A Pilot Study on Detecting Violence in Videos Fusing Proxy Models

Marc Roig Vilamala, Liam Hiley, Yulia Hicks, Alun Preece, and Federico Cerutti
Cardiff University

***Abstract*—We propose a robust—to new dataset and situation—approach to detect violence in CCTV feeds that breaks with the traditional assumption of having large amounts of training data that are representative samples. Detecting violence in CCTV feeds is an objectively hard problem that is of paramount importance to solve for effective situational understanding. Violence comprises a large spectrum of activities that can go from abuse, to fighting, to road accidents, that can therefore take place in completely different environments, from public buildings, to underground stations, to roads during the day or the night. This is therefore one of those tasks at which humans excel, while machines still lag behind. We show that there are specific, detectable, and measurable features of video feeds that correlate with—among other things—violence and, by fusing such features with semantic knowledge, we can in principle provide estimates of sequences of videos that correlate with violence.**

***Index Terms*—uncertain sources, complex event processing**

I. Introduction

Detecting violence in CCTV feeds is an objectively hard problem that is of paramount importance to solve for effective situational understanding. Situational understanding requires both insight and foresight. In its traditional definition [1] it is the "product of applying analysis and judgement to the unit's situation awareness to determine the relationships of the factors present, and form logical conclusions concerning threats to the mission accomplishment, opportunities for mission accomplishment, and gaps in information." The UK Ministry of Defence Doctrine [2] goes further, explicitly mentioning that (situational) "understanding involves acquiring and developing knowledge to a level that enables us to know why something has happened or is happening (insight) and be able to identify and anticipate what may happen (foresight)."

Violence, in particular, comprises a large spectrum of activities that can go from abuse, to fighting, to road accidents, that can therefore take place in completely different environments, from public buildings, to underground stations, to roads during the day or the night. This is therefore one of those tasks at which humans excel, while machines still lag behind.

It is therefore surprising that state-of-the-art approaches [3] assume—as is traditional—the existence of a large set of training data that is representative of the domain. Such machine learning techniques can only interpolate from the data that trained them: they cannot extrapolate knowledge about a test sample that is far different than the training data [4]. Such a traditional approach is not sustainable: a novel, robust approach to learning and reasoning must be developed. Robustness (Section II) against new dataset and situations depends on specific and measurable attributes of violence: our assumptions are that (1) rapid movements correlate with acts of violence; and (2) acts of violence—involving human activities—last more than fractions of a second. If such hypotheses are satisfied, a solution can be envisaged by fusing simple events detected in a few frames, together with semantic knowledge of the scenes obtained via object detection. Both detectors are not expected to be trained on datasets that include violence, thus breaking with the traditional approach. We rely on Complex Event Processing (CEP) [5], [6] as a method to fuse symbolic and sub-symbolic pieces of information. CEP is a mature technique used for processing and analysing streams of data from multiple sources to detect complex event patterns that suggest complicated situations. In particular, we will rely on CEP via event calculus over probabilistic logic programming [7] (cf. Section III-C).

We found evidence (Section IV) in favour of our hypotheses when using UCF-Crime [3], a state-of-the-art dataset on crime and violent activities. We also found evidence that even one of the simplest definitions of complex event—two simple events in a row—together with basic background knowledge—e.g., abusing a person requires having at least two people in the frame—helps detect scenes of violence in our pilot study (Section V). This is a significant milestone in our ongoing research aimed at providing robust methodologies to detect violence in CCTV video feeds, and in Section VI we discuss our next steps.

II. CCTV Analytic Needs to be Robust

The problem of detecting violence in CCTV footage is a special case of the *anomaly detection problem*, one of the most challenging and long standing problems in many domains, from cybersecurity [8] to video analytics in general [9]–[17], [17]–[20]. In [9], [20] the authors used deep learning based autoencoders to learn the model of normal behaviours: this is clearly one of the preferred ways to address the problem, under the assumption that *normality* can be defined with sufficient accuracy in a given domain.

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

When it comes to detecting violence or aggression in videos, approaches comprise the use of (1) models of motions and limbs orientation of people [21]; (2) multi-modality learning [22]; and (3) behaviour heuristics [23]. Due to successful demonstration of deep learning for image classification, several approaches have been proposed for video action classification [24], [25].

However, obtaining annotations for training is difficult and laborious, especially for videos. To our knowledge, [3], demonstrated on UCF-Crime, is currently the state-of-the-art in this domain. UCF-Crime [3] is a dataset comprised of 950 clips taken from CCTV footage, each clip categorised as one of 13 classes of *abnormal* behaviour, such as abuse—to be interpreted as *cruel and violent treatment of a person or animal*[1]—, arrest, arson, assault, burglary, explosion, fighting, road accidents, robbery, shooting, shoplifting, stealing, and vandalism. UCF-Crime is also complemented with 950 *normal* videos where no such activities take place: the original intention was to create a larger dataset for anomaly detection.

In [3] the authors provide temporal labels for 140 videos to train an anomaly detector using multiple instance learning (MIL) [26], [27] achieving an overall AUC of 75.41. The authors treat normal and anomalous surveillance videos as bags and short segments/clips of each video as instances in a bag. Based on training videos, they automatically learn an anomaly ranking model that predicts high anomaly scores for anomalous segments in a video. During testing, a long, untrimmed video is divided into segments and fed into their deep network which assigns anomaly score for each video segment such that an anomaly can be detected.

A. Robustness Assurance: Working Assumptions/Hypotheses

Differently from most of the literature in machine learning and computer vision, we cannot commit to the assumption that it is possible to collect representative samples of anomaly situations we wish to detect. In [4] we discuss cases, such as controlling the crowd during a peace-keeping mission before riots will happen, that are extremely rare and unpredictable, thus posing a real challenge for sample data acquisition.

Our working assumption is like having a blind software agent, Sandy, parachuted into uncharted territory. What Sandy roughly knows is types of violent situations they will witness—let us say violence between humans—and they possess (1) sensors informing them whether what we will name simple events of unknown types happen in a short, fixed—but unknown—timeframe (typically 0.5 seconds) and their probability; and (2) sensors informing them of the presence of humans, animals, and certain objects with associated probability. Sandy's mission is to develop heuristics to guess whether they witness a situation or not. We therefore assume Sandy can leverage:

(a) a system able to detect simple events in very short video segments, trained on datasets unrelated to those Sandy will witness;

(b) a system able to detect humans, objects, and animals in images, once again trained on datasets unrelated to those Sandy will witness;

(c) a system for fusing information coming from the above two systems, and exogenous pieces of information.

Section III discusses in detail the three systems we chose for this investigation. To ensure Sandy can reasonably perform its duty, we prove the following two hypotheses considering the state-of-the-art dataset UCF-Crime:

Hypothesis 1: there is a relationship between the number of simple events taking place in each video of UCF-Crime and the type of violent situation present in it;

Hypothesis 2: there is a relationship between the number of simple events taking place in each video of UCF-Crime and the number of pre-defined combinations of simple events taking place in the same video, e.g., the number of cases where two simple events took place consecutively.

B. An Example from the UCF-Crime Dataset

Figure 1 depicts four frames of the video Abuse001, an instance of abuse. It is a portion of surveillance video recording the north vestibule of St Cecilia cathedral in Omaha, Nebraska, USA, where on 18th August 2015 CE, at about 11:06 am, a man struck a 76-year-old woman after another man stole her purse.[2] While it is known that Abuse001 contains element of abuse, only when looking at the video does it become apparent that the act of violence begins in frame 258 (Fig. 1b), when the first man steals the woman's purse, and then it continues in frame 300 (Fig. 1c), when the second man strikes the woman. The woman then falls to the ground where she will remain from frame 380 (Fig. 1d) until frame 765.[3] Therefore, only about 16% of these frames contain genuine abusive activity: cf. [3, Figure 5] for statistics on the whole testing set.

III. Proxy Models and How to Fuse Them

Our software agent can leverage the following three systems.

A. System (a): Kinetics

Our software agent is expected to be able to leverage a system able to detect simple events in very short video segments, trained on datasets unrelated to those it will witness. To this aim, in this analysis we consider Kinetics-400, a large-scale labelled dataset of various human-focused actions. The dataset is composed of approximately 300,000 YouTube video URLs containing examples of mostly human action [28]. The labelling for each video is decided automatically by inferring

[1] https://en.oxforddictionaries.com/definition/abuse (on 8th March 2019).

[2] Edited video is available on the Youtube channel of KETV NewsWatch 7—a local TV station—at https://www.youtube.com/watch?v=uswCrbMymvE (on 8th March 2019); the news story was reported also by the Omaha World Herald—a local newspaper—at https://www.omaha.com/news/crime/video--year-old-woman-attacked-at-st-cecilia-cathedral/article_c60f4dcc-45de-11e5-bd6c-53122e3ec616.html.

[3] In Abuse001 these 765 frames are repeated 3 more times, with the last repetition interrupting at the moment the two men enter the church, for a total of 2,727 frames.

(a) Frame 1 (b) Frame 258 (c) Frame 300 (d) Frame 380

Fig. 1: Frames of Abuse001 video related to the aggression of a woman in Omaha, Nebraska, USA. Original size: 320x240px RGB, including black bars (28px left and right, 30px top and bottom) removed here thus reducing images to 264x180px.

the activity from the title of the video, given by the uploader. Kinetics has come to be used as an analogue to ImageNet, as was the intent of the dataset's authors, useful in pre-training models for smaller datasets or more specific tasks. Following [29], we use a 3D ResNet with 34 convolutional layers pre-trained on Kinetics.

B. System (b): COCO

Our software agent also needs to leverage a system able to detect humans, objects, and animals in images, once again trained on datasets unrelated to those it will witness. To this end, we consider Mask R-CNN [30], a framework for object instance segmentation. It is based on Faster R-CNN [31] and is designed to output a class label, a bounding box, and an object mask for each candidate object. Mask R-CNN consists of two stages. The first stage is a Region Proposal Network, which proposes potential object bounding boxes. In the second stage, Mask R-CNN predicts the mask for the object in parallel to predicting the class and the box offset. COCO [32] is a large-scale dataset that can be used for object detection, segmentation and captioning. The dataset contains 91 different classes with 330,000 images of complex everyday situations. An implementation of Mask R-CNN has been made available [33], which includes pre-trained weights for COCO.

C. System (c): CEP with Probabilistic Logic Programming

Finally, our software agent needs to leverage a system for fusing information coming from the above two systems, and exogenous pieces of information. To this end, CEP aims at identifying complex events from a stream of simpler events, which tend to be records of short or instantaneous activities in a the stream of data. Complex events instead are longer and are composed by a sequence of simple events. There are many different approaches to find which complex events can be extracted from which lower-level events. Here we will focus on an approach [7] using probabilistic logic programming (PLP).

Probabilistic Logic Programming: ProbLog [34], [35][4] belongs to a family of PLP languages [36] following Sato's distribution semantics [37]. It extends logic programming by annotating some ground facts with their probability of being true, which generalizes a single program into a distribution over programs that share their rules, but differ in their databases. More specifically, a ProbLog program consists of two parts, a set F of ground probabilistic facts `p::f` where `p` is a probability and `f` a ground atom, and a set R of rules `h :- b1,...,bn` where `h` is a logical atom and the `bi` are literals.[5] While the semantics is defined for countably-infinite sets of probabilistic facts [37], we restrict the discussion to the finite case in the following. ProbLog considers the ground probabilistic facts as independent random variables, i.e., we obtain the following probability distribution P_F over truth value assignments to sets of ground facts $F' \subseteq F$: $P_F(F') = \prod_{f_i \in F'} p_i \cdot \prod_{f_i \in F \setminus F'} (1 - p_i)$.

As each logic program obtained by choosing a truth value for every probabilistic fact has a unique least Herbrand model, P_F can be used to define the *success probability* $P(q)$ of a query q, that is, the probability that q is true in a randomly chosen such program, as the sum over all programs that entail q: $P(q) := \sum_{\substack{F' \subseteq F \\ \exists \theta F' \cup R \models q\theta}} P_F(F') = \sum_{\substack{F' \subseteq F \\ \exists \theta F' \cup R \models q\theta}} \prod_{f_i \in F'} p_i \cdot \prod_{f_i \in F \setminus F'} (1 - p_i)$.

Inference in ProbLog is concerned with computing marginal probabilities of queries, i.e., ground atoms, under this distribution, potentially conditioned on a conjunction of evidence atoms. While this is a #P-hard problem in general, ProbLog relies on state-of-the-art knowledge compilation techniques to achieve scalable inference across a wide range of models.

Event Calculus in ProbLog: ProbEC [7] is an approach to complex event recognition that tries to adapt event calculus to handle uncertainty in event recognition. It is based on the idea from event calculus that there are variables, called *fluents*, that can assume different values at different points in time. For example, if F is a *fluent* it is possible to express $F = V$, which denotes that F has value V (at a specific point in time).We can determinate that $F = V$ holds for a particular point in time if it has been *initiated* at a previous point and it has not been *terminated* since. A fluent can be initiated and terminated multiple times in a row with the same value. In event calculus, the first initialisation and termination are the ones that change the state of the fluent. ProbEC assigns the statement $F = V$ with a probability that increases every time $F = V$ is initiated

[4] https://dtai.cs.kuleuven.be/problog/.

[5] For the semantics of ProbLog to be well-defined, the set of rules has to have a two-valued well-founded model for each subset of the probabilistic facts: a sufficient condition for this is for programs to be stratified, i.e., have no loops through negation [35], [36].

and decreases every time it is terminated. The amount by which this value increases or decreases is proportional to the confidence we have on the fact that $F = V$ is initiated or terminated.

ProbEC is implemented in ProbLog:[6] (Listing 1). Lines 1–8 in Listing 1 check if a fluent has a certain value at a certain point on time. The fluent F represents the event. If we want to represent that the event is happening we will use the value V of $true$. Otherwise, we will use the value of $false$. In order to check if it holds that the fluent has a certain value it checks if the *broken* clauses are false. Defined in lines 10–19, those clauses will be true if the assignment of the value we are asking about has finished, either because it has been terminated or because it has been assigned a different value. Without the need to modify any of this code, the user can then define when the events they are interested in are initialised or terminated with the clauses $initiatedAt$ and $terminatedAt$ respectively. This can be done with whichever conditions the user seems appropriate for the event they are trying to detect.

```
1  holdsAt(F = V, T) :-
2    initially(F = V),
3    not broken(F = V, 0, T).
4
5  holdsAt(F = V, T) :-
6    initiatedAt(F = V, Tprev),
7    Tprev < T,
8    not broken(F = V, Tprev, T).
9
10 broken(F = V, Ti, Tf) :-
11   terminatedAt(F = V, Tm),
12   Ti < Tm,
13   Tm < Tf.
14
15 broken(F = V1, Ti, Tf) :-
16   initiatedAt(F = V2, Tm),
17   V1 \== V2
18   Ti < Tm,
19   Tm < Tf.
```

Listing 1: ProbEC [7] implementation in ProbLog.

IV. Testing Our Hypotheses

Figure 2 illustrates the pipeline we developed for analysing the UCF-Crime dataset. Each video is segmented in overlapping windows of 16 frames each, i.e. $\langle s_0, s_8, s_{16}, \ldots \rangle$.[7] Each segment s_i is then analysed by a model trained on Kinetics-400, returning the Kinetics-400 class (henceforth named simple event) with highest probability from the softmax layer if above a threshold $\mu = 0.085$ (empirically determined), or the special class χ (representing the absence of information) with probability $1 - p_i$ where p_i the highest probability identified by the softmax layer. Each segment s_i is also passed to a model trained on COCO that identifies objects in each of the 16 frames. This serves as input to a function f that returns (1) the probability of having at least N objects of the same class in the segment s_i; and (2) the probability of having two objects close to each other within a segment.

[6]We changed the original code to adapt it to the current ProbLog2 architecture.

[7]Videos were padded to contain a number of frames multiple of 8.

Class	λ_{l_i}		$\widehat{\lambda}_{l_i}$		Num Frames	
	E	**SD**	**E**	**SD**	**E**	**SD**
Abuse	93.6	1.4	25.3	0.7	483.2	579.3
Arrest	148.9	1.7	40.9	0.9	742.9	1127.0
Arson	119.1	1.5	26.7	0.7	679.2	2179.5
Assault	72.1	1.2	20.9	0.6	324.3	358.0
Burglary	72.0	0.8	17.0	0.4	588.5	843.4
Explosion	105.3	1.5	21.1	0.6	630.5	2463.1
Fighting	164.5	1.8	50.0	1.0	646.7	724.1
RoadAccidents	36.1	0.5	8.6	0.2	216.8	267.1
Robbery	68.0	0.7	17.9	0.3	351.6	313.3
Shooting	66.9	1.2	17.1	0.6	368.2	441.5
Shoplifting	145.2	1.7	35.4	0.8	810.4	1309.5
Stealing	69.2	0.8	14.4	0.4	583.7	551.2
Vandalism	61.8	1.1	15.4	0.6	367.4	332.8

TABLE I: Expected value (E) and standard deviation (SD) of λ_{l_i}, $\widehat{\lambda}_{l_i}$, and number of frames in videos (mean and standard deviation of the sample) for each of the 13 l_i in UCF-Crime, in alphabetical order.

Let $\boldsymbol{v}$ be the list of all the videos in UCF-Crime, such that $\boldsymbol{v} = \boldsymbol{v}^{l_1} || \ldots || \boldsymbol{v}^{l_{13}}$, where l_i is any of the 13 labels of the dataset, and $||$ represents vector concatenation. For a video $v_x^{l_i}$, let $\phi_x^{l_i} = |\{\langle i, \xi \rangle \mid \exists a_i^{v_x} = \langle i, p_i, \xi \rangle, \xi \neq \chi\}|$, i.e., the number of simple events detected by the model trained on Kinetics-400 in the video $v_x^{l_i}$. Given a label l_i, let $\boldsymbol{\phi}^{l_i} = \langle \phi_1^{l_i}, \ldots, \phi_1^{|\boldsymbol{v}^{l_i}|} \rangle$ be the vector listing the occurrences of simple events between the videos with weak label l_i (cf. Section II-A).

Let Φ^{l_i} be the number of occurrences of simple events in videos weakly labelled l_i, $\Phi^{l_i} \sim Poisson(\lambda_i)$. Let us assume as prior $\Lambda_{l_i} \sim Gamma(\alpha_{l_i}, \beta_{l_i})$, with $\alpha_{l_i} = \beta_{l_i} = \epsilon$ arbitrary small constant (in the following $\epsilon = 10^{-100}$). Hence,

$$\Lambda_{l_i} \mid \Phi^{l_i} \;\sim\; Gamma\left(\alpha_{l_i} + \sum_{j=1}^{|\boldsymbol{v}^{l_i}|} \phi_j^{l_i},\; \beta_{l_i} + |\boldsymbol{v}^{l_i}|\right) \qquad (1)$$

With this analysis, we are in the position to discuss the first of the two hypothesis we list in Section II-A,

A. Hypothesis 1: Correlation between Number of Simple Events and Weak Label

Figure 3a and Table I evidence that there is a correlation between the number of simple events in videos and their weak labels. Figure 3a depicts the Gamma distributions of the various λ_{l_i} for each of the 13 weak labels l_i in UCF-Crime. Expected values and standard deviation are also provided in Table I, where the rather constrained values for standard deviations strengthen the validity of our analysis reinforcing the assumption of homogeneity, viz. that the number of simple events in videos with the same weak label l_i is homogeneous across all the videos. Such a correlation does not seem to depend on the length of the videos in the dataset, cf. Table I.

While such a correlation exists, it is not necessarily unique. From visual inspection of Figure 3a it is evident that there

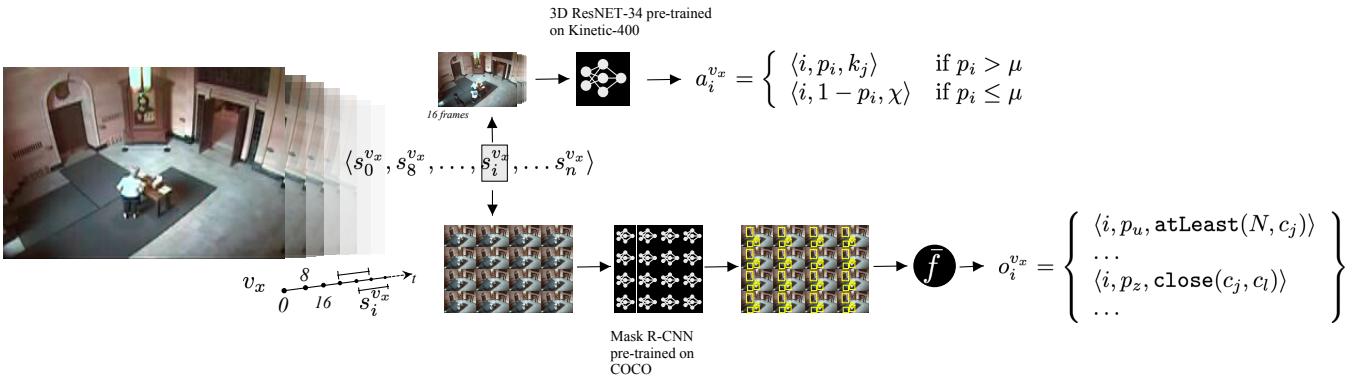

Fig. 2: Data processing pipeline for UCF-Crime.

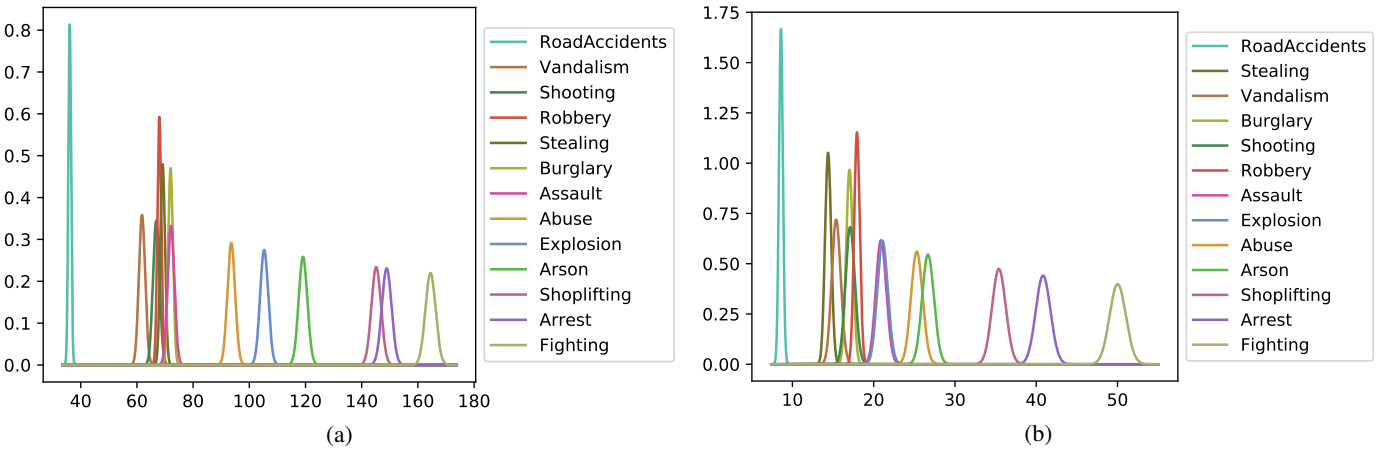

Fig. 3: Gamma distributions of λ_{l_i} (a), and of $\widehat{\lambda}_{l_i}$ (b) for each weak label l_i of UCF-Crime. Labels in each legend sorted by expected values.

are seven classes of videos that can be uniquely identified, namely *RoadAccidents*, *Explosion*, *Vandalism*, *Abuse*, *Shoplifting*, *Arrest*, and *Fighting*. The distributions for the remaining six classes—*Shooting*, *Arson*, *Robbery*, *Stealing*, *Assault*, and *Burglary*—are instead substantially overlapping. As discussed in Section VI, we will further investigate these cases by considering the (temporal) distribution of the 400 types of simple events within videos with such weak labels.

B. Hypothesis 2: Correlation between Number of Simple Events and Number of Combination of Simple Events

For a video $v_x^{l_i}$, let $\widehat{\phi}_x^{l_i} = |\{\langle i, \xi\rangle \mid \exists a_i^{v_x} = \langle i, p_i, \xi\rangle \text{ and } \exists a_i^{v_x} = \langle i-8, p_{i-8}, \xi'\rangle, \xi, \xi' \neq \chi\}|$, i.e., the number of occurrences of detecting two simple events consequently via the model trained on Kinetics-400 in the video $v_x^{l_i}$. Given a label l_i, let $\widehat{\boldsymbol{\phi}}^{l_i} = \langle \widehat{\phi}_1^{l_i}, \ldots, \widehat{\phi}_{|\boldsymbol{v}^{l_i}|}^{l_i}\rangle$ be the vector listing the occurrences of simple events between the videos with weak label l_i. Then, let $\widehat{\Phi}^{l_i}$ be the number of occurrences of two consequent simple events in videos weakly labelled l_i: $\widehat{\Phi}^{l_i} \sim Poisson(\widehat{\lambda}_i)$. From (1), we then have:

$$\widehat{\Lambda}_{l_i} \mid \widehat{\Phi}^{l_i} \sim Gamma\left(\alpha_{l_i} + \sum_{j=1}^{|\boldsymbol{v}^{l_i}|} \widehat{\phi}_j^{l_i},\ \beta_{l_i} + |\boldsymbol{v}^{l_i}|\right) \quad (2)$$

Figure 3b depicts the Gamma distributions of the various $\widehat{\lambda}_{l_i}$ for each of the 13 weak labels l_i of UCF-Crime: expected values and standard deviation are also provided in Table I.

The relationship that emerges between Figure 3a and Figure 3b (similarly between λ_{l_i} and $\widehat{\lambda}_{l_i}$ in Table I) evidences that there is a correlation between the number of simple events in videos and the number of consecutive occurrences of two simple events in the same video. In particular, the sampled ratio $\frac{E[\Lambda_{l_i}|\Phi^{l_i}]}{E[\widehat{\Lambda}_{l_i}|\widehat{\Phi}^{l_i}]}$ has E=4.04 and SD=0.48. Classes with a ratio outside one σ are: Assault: 3.45; Explosion: 4.99; Fighting: 3.29; and Stealing: 4.80.

In the following we will investigate specific cases belonging to the classes *RoadAccidents* and *Fighting* that manifest the lowest and highest λ_{l_i} and $\widehat{\lambda}_{l_i}$ respectively; as well as cases of the class *Abuse* that is average w.r.t. λ_{l_i}.

V. Four Case Studies

The four case studies we manually chose are:

RoadAccidents002. The video shows a street—with a few cars and bus moving—and the sidewalk with about 15 persons walking. After 7 seconds, a bus enters the scene and almost immediately runs into a sign, a lamp post, and a street advertising board (1 in Fig. 4a).

RoadAccidents110. The camera looks at a wide sidewalk showing open air market stalls and about 10 people walking (1 in Fig. 4b), with a road in the background. After about 4 seconds a blue truck driving in the inner lane of the road enters the scene (2 in Fig. 4b), followed one second later by a white truck. Another second later this second truck drives out of the street into the sidewalk, where it runs over a pedestrian (3 in Fig. 4b). The truck then falls to its side (4 in Fig. 4b).

Abuse001. This is the video described at length in Section II-B, cf. also Figure 1, about a woman robbed and punched to the floor by two men (1 in Fig. 4c) who then tried to stand up (2 in Fig. 4c).

Abuse007. This video starts with two teenagers and what appears to be a guard walking in a hallway. About 3 seconds into the video, the guard punches one of the teenagers in the face (1 in Fig. 4d), which causes him to fall to the floor. In the following, the characters move around the hallway (2, 3, 4, and 5 in Fig. 4d) but no further violence happens.

Fighting003. The video starts with a view of an underground station with about 10 people waiting. As the video goes on some other people come into frame (1 in Fig. 4e). After about one minute, the 15 people in the frame start a fight which leads some of them to run across the station (2 in Fig. 4f).

Fighting018 The video is clearly edited, as it begins with a title section stating "Teen girl using real karate. Self defence to beat a robber." In an underpass a man attempts to rob a girl, who then stops the robber, fights with him, knock him down (1 in Fig. 4f), and then runs away (2 in Fig. 4f). Other people walk pasts the robber on the floor (3, 4, 5 in Fig. 4f) until the video terminates with an animated closing section (6 in Fig. 4f).

A. Anomaly as Complex Event

Systems (a) and (b) (cf. Figure 2) analyse each video and produce as output a file listing whether something or not happened in a specific 16-frames window, as well as semantic information such as the presence of people or animals, and how *close* they are in the picture. Each information is augmented with its associated probability (cf. Listing 2).

```
1  0.1933::happensAt(something, 0).
2  0.9617::happensAt(nothing, 8).
3  0.3808::happensAt(atLeast(1, person), 8).
4  0.9249::happensAt(close(person, dog), 8).
```

Listing 2: Example of input to System (c)

Receiving this as input, System (c) can now exploit the relationship we show in Section II-A between λ_{l_i} and $\widehat{\lambda}_{l_i}$ (Hypothesis 2), namely to define an *anomaly* as a *complex event* each time there are two consecutive simple events. Listings 3 and 4 show our implementation of such a definition using the ProbEC approach (Section III).

```
1  initiatedAt(videoOnly = true, T) :-
2      happensAt(something, T),
3      Tnext is T + 8,
4      happensAt(something, Tnext).
5
6  initiatedAt(videoOnly = false, T) :-
7      happensAt(nothing, T),
8      Tprev is T - 8,
9      happensAt(nothing, Tprev).
```

Listing 3: Anomaly detection on UCF-Crime with ProbEC and data from System (a) only.

In Listing 3, lines 1 to 10 identifies boundaries of complex events on the basis of System (a) only. More specifically, lines 2 to 5 control the initialisation of the events and lines 7 to 10 control the termination. The initialisation checks that an event happens at the given frame (line 3) and then 8 frames later (lines 4 and 5), cf. Figure 2. Similarly for the termination of the complex event.

```
1  initiatedAt(videoObjDet = true, T) :-
2      initiatedAt(videoOnly = true, T),
3      happensAt(atLeast(2, person), T),
4      happensAt(close(person, person), T).
5
6  initiatedAt(videoObjDet = true, T) :-
7      initiatedAt(videoOnly = true, T),
8      happensAt(atLeast(1, person), T),
9      isAnimal(A),
10     happensAt(atLeast(1, A), T),
11     happensAt(close(person, A), T).
12
13 initiatedAt(videoObjDet = false, T) :-
14     initiatedAt(videoOnly = false, T).
15
16 isAnimal(dog).
```

Listing 4: Anomaly detection on UCF-Crime with ProbEC, fusing data from System (a) and System (b).

Data from System (b) is then fused together with data from System (a) in rules that begin respectively at line 1 and at line 6 of Listing 4. The first one (lines 1 to 4) says that there must be at least 2 persons in the frame (line 3). Then, it also says that two persons have to be close (line 4). Alternatively, the second one (lines 6 to 11) considers the cases where an animal is involved and it expects the animal to be close to a person (line 11). Line 16 shows an example on how to include an arbitrary long list of animals that System (b) can detect.

B. Empirical results

Figures 4(a–f) show the results of applying each of the two methods (cf. Listings 3 and 4) on the six selected videos. For each frame (X-axis), it is represented the probability that it is associated to an anomaly (complex event). Such probabilities appears small at first sight: this results from considering anything that the softmax layer of System (a) provides with probability greater than $\mu = 0.085$. For the purposes of this paper anything greater than 0 should be considered as a frame in which the System (c) detects anomaly.

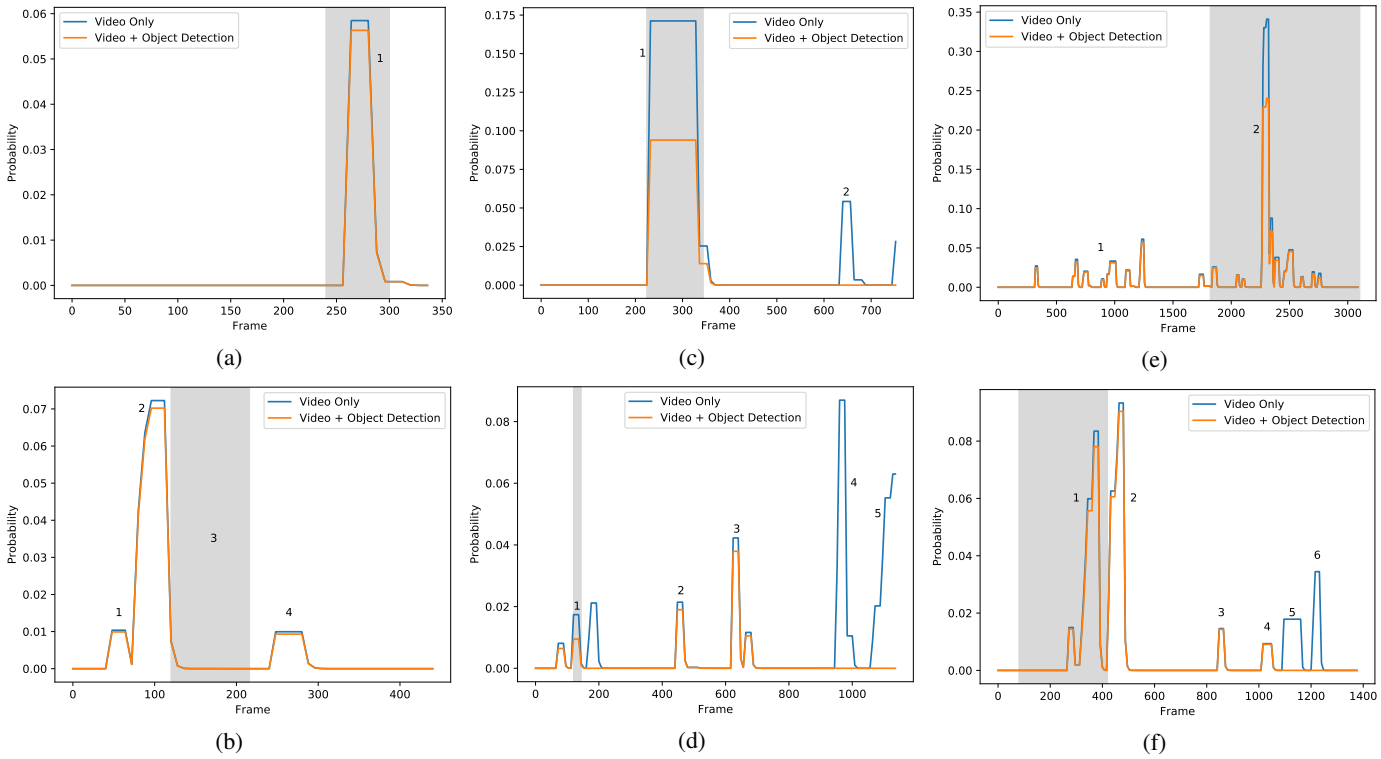

Fig. 4: Results for the six videos: RoadAccidents002 (a) and RoadAccidents110 (b) are instances of *RoadAccidents*; Abuse001 (c) and Abuse007 (d) are instances of *Abuse*; Fighting003 (e) and Fighting018 (f) are instances of *Fighting*. In grey the ground truth for the act of violence. In blue (Video only) results using only events detected by System (a) (Listing 3); in orange (Video + Object Detection) results obtained by fusing input from both System (a) and System (b) (Listing 4).

From visual inspection, it appears that there is correlation between the frames belonging to detected anomalies and the ground truth regarding violent actions. Violent actions are composed—if not identified—by anomalous complex events as detected by System (c) in five out of six of the videos that compose this pilot study. It is also worth noticing that fusing almost trivial background knowledge—cf. Listing 4—help filter out some false positives. For instance, in the case of Abuse001 (cf. Figure 1 and Section II-B), System (c) using only information coming from System (a) would identify two complex events (Figure 4c), one starting around frame 210—when the woman is attacked—and one starting around frame 620—when the woman tries to stand up. However, this second complex event is not considered when fusing with information coming from System (b) (cf. Listing 4), with the simple rule that for having an act of abuse on a person, at least two people need to be in the frame. From frame 620 onward only the woman is in the frame, hence it cannot be the case that there is an act of abuse happening (cf. orange line in Fig. 4c). The same can be said for the rule that the two people have to be close to each other. For example, in Abuse007, some of the false positives are removed thanks to this rule, despite the fact that there are always at least two people in the frame.

This pilot study shows that analysing combinations of simple events detected by System (a), fused with knowledge gathered by System (b) and by human expertise, can in principle help addressing the problem of identifying violence in real videos using proxy models. However, our pilot study also shows that there are clear limitations. Figure 4b depicts the results of System (c) on RoadAccidents110: the actual act of violence is completely missed. However, a different definition of complex event—notably at least two simple events detected by System (a) within any window of 16 frames (instead of the current 8)—would address it. Moreover, as also expected from the analysis of Section II-A, fights—cf. Figures 4e and 4f—will be more complicated to be analysed within our framework. However, multiple approaches that we believe can improve the performance will be considered in future work.

VI. Conclusion and Future Work

We present the results of a pilot study investigating the effectiveness of a novel, robust approach to learning and reasoning for detecting violence in CCTV streams. We found evidence (Section IV) in favour of the generality of our approach when using UCF-Crime [3], a current state-of-the-art dataset on crime and violence. We also found evidence that adopting the simplest fusion mechanism of actions detected by a 3D ResNet and knowledge obtained by object detection, helps detect scenes of violence in our pilot study (Section V).

In future work we will move away from the standard 3D ResNet architecture in favour of an Evidential Deep Learning (EDL) architecture. EDL aims at extending classical deep

learning with ideas from evidential reasoning [38] to quantify uncertainty in classification tasks [39]. For a sample, EDL tries to learn parameters of a predictive posterior as a Dirichlet density function for the classification of the sample. This is done by replacing the *softmax* layer in a classical deep classifier with an activation function that produces only non-negative outputs (e.g., *relu*, *softplus*, etc). The resulting output is considered as the evidence for the classification of the sample over n class labels. From the calculated evidence, parameters of the corresponding Dirichlet density is calculated. In this way we will be able to provide a threshold μ in a principled manner.

Secondly, we will take into consideration also the classes identified by the 3D architecture we will be using. In this pilot study we just distinguished between whether an action was detected or not (according to the μ parameter). An analysis in the sequence of actions detected and their correlation with violence activities is already envisaged.

Finally, we will investigate more articulated definitions of complex events. As discussed in Section V-B, in the case of RoadAccidents110 (cf. Figure 4b) the actual act of violence is completely missed. However, a different definition of complex event—notably at least two simple events detected by System (a) within any window of 16 frames (instead of the current 8)—would detect it. Striking a balance between accuracy over the UCF-Crime without overfitting is one of the major tasks for us in the foreseen future.

REFERENCES

[1] B. C. Dostal, "Enhancing situational understanding through employment of unmanned aerial vehicle," *Army Transformation Taking Shape: Interim Brigade Combat Team Newsletter*, vol. 01-18, 2007.

[2] "Understandig: Joint Doctrine Publication 04 (JDP 04)," Ministry of Defence, UK, 2016.

[3] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *CVPR*, 2018, pp. 6479–6488.

[4] L. Kaplan, F. Cerutti, M. Sensoy, A. Preece, and P. Sullivan, "Uncertainty Aware AI ML: Why and How," in *AAAI FSS-18: Artificial Intelligence in Government and Public Sector*, 2018. [Online]. Available: http://arxiv.org/abs/1809.07882

[5] L. J. Fülöp, G. Tóth, R. Rácz, J. Pánczél, T. Gergely, A. Beszédes, and L. Farkas, "Survey on complex event processing and predictive analytics," in *Proceedings of the Fifth Balkan Conference in Informatics*. Citeseer, 2010, pp. 26–31.

[6] I. Flouris, N. Giatrakos, A. Deligiannakis, M. Garofalakis, M. Kamp, and M. Mock, "Issues in complex event processing: Status and prospects in the big data era," *Journal of Systems and Software*, vol. 127, pp. 217–236, 2017.

[7] A. Skarlatidis, A. Artikis, J. Filippou, and G. Paliouras, "A probabilistic logic programming event calculus," *Theory and Practice of Logic Programming*, vol. 15, no. 2, pp. 213–245, 2015.

[8] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava, *A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection*, pp. 25–36.

[9] D. Xu, E. Ricci, Y. Yan, J. Song, and N. Sebe, "Learning deep representations of appearance and motion for anomalous event detection," in *BMVC*, September 2015, pp. 8.1–8.12.

[10] S. Wu, B. E. Moore, and M. Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," in *CVPR*, 2010, pp. 2054–2060.

[11] A. Basharat, A. Gritai, and M. Shah, "Learning object motion patterns for anomaly detection and improved object detection," in *CVPR*, 2008, pp. 1–8.

[12] X. Cui, Q. Liu, M. Gao, and D. N. Metaxas, "Abnormal detection using interaction energy potentials," in *CVPR*, 2011, pp. 3161–3167.

[13] B. Antić and B. Ommer, "Video parsing for abnormality detection," in *International Conference on Computer Vision*, 2011, pp. 2415–2422.

[14] T. Hospedales, S. Gong, and T. Xiang, "A markov clustering topic model for mining behaviour in video," in *International Conference on Computer Vision*, 2009, pp. 1165–1172.

[15] Y. Zhu, N. M. Nayak, and A. K. Roy-Chowdhury, "Context-aware activity recognition and anomaly detection in video," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 1, pp. 91–101, 2013.

[16] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 1, pp. 18–32, 2014.

[17] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *CVPR*, 2009, pp. 1446–1453.

[18] C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 fps in matlab," in *International conference on computer vision*, 2013, pp. 2720–2727.

[19] B. Zhao, L. Fei-Fei, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *CVPR*, 2011, pp. 3313–3320.

[20] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," in *Conference on Computer Vision and Pattern Recognition*, 2016, pp. 733–742.

[21] Y. Gao, H. Liu, X. Sun, C. Wang, and Y. Liu, "Violence detection using oriented violent flows," *Image and vision computing*, vol. 48, pp. 37–41, 2016.

[22] J. F. Kooij, M. Liem, J. D. Krijnders, T. C. Andringa, and D. M. Gavrila, "Multi-modal human aggression detection," *Computer Vision and Image Understanding*, vol. 144, pp. 106–120, 2016.

[23] S. Mohammadi, A. Perina, H. Kiani, and V. Murino, "Angry crowds: detecting violent events in videos," in *European Conference on Computer Vision*. Springer, 2016, pp. 3–18.

[24] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *CVPR*, 2014, pp. 1725–1732.

[25] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *International conference on computer vision*, 2015, pp. 4489–4497.

[26] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Prez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1, pp. 31 – 71, 1997.

[27] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *NIPS*, 2002, pp. 577–584.

[28] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *CVPR*, 2017.

[29] K. Hara, H. Kataoka, and Y. Satoh, "Can Spatiotemporal 3d CNNs Retrace the History of 2d CNNs and ImageNet?" in *CVPR*, 2018.

[30] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *CVPR*, 2017.

[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, 2014, pp. 740–755.

[33] W. Abdulla, "Mask r-cnn for object detection and instance segmentation on keras and tensorflow," https://github.com/matterport/Mask_RCNN, 2017.

[34] L. De Raedt, A. Kimmig, and H. Toivonen, "ProbLog: A probabilistic Prolog and its application in link discovery," in *IJCAI*, 2007, pp. 2462–2467.

[35] D. Fierens, G. Van den Broeck, J. Renkens, D. Shterionov, B. Gutmann, I. Thon, G. Janssens, and L. De Raedt, "Inference and learning in probabilistic logic programs using weighted Boolean formulas," *Theory and Practice of Logic Programming*, vol. 15, no. 03, pp. 358–401, 2015.

[36] L. De Raedt and A. Kimmig, "Probabilistic (logic) programming concepts," *Machine Learning*, vol. 100, no. 1, pp. 5–47, 2015.

[37] T. Sato, "A statistical learning method for logic programs with distribution semantics," in *ICLP*, 1995.

[38] A. P. Dempster, "A generalization of Bayesian inference," in *Classic works of the Dempster-Shafer theory of belief functions*. Springer, 2008, pp. 73–104.

[39] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential deep learning to quantify classification uncertainty," in *NIPS*, 2018.