CNN has a superiority for short-term event detection.

5.3.2. Long-duration events evaluation

The rest of four events of SEDtest2015, i.e., "CellToEar", "PeopleMeet", "PeopleSplitUp" and "PersonRuns" are defined as the long-duration event. We firstly resized videos to 320x240 pixels, and then split them into temporal clips with a fixed length (L_v=60 frames) and keep 50% overlap with the adjacent clips. Because DT and IDT track feature points within 15 frames, thus for each video clip we append 15 subsequent frames behind the original 60 frames. Totally, 350k clips are generated to extract DT and IDT features respectively. After-

Table 4: Performance comparison for short-term events.

Event	Casade CNN		Single CNN		DT+IDT		SED2015[1][47]	
	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}
Embrace	0.790	0.790	0.833	0.833	0.868	0.845	0.991	0.945
ObjectPut	1.012	0.996	1.015	0.999	1.016	0.988	1.033	0.999
Pointing	1.004	0.998	1.008	0.999	1.014	0.994	1.073	0.997

Event	SED2014[1][46]		SED2013[1][10]		Chen[9]	
	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}
Embrace	0.811	0.811	0.835	0.833	0.800	0.779
ObjectPut	0.971	0.967	0.998	0.997	1.000	0.999
Pointing	0.999	0.995	0.995	0.989	1.017	0.992

[1] note: SED2015 denotes the best results of other systems at SED'2015 evaluation, and SED2014 and SED2013 stand for the best results at SED'2014 and SED'2013 evaluations respectively.

wards, both normal PCA and PCA-whiten are applied to reduce the descriptors to 80-D, and the power and ℓ_2 normalizations follow successively. Meanwhile, K-means clustering is applied to initialize the GMM training, where its covariance matrix of each mixture is defined as a diagonal one. The number of Gaussian components K is set to 128. Thirdly, a 116736-dimensinal feature vector is yielded and input into LIBLINEAR to train the SVM detector for each event. Finally, a late fusion is employed to combine two results from DT and IDT detectors.

To determine which feature and PCA are the best, we conduct heuristic experiments on the validation set. Our scheme averagely fuses the detection scores from two kinds of features and PCAs to rank the best results, i.e., 1) DT with normal PCA (DT-FV); 2) DT with whiten-PCA (DT-WFV); 3) IDT with PCA (IDT-FV); 4) IDT with whiten-PCA (IDT-WFV). Table 5 lists four mutual combinations and corresponding ADCR scores. As a whole, the combination of DT-WFV and IDT-WFV is better than others, thus we choose it as the final test method. Note that "DT+IDT" in Table 4 also adopts this combination.

Table 5: The comparison of different features and fusion schemes.

Event	Fusion method ($ADCR$)			
	IDT	DT-FV+ IDT-WFV	IDT-FV+ IDT-WFV	DT-WF+ IDT-WFV
CellToEar	1.005	1.000	1.003	1.004
Embrace	1.006	0.925	0.919	0.910
Pointing	1.014	0.992	0.989	0.985
ObjectPut	1.004	1.002	1.002	1.002
PeopleMeet	0.952	0.923	0.9369	0.929
PeopleSplitUp	0.961	0.893	0.903	0.886
PersonRuns	0.644	0.647	0.654	0.629

Table 6 gives the comparison. The results indicate that on SEDtest2015 dataset, our method outperforms state-of-the-art apporaches, which demonstrates the effectiveness of the proposed framework. The best results of recent years' SED evaluation are also referentially given. We find that the event "PersonRuns" achieves a new record, and "PeopleMeet" and "PeopleSplitUp" also obtain comparable results, indicating the fusion of DT and IDT is beneficial to the detection of long interaction events.

5.3.3. Discussion and visualization analysis

According to above experiments, we observe that CNN-based approach can surely capture representative postures in short-term events, while conventional hand-crafted features such as STIP, DT and IDT, cannot abstract the most informative part well. On the contrary, for long-duration events, the trajectory-based method can catch clear motion patterns so as to represent the interaction of individuals, while the deep model cannot achieve remarkable progresses. Meanwhile, DT and IDT are complementary, and their fusion can boost the performance.

Table 6: The comparison for long-duration events.

Event	DT + IDT		SED15[1][47]		SED14[1][46]		SED13[1][10]		Chen[9]	
	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}	A_{DCR}	M_{DCR}
CellToEar	1.004	1.004	1.370	1.370	0.992	0.991	1.000	1.000	1.000	1.000
PeopleMeet	0.893	0.893	1.042	1.042	0.858	0.858	0.949	0.945	1.036	0.949
PeopleSplitUp	0.893	0.893	0.938	0.938	0.835	0.833	0.894	0.887	0.843	0.788
PersonRun	0.576	0.576	0.970	0.970	0.830	0.830	0.770	0.764	0.834	0.787

[1] note: SED15 denotes the best results of other systems at SED'2015, and SED14 and SED13 stand for the best results at SED'2014 and SED'2013 evaluations.

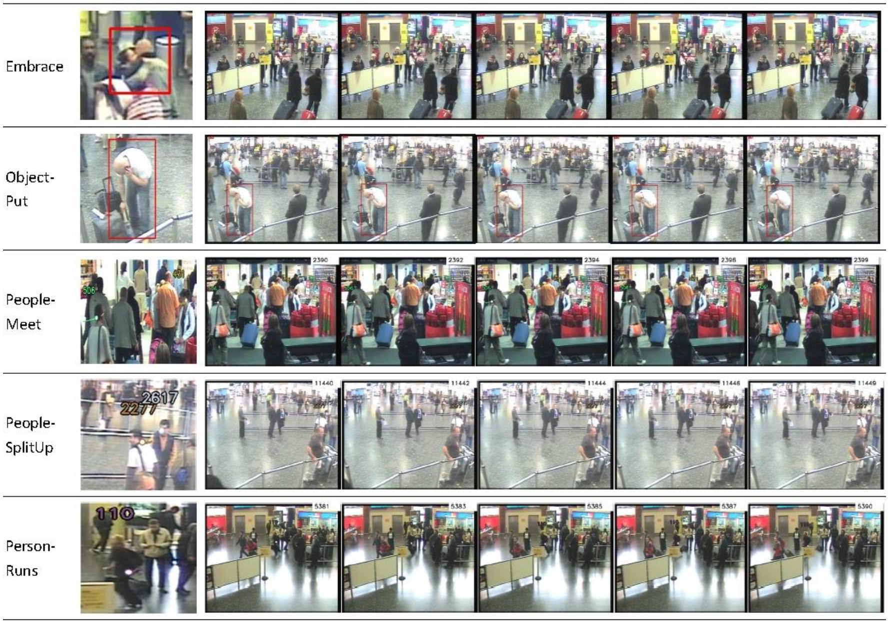


Figure 11: The detected instances of SED.

In order to reduce the impacts of the data imbalance, the data augmentation scheme can play a positive role. In addition, the cascade network architecture may be another feasible method, because it can quickly eliminate vast negative samples in the early training. To some extent, the cascade architecture alleviates the data imbalance and decreases the overfitting.

Fig. 11 visualizes several detected event sequences. The short-term events are localized with bounding boxes (marked by red rectangle) and the frame number, and long-duration events show trajectories (marked by color curve) of pedestrains (denoted by a white point with a number) and the frame number. The first image of each row is a closeup of the corresponding event. We observe that our method could localize events in complex surroundings such as heavy occlusion, crowded people stream etc. The detected events may rapidly provide important clues in dealing with the emergencies. However, some false and missing detections are still exist, especially for "CellToEar", containing a small object. We believe this kind of mistakes can be improved by appending more training data and introducing a deeper network architecture.

6. Conclusion

SED is an extremely challenging task due to many factors such as uncontrolled surrounding, multiple camera views, crowded people stream, the data imbalance and insufficient annotations etc. In order to address these problems, we firstly build a new large-scale pedestrian dataset (SED-PD) and propose a novel cascade CNN to detect pedestrains. Secondly, we present a unified SED framework to detect two kinds of events: short-term events and long-duration events. For the former, we regard the event detection as a classification problem of static key-poses, and adopt a CNN-based method. For the latter, we apply trajectory-based approach to represent the spatio-temporal context. The experimental results on SED-PD and TRECVID SED datasets show the effectiveness of proposed framework.

7. Acknowledgments

This work is supported by Chinese National Natural Science Foundation (61471049, 61372169, 61532018), and Key Laboratory of Forensic Marks, Ministry of Public Security of China.

References

[1] P. Over, G. Awad, M. Michel, J. Fiscus, W. Kraaij, A. F. Smeaton, G. Quenot, R. Ordelman, Trecvid 2015 – an overview of the goals, tasks, data, evaluation mechanisms and metrics, in: Proceedings of TRECVID 2015, 2015.

[2] R. T. Rose, J. G. Fiscus, P. Over, J. S. Garofolo, M. Michel, The trecvid 2008 event detection evaluation, in: IEEE Workshop on Applications of Computer Vision WACV 2009, 2009, pp. 1–8.

[3] C. Schüldt, I. Laptev, B. Caputo, Recognizing human actions: A local SVM approach, in: 17th International Conference on Pattern Recognition, ICPR 2004, 2004, pp. 32–36.

[4] M. Marszalek, I. Laptev, C. Schmid, Actions in context, in: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2009, pp. 2929–2936.

[5] K. Soomro, A. R. Zamir, M. Shah, UCF101: A dataset of 101 human actions classes from videos in the wild, CoRR abs/1212.0402.

[6] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, F. Li, Large-scale video classification with convolutional neural networks, in: Conference on Computer Vision and Pattern Recognition, CVPR 2014, 2014, pp. 1725–1732.

[7] C. Gao, D. Meng, W. Tong, Y. Yang, Y. Cai, H. Shen, G. Liu, S. Xu, A. G. Hauptmann, Interactive surveillance event detection through mid-level discriminative representation, in: International Conference on Multimedia Retrieval, ICMR'14, 2014.

[8] G. Zhu, M. Yang, K. Yu, W. Xu, Y. Gong, Detecting video events based on action recognition in complex scenes using spatio-temporal descriptor, in: Proceedings of Multimedia 2009, 2009, pp. 165–174.

[9] Q. Chen, Y. Cai, L. M. Brown, A. Datta, Q. Fan, R. S. Feris, S. Yan, A. G. Hauptmann, S. Pankanti, Spatio-temporal fisher vector coding for surveillance event detection, in: ACM Multimedia Conference, MM'13, 2013.

[10] X. Yang, ATT research at trecvid 2013: surveillance event detection, in: Proceedings of TRECVID 2013, 2013.

[11] W. Ouyang, X. Wang, Joint deep learning for pedestrian detection, in: IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013, 2013, pp. 2056–2063.

[12] P. Luo, Y. Tian, X. Wang, X. Tang, Switchable deep network for pedestrian detection, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, 2014, pp. 899–906.

[13] J. H. Hosang, M. Omran, R. Benenson, B. Schiele, Taking a deeper look at pedestrians, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, 2015, pp. 4073–4082.

[14] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, in: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, 2014, pp. 568–576.

[15] Z. Zhao, B. Zhao, F. Su, Person re-identification via integrating patch-based metric learning and local salience learning, International Journal of Pattern Recognitiondoi:https://doi.org/10.1016/j.patcog.2017.03.023.

[16] I. Laptev, On space-time interest points, International Journal of Computer Vision 64 (2-3) (2005) 107–123.

[17] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2005, 2005, pp. 886–893.

[18] I. Laptev, M. Marszalek, C. Schmid, B. Rozenfeld, Learning realistic human actions from movies, in: 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008, 2008.

[19] S. Hongeng, R. Nevatia, Large-scale event detection using semi-hidden markov models, in: 9th IEEE International Conference on Computer Vision (ICCV 2003), 2003, pp. 1455–1462.

[20] H. Meng, N. Bianchi-Berthouze, Y. Deng, J. Cheng, J. P. Cosmas, Time-delay neural network for continuous emotional dimension prediction from facial expression sequences, IEEE Trans. Cybernetics 46 (4) (2016) 916–929.

[21] Z. Ma, Z. Tan, J. Guo, Feature selection for neutral vector in EEG signal classification, Neurocomputing 174 (2016) 937–945.

[22] H. Meng, N. Bianchi-Berthouze, Affective state level recognition in naturalistic facial and vocal expressions, IEEE Trans. Cybernetics 44 (3) (2014) 315–328.

[23] B. Laxton, J. Lim, D. J. Kriegman, Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video, in: 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2007, 2007.

[24] Z. Ma, A. Leijon, Bayesian estimation of beta mixture models with variational inference, IEEE Trans. Pattern Anal. Mach. Intell. 33 (11) (2011) 2160–2173.

[25] Z. Ma, A. E. Teschendorff, A. Leijon, Y. Qiao, H. Zhang, J. Guo, Variational bayesian matrix factorization for bounded support data, IEEE Trans. Pattern Anal. Mach. Intell. 37 (4) (2015) 876–889.

[26] P. Xu, Q. Yin, Y. Huang, Y.-Z. Song, Z. Ma, L. Wang, T. Xiang, W. B. Kleijn, J. Guo, Cross-modal subspace learning for fine-grained sketch-based image retrieval, NEUROCOMPUTING arXiv: 1705.09888.

[27] Z. Ma, P. K. Rana, J. Taghia, M. Flierl, A. Leijon, Bayesian estimation of dirichlet mixture model with variational inference, Pattern Recognition 47 (9) (2014) 3143–3157.

[28] F. Perronnin, C. R. Dance, Fisher kernels on visual vocabularies for image categorization, in: 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2007, 2007.

[29] H. Wang, A. Kläser, C. Schmid, C. Liu, Action recognition by dense trajectories, in: The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, 2011, pp. 3169–3176.

[30] H. Wang, C. Schmid, Action recognition with improved trajectories, in: IEEE International Conference on Computer Vision, ICCV 2013, 2013, pp. 3551–3558.

[31] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing System(NIPS), 2012, pp. 1106–1114.

[32] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, M. Paluri, C3D: generic features for video analysis, CoRR abs/1412.0767.

[33] J. Y. Ng, M. J. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, G. Toderici, Beyond short snippets: Deep networks for video classification, CoRR abs/1503.08909.

[34] Z. Zhao, Y. Song, F. Su, Specific video identification via joint learning of latent semantic concept, scene and temporal structure, Neurocomputing 208 (2016) 378–386.

[35] J. Doshi, L. Brown, Y. Cheng, Ibm-georgia tech@trecvid 2015: surveillance event detection, in: Proceedings of TRECVID 2015, 2015.

[36] P. F. Felzenszwalb, D. A. McAllester, D. Ramanan, A discriminatively trained, multiscale, deformable part model, in: 2008 IEEE Computer Soci-

ety Conference on Computer Vision and Pattern Recognition CVPR 2008, 2008.

[37] Q. Chen, W. Jiang, Y. Zhao, Z. Zhao, Part-based deep network for pedestrian detection in surveillance videos, in: 2015 Visual Communications and Image Processing, VCIP 2015, 2015, pp. 1–4.

[38] P. Dollár, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: A benchmark, in: Proceedings of the Computer Vision and Pattern Recognition, 2009, pp. 304–311.

[39] Y. Tian, P. Luo, X. Wang, X. Tang, Pedestrian detection aided by deep learning semantic tasks, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, 2015, pp. 5079–5087.

[40] Online:, `https://code.google.com/p/cuda-convnet/`.

[41] B. Xu, W. Liao, Z. Liu, W. Bao, Y. Li, D. Yang, Z. Chen, IIPWHU at trecvid 2015, in: Proceedings of TRECVID 2015, 2015.

[42] X. Peng, L. Wang, X. Wang, Y. Qiao, Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice, CoRR abs/1405.4506.

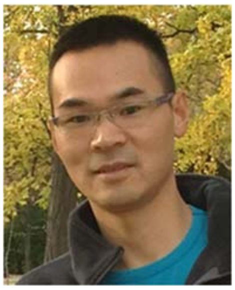
[43] R. Fan, K. Chang, C. Hsieh, X. Wang, C. Lin, LIBLINEAR: A library for large linear classification, Journal of Machine Learning Research 9 (2008) 1871–1874.

[44] H. Lin, C. Lin, R. C. Weng, A note on platt's probabilistic outputs for support vector machines, Machine Learning 68 (3) (2007) 267–276.

[45] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the ACM International Conference on Multimedia, MM'14, 2014, pp. 675–678.

[46] X. Du, Y. Cai, Y. Zhao, H. Li, Y. Yang, A. Hauptmann, Informedia@trecvid 2014:surveillance event detection, in: Proceedings of TRECVID, 2014.

[47] Y. Su, A. Liu, Z. Gao, W. Nie, TJU-TJUT at trecvid 2015:surveillance event detection, in: Proceedings of TRECVID, 2015.



Zhicheng Zhao is an associate professor of Beijing University of Posts and Telecommunications. He was a visiting scholar at School of Computer Science, Carnegie Mellon University. His research interests are computer vision, image and video semantic understanding and retrieval. He has authored and coauthored more than 50 journal and conference papers.

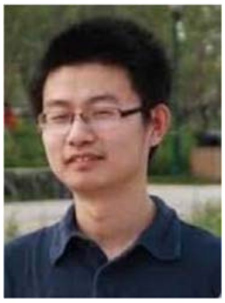
Xuanchong Li received the B.S. degree from Zhijiang University, Hangzhou, China, in 2012. He is currently and working toward the M.S. degree in School of Computer Science, Carnegie Mellon University, under the supervision of Dr. Alexander G. Hauptmann. His research interests includes machine learning, computer vision, and multimedia analysis.

Xingzhong Du received the B.S. and M.S. degree from Nanjing University, Nanjing, China, in 2010 and 2013 respectively. He is currently a Ph.D candidate at the Data and Knowledge Engineering (DKE) Group in the School of Information Technology and Electrical Engineering, University of Queensland under Prof. Xiaofang Zhou and Dr. Yi Yang supervision. His research interests contains machine learning, computer vision and video database.

Qi Chen received the M.S. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2016. He is currently a researcher with Beijing Bytedance Technology Co. Ltd. His research interests include machine learning and computer vision.

Yanyun Zhao is a female associate professor in the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. She received the Ph.D. degree from Beijing University of Posts and Telecommunications in 2009. Her research interests include pattern recognition, image and video processing. She has authored and coauthored more than 60 journal and conference papers and some textbooks.

Fei Su is a female professor of Beijing University of Posts and Telecommunications. She received the Ph.D. degree majoring in Communication and Electrical Systems from BUPT in 2000. She was a visiting scholar at Carnegie Mellon University from 2008 to 2009. Her current interests include pattern recognition, image and video processing and biometrics. She has authored and coauthored more than 70 journal and conference papers and some textbooks.



Xiaojun Chang is a Ph.D. student at University of Technology Sydney, under the supervision of Dr. Yi Yang. His research interests include machine learning, data mining and computer vision. His publications appear in proceedings of prestigious international conference like CVPR, ICML, AAAI, IJCAI and etc.

Alexander G. Hauptmann received the B.A. and M.A. degrees in psychology from Johns Hopkins University, Baltimore, MD, the degree in computer science from the Technische Universitt Berlin, Berlin, Germany, in 1984, and the Ph.D. degree in computer science from Carnegie Mellon University (CMU), Pittsburgh, PA, in 1991. He is currently with the faculty of the School of Computer Science, CMU. His research interests include several different areas: man-machine communication, natural language processing, speech understanding and synthesis, video analysis, and machine learning. From 1984 to 1994, he worked on speech and machine translation, when he joined the Informedia project for digital video analysis and retrieval, and led the development and evaluation of news-on-demand applications.