2010 Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance

Incremental Learning Approach for Events Detection from large Video dataset

Ali WALI, Adel M. ALIMI
REGIM: REsearch Group on Intelligent Machines, University of Sfax,
National School of Engineers (ENIS), BP 1173, Sfax, 3038, Tunisia
ali.wali@ieee.org, adel.alimi@ieee.org

Abstract

In this paper, we propose a strategy of multi-SVM incremental learning system based on Learn++ classifier for detection of predefined events in the video. This strategy is offline and fast in the sense that any new class of event can be learned by the system from very few examples. The extraction and synthesis of suitably video events are used for this purpose. The results showed that the performance of our system is improving gradually and progressively as we increase the number of such learning for each event. We then demonstrate the usefulness of the toolbox in the context of feature extraction, concepts/events learning and detection in large collection of video surveillance dataset.

1. Introduction

The proliferation of cameras in video surveillance systems creates a flood of information increasingly important, which is difficult to analyze with standard tools. If the image processing techniques currently used to detect abnormal events (such as cars against the sense objects abandoned), the search video data in a post, for example to find events of interest, represents a huge task if it must be done manually. Therefore, it is necessary to develop solutions in order to aid the search in a sequence of video surveillance. In this type of sequence, the user does not scenes or visuals, as in the case of excavation in movies or video archive, but rather events. The main use cases of search in data from video surveillance are:

- Detection and search of special events.
- To lead and / or optimize the methods of classification and automatic detection of abnormal events.
- For the management of infrastructure, for example; roads, access to shopping malls and public spaces.

Work on the annotation and retrieval of video data is very numerous. In our case, only the particular case of video surveillance will be addressed. This presents some specific characteristics that include:

- Using a fixed camera. Then, there is a background image that can be separate objects of interest easily.
- We can observe different types of visual objects (person, package, car, etc..) and extract their characteristics (appearance / disappearance, position, direction, speed) The semantic description is relatively small even for specific applications (detection of cars, people tracking, detection of abandoned packages).

We can also distinguish the methods involved in detecting rare events (abandoned packages, etc...). Methods performing a detection of current event (counting cars, etc.). Many events can be represented as object activities and interactions (such as Walking and Airplane Flying), and show different motion patterns. Motion is thus an important cue in describing the course of an event, and has been employed in some previous works in order to capture the event evolution information. In [3], a new integrated robot vision system designed for multiple human tracking and silhouette extraction using an active stereo camera. A fast histogram based tracking algorithm is presented by using the mean shift principle. In [4] a hybrid method that combines HMM with SVM to detect semantic events in video is proposed. The proposed detection method has some advantages that it is suitable to the temporal structure of event thanks to Hidden Markov Models (HMM) and guarantees high classification accuracy thanks to Support Vector Machines (SVM). The performance of this method is compared with that of HMM based method, which shows the performance increase in both recall and precision of semantic event detection.

We propose in this paper a technique for detection of events based on a generic learning system called M-SVM (Multi-SVM). The objective is to enable the detector to cause various types of events by presenting an incremental way a significant number of examples and for the sake of genericity. Examples of application of this technique are the intelligent video surveillance of airports and shopping

978-0-7695-4264-5/10 $26.00 © 2010 IEEE
DOI 10.1109/AVSS.2010.54

malls for the automatic detection of events such as the disappearance of objects or detecting events that may constitute a threat to report them to an operator human (person leaving a package for example).

2. Overview of our system

We propose in Figure 1 our system of detection and recognition event in the video. This system allows describing the main treatments that can be induced to make and their goals. There are two steps for the recognition event: the first is to learn the event description from a database of examples (learning), the second will detect and recognize an event from his description extracted from key frames (classifications).

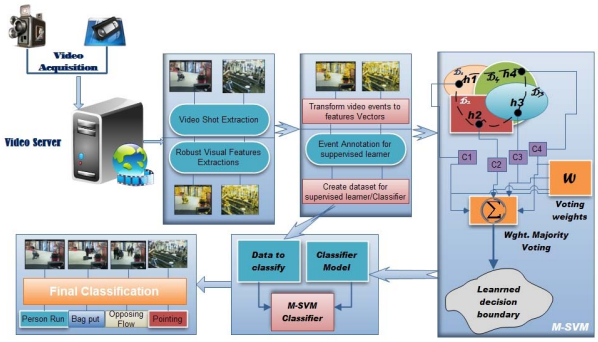


Figure 1. Global Overview of our System (I-VidEvDetect)

Learning Phase

Event Description: The purpose of this step is to obtain a data representation which is then used for learning, *Learning:* from a set of copies, we construct a representation of classes.

Recognition Phase

Pre-processing: noise removal, normalization, resampling, contrast enhancement, etc. ...

Segmentation: extracting interest areas of images (contour approach or region)

Description of events: the extraction of data representation compatible with the decision tools used, *Assignment:* Assigning an event unknown to a class (with the possibility of having an index of confidence) *Post-treatment:* validation of analysis decisions on the basis of knowledge. In our system we target six class of event. We divide this list of events into two categories:

Collaborative events:

- Embrace
- People Split Up

Individual events:

- Elevator No Entry
- Object Put
- Person Runs
- Opposing Flow

Figure 2. Elevator No entry event.

Person Run

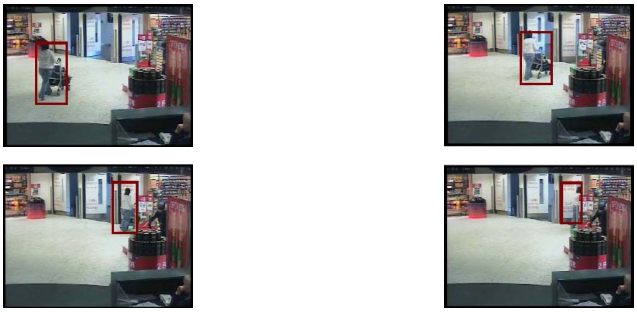
Embrace

Figure 3. Person Run and Embrace events.

3. Supported Visual Feature Extraction

We use a set of different visual descriptors at various granularities for each frame, rid of the static background and the moving objects, of the video shots. The relative performance of the specific features within a given feature modality is shown to be consistent across all events. However, the relative importance of one feature modality vs. another may change from one event to the other. In addition to the descriptors that are described in a recent paper [9], we use the following descriptors:

- The moments of Hu: These moments are invariant under translation and scaling. The moments of Hu [5] are calculated from standardized moments and are invariant under translation, rotation and scaling. These moments are number 7.



Opposing Flow

People Split Up

Figure 4. Opposing Flow and People Split Up events.

- Zernike moment [2]: Zernike moment has many invariant characteristics, including translation invariance, rotation invariance and scale invariance. We use the 10-order Zernike moment, there are 36 elements in the image Zernike moments shape feature vector and one 37-dimensional vector is used to represent it.

- Motion Activity: The descriptors that we use are corresponded to the energy calculated on every sub-band, by decomposition in wavelet of the optical flow estimated between every image of the sequence. We use two optical flow methods the first is Horn-Schunck and the second is the Lucas-Kanade method described in [1]. We obtain two vectors of 10 bins everyone; they represent for every image a measure of activity sensitive to the amplitude, the scale and the orientation of the movements in the shot.

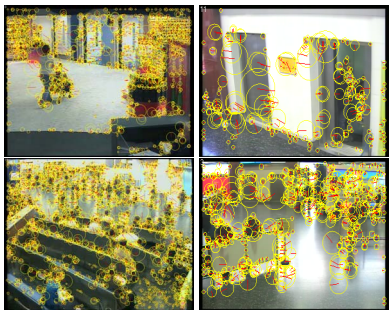


Figure 5. SIFT Features Example.

4. Combining single SVM classifier for learning video event

Support Vector Machines (SVMs) have been applied successfully to solve many problems of classification and regression. However, SVMs suffer from a phenomenon called 'catastrophic forgetting', which involves loss of information learned in the presence of new training data.
Learn++ [6] has recently been introduced as an incremental learning algorithm. The strength of Learn++ is its ability to learn new data without forgetting prior knowledge and without requiring access to any data already seen, even if new data introduce new classes. To benefit from the speed of SVMs and the ability of incremental learning of Learn++, we propose to use a set of trained classifiers with SVMs based on Learn++ inspired from [10].
Experimental results of detection of events suggest that the proposed combination is promising. According to the data, the performance of SVMs is similar or even superior to that of a neural network or a Gaussian mixture model.

4.1. SVM Classifier

Support Vector Machines (SVMs) are a set of supervised learning techniques to solve problems of discrimination and regression. The SVM is a generalization of linear classifiers. The SVMs have been applied to many fields (bio-informatics, information retrieval, computer vision, finance ...). According to the data, the performance of SVMs is similar or even superior to that of a neural network or a Gaussian mixture model. They directly implement the principle of structural risk minimization [8] and work by mapping the training points into a high dimensional feature space, where a separating hyperplane (w, b) is found by maximizing the distance from the closest data points (boundary-optimization). Given a set of training samples $S = \{(x_i, y_i)|i = 1, .., m\}$, where $x_i \in R_n$ are input patterns, $y_i \in +1, -1$ are class labels for a 2-class problem, SVMs attempt to find a classifier $h(x)$, which minimizes the expected misclassification rate. A linear classifier $h(x)$ is a hyperplane, and can be represented as $h(x) = sign(w^T x + b)$. The optimal SVM classifier can then be found by solving a convex quadratic optimization problem:

$$\begin{aligned} &\underbrace{max}_{w,b} \tfrac{1}{2} \|w\|^2 + C \textstyle\sum_{i=1}^{m} \xi_i \; subject\; to \\ &y_i \left(\langle w, x_i \rangle + b\right) \geq 1 - \xi_i \; and \; \xi_i \geq 0 \end{aligned} \tag{1}$$

Where b is the bias, w is weight vector, and C is the regularization parameter, used to balance the classifier's complexity and classification accuracy on the training set S. Simply replacing the involved vector inner-product with a non-linear kernel function converts linear SVM into a more

flexible non-linear classifier, which is the essence of the famous kernel trick. In this case, the quadratic problem is generally solved through its dual formulation:

$$L(w,b,\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\left(\sum_{i=1}^{m} y_i y_j \alpha_i \alpha_j K(x_i, x_j)\right)$$
$$subject\ to\ C \geq \alpha_i \geq 0\ and\ \sum_{i=1}^{m} y_i \alpha_i y_i = 0 \tag{2}$$

where a_i are the coefficients that are maximized by Lagrangian. For training samples x_i, for which the functional margin is one (and hence lie closest to the hyperplane), $\alpha_i \succ 0$. Only these instances are involved in the weight vector, and hence are called the support vectors [7]. The non-linear SVM classification function (optimum separating hyperplane) is then formulated in terms of these kernels as:

$$h(x) = sign\left(\sum_{i=1}^{m} \alpha_i y_i K(x_i, x_j) - b\right) \tag{3}$$

4.2. M-SVM Classifiers

M-SVM is based on Learn++ algorithm. This latter, generates a number of weak classifiers from a data set with known label. Depending on the errors of the classifier generated low, the algorithm modifies the distribution of elements in the subset according to strengthen the presence of the most difficult to classify. This procedure is then repeated with a different set of data from the same dataset and new classifiers are generated. By combining their outputs according to the scheme of majority voting Littlestone we obtain the final classification rule.

The weak classifiers are classifiers that provide a rough estimate - about 50% or more correct classification - a rule of decision because they must be very quick to generate. A strong classifier from the majority of his time training to refine his decision criteria.

Finding a weak classifier is not a trivial problem and the complexity of the task increases with the number of different classes, however, the use of NN algorithms can correctly resolved effectively circumvent the problem. The error is calculated by the equation:

$$error_t = \sum_{i:h_i(x_i)\neq y_i} S_t(i)\,[|h_t(x_i) \neq y_i|] \tag{4}$$

with $h - t : X \rightarrow Y$ an hypothesis and where TR_t is the subset of training subset and the TE_t is the test subset. The synaptic coefficients are updated using the following equation:

$$w_{t+1}(i) = w_t(i) * \left\{ \begin{array}{l} \beta_t\ if\ H_t(x_i) = y_i \\ 1\ else \end{array} \right\} \tag{5}$$

Where t is the iteration number, B_t composite error and standard composite hypothesis H_t.

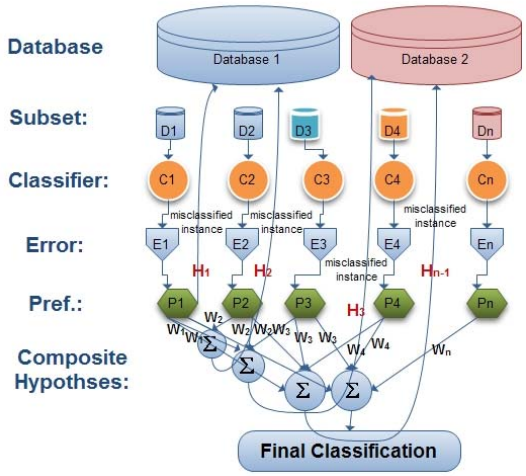


Figure 6. M-SVM classifier

Finally, M-SVM classifier (figure 6) is based on the following intuition: The ensemble is obtained by retraining a single SVM using strategically updated distribution of the training dataset, which ensures that examples that are misclassified by the current ensemble have a high probability of being resampled. The examples that have a high probability of error are precisely those that are unknown or that have not yet been used to train the previous classifiers. Distribution update rule is optimized for incremental learning of new data.

In our approach we replace each weak classifier by SVM. After T_k classifiers are generated for each D_k, the final ensemble of SVMs is obtained by the weighted majority of all composite SVMs:

$$H_{final}(x) = arg \underbrace{max}_{y\in Y} \sum_{k=1}^{K} \sum_{t:h_t(x)=y} log\frac{1}{\beta_t} \tag{6}$$

5. Experimental Results

To assess the performance of our algorithm, we use a part of TRECVID'2009 video data for event detection task. The database consist of Airport surveillance video. We targeted six events: 'Person Runs', 'People Split Up', 'Object Put', 'Embrace', 'Elevator No Entry' and 'Opposing Flow' (see figure 2, 3 and 4). The video dataset are divided into five sets one set for every event. Every dataset is divided into 4 sets, 3 sets to learning process and one set to classifying process.

5.1. System Setup

All features described in 3 are extracted from every keyframe of every shot.

Table 1. Event detection Results (Run 1: with 20 learning Samples, Run 2: with 30 learning Samples)

	Run 1		Run 2	
Event	**A.NDCR**	**M.NDCR**	**A.NDCR**	**M.NDCR**
Embrace	1.143	0.991	1.121	0.971
PeopleSplitUp	3.522	0.993	3.371	0.963
ElevatorNoEntry	0.361	0.342	0.321	0.312
ObjectPut	1.376	0.999	1.154	0.972
PersonRuns	1.177	0.986	1.116	0.954
OpposingFlow	1.142	1.124	0.999	0.943

One against all M-SVM model for each event of each camera view:
120 models are built (6 actions * 5 camera views): 2 models per action and per camera views. The first one is obtained by the global-descriptor-based events detection and identifier subsystem of I-VidEvDetect. The second model is obtained by Rich-descriptor-based events detection subsystem of I-VidEvDetect (figure 7).

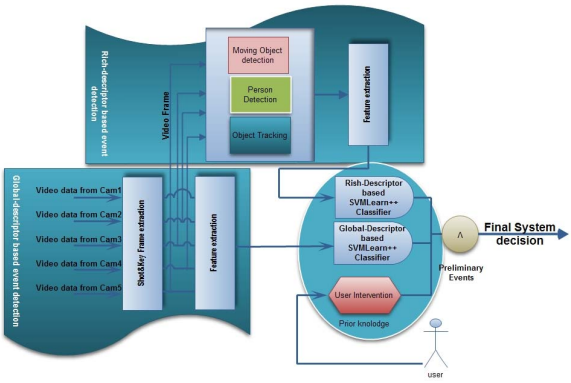


Figure 7. Intelligent-Video Event Detection (I-VidEvDetect)

5.2. Performance evaluation of our system

To evaluate the performance of our system we use the TRECVID'2009 event detection metrics. The evaluation uses the Normalized Detection Cost Rate (NDCR).
NDCR is a weighted linear combination of the system's Missed Detection Probability and False Alarm Rate (measured per unit time). The measure's derivation can be found in $(http://www.itl.nist.gov/iad/mig/tests/trecvid/2009/doc/EventDet09-EvalPlan-v03.htm)$ and the final formula is summarized below. Two versions of the NDCR will be calculated for the system: the Actual NDCR and the Minimum NDCR.
The actual and minimum NDCRs for each of the events can be seen in Table 1. We have achieved very competitive minimum DCR results on the events of embrace, people Split UP, Object Put, opposing Flow and especially for Elevator No Entry. We did not extensively tune parameters with the aim of producing low actual DCR score; our actual DCR looks relatively higher (the lower the score, the better the performance). But our system achieved very good minimum DCR scores.

6. Conclusion

Event detection in surveillance video becomes a hot research topic of multimedia content analysis nowadays. In this paper, we have presented a strategy of incremental learning system for detection of predefined events in the video surveillance dataset. The results obtained so far are interesting and promoters. The advantage of this approach its allows human operators to use context-based queries and the response to these queries is much faster. We have recently focused on detecting and tracking people in image sequences to improve the accuracy of our system.

7. Acknowledgement

The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

[1] Andrs B. and Joachim W. Lucas/kanade meets horn/schunck. *International Journal of Computer Vision*, 61(3):211–231, 2005.

[2] Khotanzad A.; Hong Y. H. Invariant image recognition by zernike moments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12:489–497, 1990.

[3] Jung-Ho A.; Sooyeong K. and al. An integrated robot vision system for multiple human tracking and silhouette extraction. *LNCS Advances in Artificial Reality and Tele-Existence*, 4282:113–122, 2006.

[4] Tae Meon B.; Cheon Seog K. and al. Semantic event detection in structured video using hybrid hmm/svm. *LNCS Image and Video Retrieval*, 3568:113–122, 2005.

[5] Sivaramakrishna R.; Shashidharf N.S. Huapos;s moment invariants: how invariant are they under skew andperspective transformations? *WESCANEX 97: Communications, Power and Computing*, 22–23:292–295, 1997.

[6] Polikar R.; Udpa L.; Udpa S. S. and Honavar V. Learn++: An incremental learning algorithm for supervised neural networks. *IEEE Trans. Sys. Man, Cybernetics (C)*, 31(4):497–508, 2001.

[7] N. Cristianini ; J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[8] V. Vapnik. *Statistical Learning Theory*. 1998.

[9] Alimi A. M. Wali A. Event detection from video surveillance data based on optical flow histogram and high-level feature extraction. *IEEE DEXA Workshops 2009*, pages 221–225, 2009.

[10] Robi P. Zeki E. and al. Ensemble of svms for incremental learning. *LNCS MCS*, 3541:246–256, 2005.