Recognition of Long-Term Behaviors by Parsing Sequences of Short-Term Actions with a Stochastic Regular Grammar

Gerard Sanromà, Gertjan Burghouts, and Klamer Schutte

TNO, Oude Waalsdorperweg 63, 2597 AK The Hague, The Netherlands
gerard.sanromaguell@tno.nl

Abstract. Human behavior understanding from visual data has applications such as threat recognition. A lot of approaches are restricted to limited time actions, which we call *short-term actions*. Long-term behaviors are sequences of short-term actions that are more extended in time. Our hypothesis is that they usually present some structure that can be exploited to improve recognition of short-term actions. We present an approach to model long-term behaviors using a syntactic approach. Behaviors to be recognized are hand-crafted into the model in the form of grammar rules. This is useful for cases when few (or no) training data is available such as in threat recognition. We use a stochastic parser so we handle noisy inputs. The proposed method succeeds in recognizing a set of predefined long-term interactions in the CAVIAR dataset. Additionally, we show how imposing prior knowledge about the structure of the long-term behavior improves the recognition of short-term actions with respect to standard statistical approaches.

Keywords: long-term behavior, stochastic context-free grammars, human activity analysis, visual surveillance.

1 Introduction

Automated recognition of long-term behavior is relevant for many applications, where particular events have to be signaled. As examples: theft of truck cargo; dwelling of people in elderly homes; shopping behavior inside a mall.

Where short-term action recognition has received much attention [5,2,3], automated recognition of long-term behavior has been studied less. Long-term behavior is an interesting research topic, as it requires temporal modeling of sequences of short term actions. In this paper, we consider long-term behavior as a sequence of short-term actions. As contribution, we will provide a method to improve the recognition of such sequences by a parsing mechanism.

A complicating factor for recognition of behaviors, is that the potential number of temporally ordered combinations of actions is very high. One way to deal with this is to learn the limited set of likely combinations. The learning of temporal sequences has been studied intensively in the past, for instance, by a HMM [3] or by the related discriminative CRF [4]. They have both shown

G.L. Gimel' farb et al. (Eds.): SSPR & SPR 2012, LNCS 7626, pp. 225–233, 2012.

their merit for solving various problems, for a comparison see [9]. A problem with these methods is that they are known to require a large training set to learn the sequences. For the applications that we envisage in this paper, like the prevention of an unwanted or even hazardous situation, typically only very few positive examples are available. This makes the HMM and CRF intractable. Often world-knowledge is available on how situations evolve. The goal of this paper is to exploit such prior knowledge explicitly for the recognition of long-term behaviors. We consider an alternative modeling of sequences that requires few learning examples by including world-knowledge by means of a hand-crafted rule set which is enforced using a stochastic grammar.

Grammars enable the encoding of sequences by simple expressions that limit the possibilities and capture the world-knowledge about how long-term behaviors evolve [8,6]. An example of such a sequence is *browse*, consisting of the actions: *walk standing still look around walk* etc. Such sequences are perfectly suited to be specified by a grammar. For the recognition of behaviors grammars have been studied previously. Our starting point is that we have only few learning examples, so the learning of grammar rules is out of this papers scope [10]. Grammars have mostly been used as a second-stage recognizer of situations or long-term behaviors, based on first-stage detectors of the constituent short-term actions [10,8]. We will follow the same strategy in this paper. In [8], context-free grammars were considered. A disadvantage of context-free grammars is that they cannot be directly interpreted as finite-state machines (FSM). Often, a FSM is the means by which expert knowledge is encoded, because it is an easy tool to model and understand. Instead, we will use a regular grammar which is built on top of an FSM, thus allowing to model the expert knowledge as a state diagram.

Our contributions in this paper are two-fold. First, we propose a regular grammar that exploits world-knowledge that is encoded by a FSM. To demonstrate the power of this grammar, we show it on the publicly available CAVIAR dataset that includes videos of realistic long-term behaviors. CAVIAR defines the included behaviors in terms of FSMs, which we will integrate into the grammar. Second, we show experimentally that long-term interactions can be recognized by the proposed grammar, and thereby the recognition of the constituent short-term actions is improved.

In section 2 we introduce the method used for recognition of short-term actions which is based on [2]. In section 3 our method for behavior recognition based on stochastic parsing is presented. In section 4 experimental validation is presented and some discussion is given. Finally, we conclude in section 5.

2 Short-Term Action Detection

As the basic observations, we are interested in recognizing a vocabulary of short-term interactions between two people from a set $\mathcal{A} = \{action_1, \ldots, action_L\}$ We use the non-parametric approach by [2]. This approach uses trajectory information from a set of previously extracted tracks by some standard method. So, for each clip we have a set of N tracks X^i, $i = 1, \ldots, N$ each one corresponding to

the trajectory followed by one person. This is, for i-th track, $X^i = \left\{\mathbf{x}_t^i,\, t \in T^i\right\}$, where $T^i = \left[t_1^i, \ldots, t_m^i\right]$ is the index-set of the frame interval that track X^i is present on the scene.

For each pair of persons (i, j) at each time t we compute the following feature vector.

$$\mathbf{f}_t^{(ij)} = \left[s_i^t, s_j^t, a_{ij}^t, d_{ij}^t, d_{ij}^{dif}, s_{ij}^{dif}\right] \tag{1}$$

where s_i^t is the distance covered by person i in between frames $t - w$ and t, a_{ij}^t is the alignment between persons, d_{ij}^t is the distance between two persons, d_{ij}^{dif} is the difference of distances w frames apart, and s_{ij}^{dif} is the difference in velocities (check [2] for more details).

We segment the clips into windows of *ws* frames with a certain overlap controlled by the offset *wo*. Therefore, for each window k we obtain the following set of feature vectors

$$F_k^{(ij)} = \left\{\mathbf{f}_t^{(ij)},\, t \in \mathcal{T}^k\right\} \tag{2}$$

where $\mathcal{T}^k = \left[\tau_a^k, \ldots, \tau_b^k\right]$ contains the indices of the frames belonging to the k-th window.

The goal of this section is to compute the probability $P\left(F^{test} | action_p\right)$ of a test window given the action $action_p \in \mathcal{A}$. Ground truth action labels $l_t^{(ij)} \in \mathcal{A}$ are attached to each feature vector $\mathbf{f}_t^{ij}$ in the training set. We define a label indicator function that returns the prevalence of an action inside a window (normalized to sum up to one). This is,

$$L\left(F_k^{(ij)};\, action_p\right) = \#\left\{l_t^{(ij)} \,|\, l_t^{(ij)} = action_p \wedge t \in \mathcal{T}^k\right\} \Big/ \left|F_k^{(ij)}\right| \tag{3}$$

where $\#\{\bullet\}$ corresponds to the cardinality of a set and $|F_k^{(ij)}|$ is the amount of vectors in the window.

In order to remove noise, we compute the PCA projections of the feature vectors $\tilde{\mathbf{f}}_t^{(ij)}$, $\forall i, j, t$ in the training set so as to retain an 80% of the total variance, obtaining also the projected windows $\tilde{F}_k^{(ij)}$. Given a test window of projected features $\tilde{F}^{test}$, we define the probability of being produced by a certain $action_p$ in the following way:

$$P\left(\tilde{F}^{test} | action_p\right) = \text{K-nn}\left(\tilde{F}^{test}, action_p\right) \Big/ K \quad , \tag{4}$$

where the function K-nn$(\tilde{F}, action_p)$ accumulates the prevalence of the action $action_p$ over the K nearest windows of $\tilde{F}$ in the training set. As distance measures to do the sort we use two variants, namely, the originally used Hausdorff distance [2] and, as an alternative, the Earth Mover's Distance (EMD) [11].

3 Behavior Recognition by Stochastic Parsing

Consider a sequence of observations $F_1, \ldots, F_T$ generated by the interaction between two people as explained in the previous section. With the model developed in the previous section each short-term action observable F_t in the sequence is

classified regardless their relationships with past or future observables. Short-term actions usually follow some activity patterns which, on the other hand, depend on the context in which they are found. Often, such patterns, or long-term behaviors, can only be characterized at long time extents comprising tens or even hundreds of short-term action observations. Therefore, we claim that more robust detection of short-term actions is achieved when shifting up to the level of long-term behavior analysis.

Inspired by the work in [8], we model long-term behaviors as grammar production rules. Recognition of long-term behaviors transforms then to finding the sequence compatible with the rules that best fits to the observables, which is essentially a parsing problem.

Stochastic grammars provide a proper framework to do so since they allow for probabilistic measurements both in the observations and the production rules. Stolcke [12] proposed an efficient parsing algorithm for stochastic grammars. A stochastic grammar is a tuple $G = (\mathcal{N}, \Sigma, \mathcal{R}, S, \mathcal{P})$, where $\mathcal{N}$ are the non-terminals, Σ the terminals, $\mathcal{R}$ the rules, S the starting non-terminal and $\mathcal{P}$ the rule probabilities. We mainly restrict to the sub-type of regular grammars. Regular grammars have the same expressive power as finite-state machines (FSM) [7], the latter ones traditionally used for representing human activity. Moreover, it is possible a direct interpretation of rule probabilities as transition probabilities, which facilitates the task of estimating them. In the FSM formalism, which we use to illustrate our method, observations and states correspond to terminal and non-terminal symbols in the grammar. The rules of a regular grammar have the following forms.

$C \rightarrow s$, where C is in $\mathcal{N}$ and s is in Σ
$C \rightarrow sD$, where C, D are in $\mathcal{N}$ and s is in Σ

As a more appropriate abstraction, we transform the sequence of observables $F_1 \dots F_T$ into a sequence $S_1 \dots S_T$, where each position $\mathcal{S}_t^{action_p} = P\left(F_t | action_p\right)$ accounts for the probability of observation of each short-term action at each time step. Given any sub-sequence $\mathcal{S}_a \dots \mathcal{S}_b$, the stochastic parser delivers:

- The Viterbi parse. This is, the most likely sequence of (unambiguous) short-term actions that we would observe if they were produced following the rules of behavior $C \in \mathcal{N}$,

$$s_a \dots s_b = \mathit{Viterbi\_parse}\left(\mathcal{S}_a \dots \mathcal{S}_b | C\right) , \tag{5}$$

  where $s_t \in \mathcal{A}$.
- The Viterbi probability of such a sequence, which in our case is a product of observation probabilities and transition probabilities as defined by the Viterbi parse. This is,

$$P\left(\mathcal{S}_a \dots \mathcal{S}_b | C\right) = P\left(F_a | s_a\right) \prod_{t=a+1}^{b} P\left(s_t | s_{t-1}\right) P\left(F_t | s_t\right) \tag{6}$$

  where $s_a \dots s_b$ is the Viterbi parse of $\mathcal{S}_a \dots \mathcal{S}_b$, and $P\left(s_i | s_j\right)$ is the probability of transition from action $s_i \in \mathcal{A}$ to action $s_j \in \mathcal{A}$.

A novel contribution of our method is that we divide our grammar into two parts: the constrained and the unconstrained part. The constrained part is responsible for interpreting the sequence of incoming short-term actions $\mathcal{S}$ according to the specified rules of the behaviors. The unconstrained part provides a straightforward interpretation that does not impose any structure at all. Separation of the grammar in constrained and unconstrained part has two advantages. On one hand it allows to parse any input sequence without interruptions (since the unconstrained part accepts any sequence). On the other hand it also provides a reference to validate candidate recognitions (as we will see later).

To illustrate this idea suppose that we want to recognize a set of M predefined long-term behaviors. Suppose that the i-th long-term behavior is composed by the sequence of short-term actions *join* - *interact* - *split*, and that the terminals of our grammar are $\Sigma = \{join, interact, split\}$. Figure 1 shows a representation.

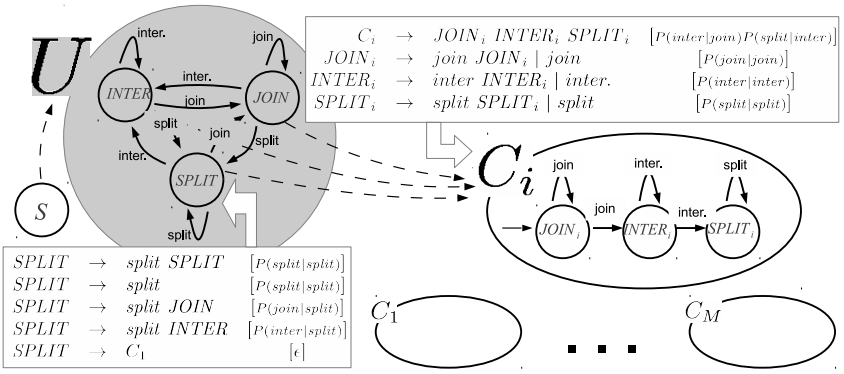

Fig. 1. Terminals and non-terminals are in non-captials and capitals, respectively. Non-terminal S is the starting symbol. The unconstrained part is encapsulated in non-terminal U and represented in grey. As an example of this part we show the production rules of the $SPLIT$ non-terminal. The unconstrained part is encapsulated in non-terminals C_i which contain the rules of the i-th pre-specified behavior. Solid arrows represent transitions associated with observations. Dashed arrows represent transitions not associated with any observation that have a fixed probability ϵ. They give the capability of detecting behaviors starting at any moment in time.

The procedure for recognizing long-term behaviors is the following. At each time step t, the stochastic parser processes the whole set of detections from that time, $\mathcal{S}_t^{action_p}, \forall action_p$. Operation consists of a series of *prediction*, *scanning* and *completion* steps. Each time step that a non-terminal C_i is *completed* means that the parser has found a sub-string that is compatible with the rules of C_i. This traduces to a candidate detection of the long-term behavior C_i from which the time interval $[a, b]$ can be easily retrieved (check [12] for details). Final decision is based upon comparison of the constrained and unconstrained interpretations. This is, behavior is recognized if the probability of the sequence $\mathcal{S}_a \ldots \mathcal{S}_b$ being

generated by the constrained rule C_i is not too low with respect to the probability of being generated by the unconstrained rule U. More precisely, behavior is recognized if

$$\frac{P\left(\mathcal{S}_a, \ldots, \mathcal{S}_b | C_i\right)}{P\left(\mathcal{S}_a, \ldots, \mathcal{S}_b | U\right)} \geq \rho^{(b-a+1)} \tag{7}$$

where $0 \leq \rho \leq 1$ controls the tolerance to false positives / false negatives, and the exponent makes this measure invariant to the length of the sequence.

As previously stated, the aim of our method is to deliver an unambiguous sequence of short-term action detections $s_1 \ldots s_T$ from an input sequence of probabilistic observations $\mathcal{S}_1 \ldots \mathcal{S}_T$. We define the *null* action *ignore* for the cases when no decision can be made. Final action detection is decided as

$$s_t = \begin{cases} \textit{ignore if } \nexists a, b, C_i \textit{ s.t. } a \leq t \leq b \wedge \dfrac{P\left(\mathcal{S}_a, \ldots, \mathcal{S}_b | C_i\right)}{P\left(\mathcal{S}_a, \ldots, \mathcal{S}_b | U\right)} \geq \rho^{(b-a+1)} \\ s'_t \qquad \textit{otherwise} \end{cases}, \tag{8}$$

where

$$\begin{aligned} & s'_a \ldots s'_b = \textit{Viterbi\_parse}\left(\mathcal{S}_a \ldots \mathcal{S}_b | C\right) \\ & \textit{such that } \{C, a, b\} = \arg\max_{C', a', b'} \frac{P\left(\mathcal{S}_{a'}, \ldots, \mathcal{S}_{b'} | C'\right)}{P\left(\mathcal{S}_{a'}, \ldots, \mathcal{S}_{b'} | U\right)} \end{aligned} \tag{9}$$

In the case that multiple partly overlapping parses we only select the one with maximum value of equation (7).

We show how we estimate transition probabilities from training data:

$$P\left(action_q | action_p\right) = \frac{\sum_{(i,j)} \sum_k L\left(F_k^{(ij)}; action_p\right) \cdot L\left(F_{next(k)}^{(ij)}; action_q\right)}{\sum_{action_{q'}} \sum_{(i,j)} \sum_k L\left(F_k^{(ij)}; action_p\right) \cdot L\left(F_{next(k)}^{(ij)}; action_{q'}\right)} \tag{10}$$

where F_{ij}^k is a particular window, $next(k)$ is a function that returns the next window in time to k, and $L\left(\bullet\right)$ is the label indicator function of equation (3).

4 Experiments and Results

We have performed experiments on the CAVIAR dataset [1]. The CAVIAR database consists of a set of clips showing long-term behaviors. There are annotations of the bounding boxes as well as labels of short-term interactions between pairs of people. Such interactions are: *join*, *fight*, *interact*, *move*, *leave victim*, *leave object* and *split*. We have created an additional label *ignore* corresponding to the *null* action for the cases when two people are close to each other without interacting. Due to the extremely low prevalence of the labels *leave victim* and *leave object* as well as to some arbitrariness of the human annotator in the case of the *leave object* label discard them by assigning to the *ignore* label. In terms of positional features, the actions *fight* and *interact* are equivalent (i.e.,

both consist on two people interacting close to each other). Therefore, we have decided to merge both labels into one called *fighteract*.

According to the structure of the behaviors defined in the CAVIAR documentation [1] and the modifications that we have made to the labels, we have identified the two long-term behaviors of figure 2 as the ones represented in the clips.

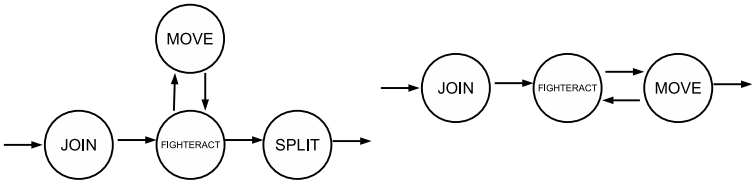

Fig. 2. Long-term behaviors shown by the CAVIAR clips.

From a total of 27 clips 7 of them are discarded because they contain no pairwise interactions of people at all (neither from the *ignore* class). From the 20 videos used in the experiments, 8 of them contain exclusively interactions of the type *ignore* (i.e., they do not show any interaction between actors but some of them get close to each other at some moment). The remaining 12 clips contain relevant interactions and eventually also *ignore*-type interactions.

We segment videos into windows of size *ws* with an overlap defined by offset *wo*, identical to our learning framework for the observations of short-term actions. We use ground truth annotations of bounding boxes to get the trajectories of each person by projecting the position of the feet with the homography relating the image plane with the ground plane. Because of the low prevalence of certain classes in the dataset (e.g., *split*), we use the data from all videos except the current test one as training set for the short-term action detectors.

In order to see the benefits of imposing the structure of the behavior through a grammar, we compare the accuracy of short-term action detection obtained using either the K-nn classifier of equation (4) or the output of the stochastic parsing as defined in equation (8).

We show both mean Matthew's Correlation Coefficient (MCC) between classes and confusion matrices. MCC is a measure of quality of two-class classification defined as

$$\mathrm{MCC} = \frac{\mathrm{TP}\times\mathrm{TN}-\mathrm{FP}\times\mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}\ , \tag{11}$$

where $\mathrm{TP}, \mathrm{TN}, \mathrm{FP}, \mathrm{FN}$ account for true positives, true negatives, false positives and false negatives, respectively. An MCC value of 1 means perfect prediction. A value of 0 means not better than random prediction. A value of -1 indicates total disagreement.

In the table below we show both the mean MCC among the classes and the MCC between the *ignore* and the rest of the classes obtained by each method.

Per-class (MCC)	
K-nn (Hausdorff)	**0.33**
K-nn (EMD)	**0.37**
Grammar + K-nn (Hausdorff)	**0.46**
Grammar + K-nn (EMD)	**0.48**

ignore vs. all (MCC)	
Grammar + K-nn (Hausdorff)	**0.77**
Grammar + K-nn (EMD)	**0.75**

As we see in the per-class results, grammar-based methods obtain better classification accuracies than the others, specially the EMD-based one. EMD-based variants usually outperform Hausdorff-based ones. This is especially true when not using grammars, when the differences are more noticeable. From these results we deduce that recognition of short-term actions in the CAVIAR dataset is improved when imposing their expected long-term structures as shown in figure 2. As we see in the *ignore* vs. all results, grammar-based methods are quite successful in discriminating between the *ignore* class and the rest. It means that they succeed in detecting when some predefined behavior happens.

Confusion matrices are shown in figure 3. Rows represent actual detections while columns represent ground truth classes. Perfect detections would show a matrix with ones in the diagonal and zeros elsewhere.

(a) ws=20, wo=10, K=5 (b) ws=50, wo=20, K=5 (c) ws=40, wo=40, K=30, ρ=0.2 (d) ws=30, wo=20, K=50, ρ=0.2

Fig. 3. Values for the parameters corresponding to the best results are shown under each confusion matrix. The methods are: (a) K-nn (Hausdorff), (b) K-nn (EMD), (c) Grammar + K-nn (Hausdorff) and (d) Grammar + K-nn (EMD).

As we see in the confusion matrices (and from the tables above) the EMD-based variant of the grammar method outperforms the rest. Short-term detectors tend to overestimate the *ignore* class. The *split* action has a significantly low prevalence in the training set. Due to this, short-term detectors tend to confuse it with the most prevalent classes *ignore* and *move*. Grammar-based methods correctly deduce that they are part of some long-term behavior but miss-classify them as *move* because action detectors tend to do so and also due to the structural compatibility between *split* and *move* states in the rules of figure 2.

5 Conclusions

We have presented a method to improve the recognition of short-term actions as well as to recognize long-term behaviors by imposing a behavior structure. It is

useful for the cases when few (or no) training data about long-term behaviors is available. It uses non-parametric detection of short-term actions in the bottom layer which are input to a stochastic parser in the top layer.

We propose a new methodology for estimating probabilities of the grammar rules as well as we introduce a novel criterion for recognizing long-term behaviors based on the allowed deviation from the straightforward interpretation. We propose a new variant of the short-term action detector based on the EMD.

We perform experiments of recognition of long-term interactions between people in the CAVIAR dataset [1]. Results show that the EMD variants usually outperform the Hausdorff-based ones. Moreover, grammars are quite successful in recognizing pre-defined behaviors in the CAVIAR dataset as we see in the *ignore* vs. *the rest* classification results. Regarding per-label classification, they present an average improvement of $\backsim$ 25%. This demonstrates that imposing long-term behavior structure improves short-term action detection.

Acknowledgments. This work has been carried out as part of the EU FP SEC project ARENA.

References

1. (2004), `http://homepages.inf.ed.ac.uk/rbf/caviar/`
2. Blunsden, S., Andrade, E.L., Fisher, R.B.: Non Parametric Classification of Human Interaction. In: Martí, J., Benedí, J.M., Mendonça, A.M., Serrat, J. (eds.) IbPRIA 2007. LNCS, vol. 4478, pp. 347–354. Springer, Heidelberg (2007)
3. Brdiczka, O., Yuen, P.C., Zaidenberg, S., Reignier, P., Crowley, J.L.: Automatic acquisition of context models and its application to video surveillance. In: ICPR, pp. 1175–1178 (2006)
4. Burghouts, G.J., Marck, J.W.: Reasoning about threats: From observables to situation assessment. IEEE Transactions on Systems, Man, and Cybernetics, Part C 41(5), 608–616 (2011)
5. Burghouts, G., Schutte, K.: Correlations between 48 human actions improve their detection. In: ICPR (2012)
6. Fernández-Caballero, A., Castillo, J.C., Rodríguez-Sánchez, J.M.: Human activity monitoring by local and global finite state machines. Expert Syst. Appl. 39(8), 6982–6993 (2012)
7. Hays, D.G.: Chomsky hierarchy. In: Encyclopedia of Computer Science, pp. 210–211. John Wiley and Sons Ltd., Chichester
8. Ivanov, Y.A., Bobick, A.F.: Recognition of visual activities and interactions by stochastic parsing. Pattern Anal. Mach. Intell. 22(8), 852–872 (2000)
9. Kasteren, T.L., Englebienne, G., Kröse, B.J.: An activity monitoring system for elderly care using generative and discriminative models. Personal Ubiquitous Comput. 14(6), 489–498 (2010)
10. Kitani, K.M., Sato, Y., Sugimoto, A.: Recovering the basic structure of human activities from a video-based symbol string. In: WMVC, p. 9 (2007)
11. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. Int. J. Comput. Vision 40(2), 99–121 (2000)
12. Stolcke, A.: An efficient probabilistic context-free parsing algorithm that computes prefix probabilities. Comput. Linguist. 21(2), 165–201 (1995)