
Using decision trees to build an event recognition framework for automated visual surveillance

Cedric Simon

Christophe De Vleeschouwer

Communication and Remote Sensing Lab, UCL, Louvain-La-Neuve, Belgium

Jerome Meessen

Multitel, Mons, Belgium

CEDRIC.SIMON@UCLouvain.be

DEVLEESCHOUWER@UCLouvain.be

JEROME.MEESSEN@MULTITEL.be

Abstract

This paper presents a classifier-based approach to recognize possibly sophisticated events in video surveillance. The aim of this work is to propose a flexible and generic event recognition system that can be used in a real world context. Our system uses the ensemble of randomized trees procedure to model each event as a sequence of structured activity patterns, without using any tracking method. Experimental results demonstrate the robustness of the system toward artifacts and passer-by, and the effectiveness of its framework for event recognition applications in visual surveillance.

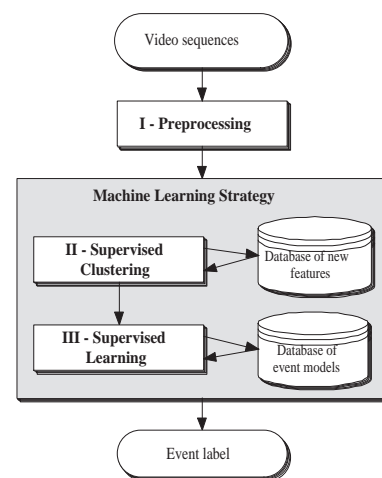


Figure 1. Overview of the system

1. Overview

The growing number of cameras in public and private areas increases the interest of the image processing community in automated visual surveillance system. Nonetheless, there is still a broad gap between what automated systems offer and the actual needs of the industry (Dee & Velastin, 2007). The system we propose aims at reducing this gap, by using a coherent framework that links local and coarse features with meaningful concepts. Those local features are attributes representing the moving objects (the blobs) in the scene, at each frame. No tracking procedure nor intermediate reasoning (eg. occlusions) is needed which leads in a greater genericity. The framework we propose (figure 1) relies on two main parts that are both based on the ensemble of randomized trees concept (Geurts et al., 2006).

In the first stage (section 2), sophisticated features are built by clustering the blobs according to their feature's values at each frame. Those clusters are tagged,

and each tag is then associated to the appropriate frames in the video sequences. In the second stage (section 3), the events are classified, based on the temporal distribution of those tags in the sequences. The main idea is to discriminate event classes by investigating the temporal relationships between the tags, for example by asking if there is a blob of one type before a blob of another type. This framework is inspired from Geman works in (Amit & Geman, 1997).

2. Construction of elaborated features and definition of tags

From the blob's features, we adopt an information-theoretic approach and cluster similar blobs by using the decision tree methodology. Two sets of *clustering trees* are built according to the type of attributes used by the trees:

- In the first pool, attributes are based on blob’s features (i.e its position, size and velocity)
- In the second pool, attributes are based on pair of blob’s features (i.e the distance between the blobs and their relative velocity). For efficiency, from all pairs of blobs in each frame, we keep only pairs having a relatively short distance between the blobs

At each node of one tree, a question is selected by choosing one attribute, i.e by picking one feature and one threshold that reduce as much as possible the class entropy within the node. Thereby, each blob (or pair of blobs) inherit the event class of the whole sequence it belongs to.

Once the trees are built, we tag each node of the trees, except for the root node. Hence, by definition, each tag correspond to a specific combination of coarse and local features, that characterize each frame of the video sequences. Diversity of the tags can then be boosted by increasing the number of the clustering trees N_{CT} , while more specificity is obtained by increasing the depth of the trees. For instance, we observed in our experiments that the system achieved the best performance for $N_{CT} \approx 10$. This supervised clustering process allows each tag to be more discriminant regarding the event classes, while using only local information (each frame individually).

3. Modeling the spatio-temporal events with randomized trees

Once the elaborated features are computed, we use another set of randomized trees to model and classify the video sequences, based on the spatio-temporal arrangement of those advanced features. This is done by defining the two output branches of a tree node based on the presence/absence of a specific temporal arrangement A of tags in the video sequence. In order for these arrangements to be scale invariant (and better fit the notion of visual surveillance event), coarse binary relations like ”before”, ”after” and ”at the same time” are used. An example of arrangement would then be: tag x exists ”before” tag y which exists ”at the same time” as tag z.

At the root node N_r a tag T_x from the full set \mathcal{T} of tags is chosen. The Question (Q_0) is ”Does T_x exist in the training sequences?”. Those training samples for which $Q_{N_r} = 0$ are in the ”no” child node and we search again through T . Those samples for which $Q_{N_r} = 1$ are in the ”yes” child node and we then put T_x among the participating tags \mathcal{T}_p . Further question

can be:

$$\begin{cases} \exists T_i? \\ \exists T_i \star T_j? \\ \exists T_{j_x} \star T_{j_y}? \end{cases} \quad \text{with} \quad \begin{cases} T_i \in \mathcal{T} \\ T_j, T_{j_x}, T_{j_y} \in \mathcal{T}_p \\ \star \in \{<, |, >\} \end{cases} \quad (1)$$

with $<$ meaning ’before’, $|$ ’during’ and $>$ ’after’.

4. Experiments and perspective

The described system was tested on two scenarios. The first one used simulated video surveillance events, while the second is a real case scenario, which occurs in the entrance lobby of a public company. More information about those datasets can be found on the web ¹. Results are encouraging our framework, even if passer-by or by-stander are significantly decreasing the accuracy.

Further improvement could then analyze how to use histograms of similarity between the blobs to enrich our elaborated features and gain some robustness, along with features characterizing each frame more globally (i.e. the number of blobs in the frame, the density of the moving objects in each frame, the average/variance of each features...). We will also focus on using an active learning procedure in order to interact with the user, and to be able to enhance the system on line.

References

- Amit, Y., & Geman, D. (1997). Shape quantization and recognition with randomized trees. *Neural Computation*, 9, 1545–1588.
- Dee, H., & Velastin, S. (2007). How close are we to solving the problem of automated visual surveillance? a review of real-world surveillance, scientific progress and evaluative mechanisms. *Machine Vision and Applications, Special Issue on Video Surveillance Research in Industry and Academic Springer*.
- Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, 63, 3–42.

¹<http://www.tele.ucl.ac.be/~csimon/trajectories.html>.