Eurographics Workshop on Intelligent Cinematography and Editing (2017)
W. Bares, V. Gandhi, Q. Galvane, and R. Ronfard (Editors)

Inferring the Structure of Action Movies

Danila Potapov[1,*], Matthijs Douze[2,*], Jerome Revaud[3,*], Zaid Harchaoui[4,*], and Cordelia Schmid[5]

[1] Video Analysis Technologies LLC, Moscow, Russia [2] Facebook AI Research, Paris, France [3] Xerox Research Centre Europe, Meylan, France
[4] University of Washington, Seattle, WA, USA
[5] Inria, Laboratoire Jean Kuntzmann, Grenoble, France
[*] The work was done at Inria, Grenoble, France

Abstract

While important advances were recently made towards temporally localizing and recognizing specific human actions or activities in videos, efficient detection and classification of long video chunks belonging to semantically-defined categories remains challenging. Examples of such categories can be found in action movies, whose storylines often follow a standardized structure corresponding to a sequence of typical segments such as "pursuit", "romance", etc.
We introduce a new dataset, Action Movie Franchises*, consisting of a collection of Hollywood action movie franchises. We define 11 non-exclusive semantic categories that are broad enough to cover most of the movie footage. The corresponding events are annotated as groups of video shots, possibly overlapping. We propose an approach for localizing events based on classifying shots into categories and learning the temporal constraints between shots. We show that temporal constraints significantly improve the classification performance. We set up an evaluation protocol for event localization as well as for shot classification, depending on whether movies from the same franchise are present or not in the training data.*

Categories and Subject Descriptors (according to ACM CCS): I.2.10 [Artificial Intelligence]: Vision and Scene understanding—Video analysis

1. Introduction

An important problem of intelligent cinematography is the automatic parsing and indexing of movie collections. While understanding the full story is currently beyond reach, learning how to annotate the structure of a movie will be a powerful tool for intelligent cinematography.

Realistic videos include a wide variety of actions, activities, scene types, etc. During the last decade, significant progress has been made for action retrieval and recognition of specific, stylized, human actions. In particular, powerful visual features were proposed towards this goal [OVS13,OVS14,WS13]. For more general types of events in videos, such as activities, efficient approaches were proposed and benchmarked as part of the TrecVid Multimedia Event Detection (MED) competitions [OAM*14]. State-of-the-art approaches combine features from all modalities (text, visual, audio), static and motion features (possibly learned beforehand with deep learning), and appropriate fusion procedures.

In this work, we aim at detecting events of the same semantic level as TrecVid MED. However, we concentrate on real action movies that follow a structured scenario, whereas the content in TrecVid MED is mainly user-generated. From a movie scriptwriter's point of view [Sny05,RTT03], a Hollywood movie is more or less constrained to a set of standard story-lines. This standardization helps matching the audience expectations and habits. However,

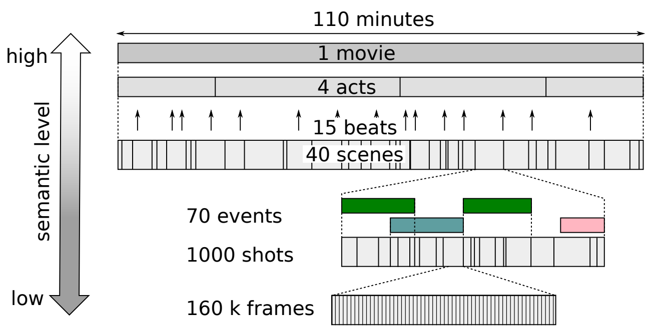

Figure 1: *Temporal structure of a movie, according to the taxonomy of "Save the Cat" [Sny05], and our level of annotation, the event.*

movies need to be fresh and novel enough to fuel the interest of the audience. Thus, some variability must be introduced in the story lines to maintain the interest. Temporally, movies are subdivided in a hierarchy of acts, scenes, shots, and finally, frames (see Figure 1). Punctual changes in the storyline give it a rhythm. They are called "beats" in [Sny05] and are common to many films. A typical example of beat is the moment when an unexpected solution saves the hero.

DOI: 10.2312/wiced.20171067

Figure 2: *Example frames illustrating the categories of the Action Movie Franchises dataset.*

From a computer vision point of view, frames are readily available and reliable algorithms for shot detection exist. Grouping shots into scenes is harder. Scenes are characterized by a uniform location, set of characters or storyline. The semantic level of beats and acts is out of reach. We propose here to attack the problem on an intermediate level by detecting "events". Temporally, they consist in sequences of consecutive shots and typically last a few minutes. Shots offer a suitable granularity, because movies are edited so that they follow the rhythm of the action. Semantically, they are of a higher level than the actions in most current benchmarks, but lower than the beats, which are hard to identify even for a human.

To support this study, we built an annotated dataset of Hollywood action movies, called **Action Movie Franchises**. It comprises 20 action movies from 5 franchises: *Rambo*, *Rocky*, *Die Hard*, *Lethal Weapon*, *Indiana Jones*. A movie franchise refers to a series of movies on the same "topic", sharing similar story lines and the same characters. In each movie, we annotate shots into several non-exclusive categories. We then create a higher level of annotation, "events", which consists of consistent sequences of shots labeled with the same category.

Figure 2 illustrates the categories that we use in our dataset. They are targeted at action movies and, thus, rely on semantic categories that often depend on the role of the characters, such as hero (good) or villain (bad). We now briefly describe all categories. First, we define three different action-related categories: *pursuit*, *battle preparation* and *battle*. We also define categories centered on the emotional state of the main characters: *romance*, *despair good* (e.g. when the hero thinks that all is lost) and *joy bad* (e.g. when the villain thinks he won the game). We also include different categories of dialog between all combinations of good and bad characters: *good argue good*, *good argue bad* and *bad argue bad*. Finally, we add two more categories identifying a temporary victory of a good or bad character (*victory good* and *victory bad*). We also consider a NULL category, corresponding to shots that can not be classified into any of the aforementioned categories.

In summary, we introduce the **Action Movie Franchises** dataset, which features dense annotations of 11 categories in 20 action movies at both shot and event levels. To the best of our knowledge, a comparable dense annotation of videos does not exist.

The semantic level of our categories will drive progress in action recognition towards new approaches based on human identity, pose, interaction and semantic audio features. Action movies and related professionally produced content account for a major fraction of what people watch on a daily basis. There exists a large potential for applications, such as access to video archives and movie databases, interactive television and automatic annotation for the shortsighted.

We believe the semantic level of these categories is appropriate, as identifying the "beats" in a movie is an exercise for would-be scriptwriters, identifying who are the good or bad characters in an action movie, even in a short extract, is accessible to 10-year-old children. However, as we will show later, state-of-the-art automatic processing methods still perform poorly on such categories. Therefore, the task has a potential to make progress research in automatic movie analysis.

Furthermore, we define several evaluation protocols, to investigate the impact of franchise-information (testing with or without previously seen movies from the same franchise) and the performance for both classification and localization tasks. We also propose an approach for classification of video shots into categories based on a state-of-the-art pipeline for multimodal feature extraction, classification and fusion. Our approach for localizing events uses a temporal structured inferred by a conditional random field (CRF) model learned from training data.

We make the Action Movie Franchises dataset publicly available to the research community to further advance video understanding. The dataset, the evaluation scripts, and the descriptors are available at `http://lear.inrialpes.fr/people/potapov/action_movies`.

2. Related work

2.1. Datasets

Table 1 summarizes recent datasets for action or activity recognition. Our Action Movie Franchises dataset mainly differs from existing ones with respect to the event complexity and the density of annotations. Similar to Coffee & Cigarettes and MediaEval Violent Scene Detection (VSD), our Action Movie Franchises

Name	# classes	example class	annotation unit	# train units	durations avg unit	annot	NULL	coverage
Classification datasets								
UCF 101 [SZS12]	101	high jump	clip	13320	7.21s	26h39	0h	-
HMDB 51 [KJG*11]	51	brush hair	clip	6763	3.7s	6h59	0h	-
TrecVid MED 11	15	birthday party	clip	2650	2m54	128h	315h	29%
Action Movie Franchises	11	good argue bad	shot	16864	5.4s	25h29	15h42	57.1%
Localization datasets								
Coffee & Cigarettes	2	drinking	time interval	191	2.2s	7m12s	3h26	3.3%
THUMOS detection 2014	20	floor gymnastics	t.i. on clip	3213	26.2s	3h22	167h54	2.0%
MediaEval VSD [DPSG14]	10	fighting	shot/segment	3206	3.0s	2h38	55h20	4.5%
Action Movie Franchises	11	good argue bad	event	2906	35.7s	28h49	14h08	61.4%

Table 1: *Comparison of classification and localization datasets; annot = total duration of all annotated parts; NULL = duration of the non-annotated (NULL or background) footage; coverage = proportion of annotated video footage.*

dataset is built on professional movie footage. However, while the former datasets only target short and sparsely occurring events, we provide dense annotations of events spanning larger time intervals. Our categories are also of significantly higher semantic level than those in action recognition datasets like Coffee & Cigarettes [LP07, WBTG09], UCF [SZS12] and HMDB [KJG*11]. A consequence is that our dataset remains very challenging for state-of-the-art algorithms, as shown later in the experiments. Events of a similar complexity can be found in TrecVid MED 2011–2014 [OAM*14], but our dataset includes precise temporally localized annotations.

2.2. Action detection in movies

Action detection (or action localization), that is finding if and when a particular type of action was performed in long and unsegmented video streams, received a lot of attention in the last decade. The problem was considered in a variety of settings: from still images [RS13], from videos [GHS13, WS13, CG16, YGG10], with or without weak supervision, etc. Most works focused on highly stylized human actions such as "open door", "sit down", which are typically *temporally salient* in the video stream.

Action or activity recognition can often be boosted using temporal reasoning on the sequence of atomic events that characterize the action, as well as the surrounding events that are likely to precede or follow the action/activity of interest. We shall only review here the "temporal context" information from surrounding events; the decomposition of action or activities into sequence of atomic events [GHS13] is beyond the scope of our paper. Early works along this line [RHM98] proposed to group shots and organize groups into "semantic" scenes, each group belonging exclusively to only one scene. Results were evaluated subjectively and no user study was conducted.

Several papers proposed to use movie (or TV series) scripts to leverage the temporal structure [ESZ06, MLS09]. In [MLS09], movie scripts are used to obtain scene and action annotations. Retrieving and exploiting movie scripts can be tricky and time-consuming. In many cases, movie scripts are simply not available. Thus, we did not use movie scripts to build our dataset and do not consider this information for training and testing. However, we do use another modality, the audio track, in a systematic way, and perform fusion following state-of-the-art approaches in multimedia [LNK04], and TrecVid competitions [OAM*14].

In [CJMT08], the authors structure a movie into a sequence of scenes, where each scene is organized into interlaced threads. An efficient dynamic programming algorithm for structure parsing is proposed. Experimental results on a dataset composed of TV series and a feature-length movie are provided. More recently, in [BBL*13], actors and their actions are detected simultaneously under weak supervision of movies scripts using discriminative clustering. Experimental results on 2 movies (*Casablanca* and *American beauty*) are presented, for 3 actions (*walking*, *open door* and *sit down*). The approach improves person naming compared to previous methods. In this work, we do not use supervision from movie scripts to learn and uncover the temporal structure, but rather learn it directly using a conditional random field that takes SVM scores as input features. The proposed approach is more akin to [HLDlT11], where joint segmentation and classification of human actions in video is performed on toy datasets [HDlT12].

3. Action Movie Franchises

We first describe the *Action Movie Franchises* dataset and the annotation protocol. Then, we highlight some striking features in the structure of the movies observed during and after the annotation process. Finally, we propose an evaluation protocol for shot classification into categories and for event localization.

3.1. The movies

The Action Movie Franchises dataset consists of 20 Hollywood action movies belonging to 5 famous franchises: *Rambo, Rocky, Die Hard, Lethal Weapon, Indiana Jones*. Each franchise comprises 4 movies; see Table 1 for summary statistics of the dataset.

Each movie is decomposed into a list of shots, extracted with a shot boundary detector [MLD*06, PDHS14]. Each shot is tagged with zero, one or several labels corresponding to the 11 categories (the label NULL is assigned to shots with zero labels). Note that the total footage for the dataset is 36.5 h, shorter than the total length

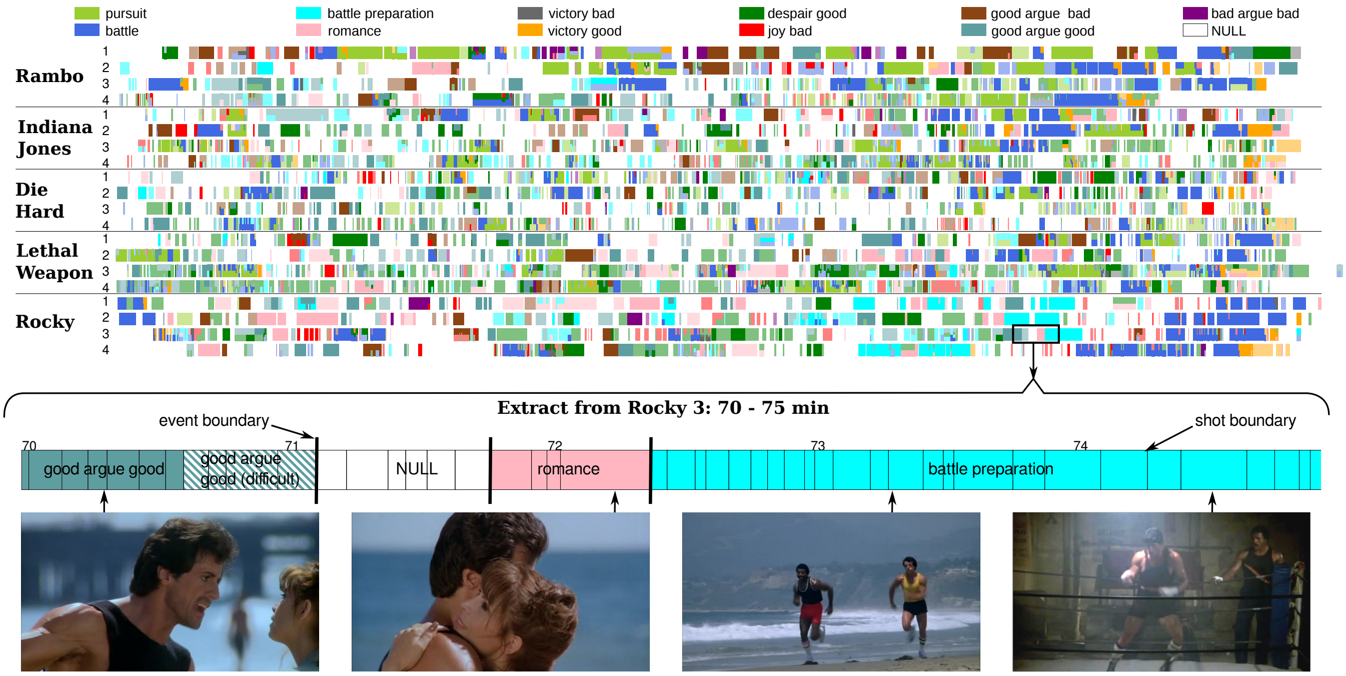

Figure 3: *Top: Events annotated for the Action Movie Franchises dataset, one movie per line, plotted along the temporal axis. All the movies were scaled to the same length. Bottom: zoom on minutes 70-75 of the* Rocky 3 *movie extract showing the shot segmentation, the annotations and the events.*

in Table 1. This is due to multiple labels. All categories are shown in Figure 2.

Series of shots with the same category label are grouped together in *events* if they all depict the same scene (ie. same characters, same location, same action, etc.). Temporally, we also allow an event to bridge gaps of a few unrelated shots. Events belong to a single, non-NULL, category.

The set of categories was inspired by the taxonomy of [Sny05], and motivated by the presence of common narrative structures in action movies. Indeed, category definitions strongly rely on a split of the characters into "good" and "bad" tags, which is typical in such movies. Each category thus involves a fixed combination of heroes and villains: both "good" and "bad" characters are present during *battle* and *pursuit*, but only "good" heroes are present in the case of *good argue good*. Initially, we considered a more extensive taxonomy, with "subject + verb", but that turned to be hard to define precisely, Thus, we focused on the most common combinations.

Large intra-class variation is due to a number of factors: duration, intensity of action, objects and actors, *and* different scene locations, camera viewpoint, filming style. For ambiguous cases we used the "difficult" tag.

3.2. Annotation protocol

The annotation process was carried out in two passes by three researchers. Ambiguous cases were discussed and resulted in a clear annotation protocol. In the first pass we manually annotated each shot with zero, one or several of the 11 category labels. In the second one we annotated the events by specifying their category, beginning and ending shots. We tolerated gaps of 1-2 unrelated shots for sufficiently consistent events. Indeed, movies are often edited into sequences of interleaved shots from two events, e.g. between the main storyline and the "B" story.

Some annotations are labeled as "difficult", if they are semantically hard to detect, or ambiguous. For instance, in *Indiana Jones 3*, Indiana Jones engages in a romance with Dr. Elsa Schneider, who actually betrays him to the "bad guy". Romance between Indiana Jones and Dr. Elsa Schneider is therefore ambiguous. We exclude these shots at training and evaluation time, as in the Pascal evaluation protocol [EVGW*10].

Our event annotations cover about 60 % of the movie footage, which is much higher than comparable datasets, see Table 1. This shows that the vocabulary we chose is representative: the dataset annotation is relatively dense. The 40% remaining footage mainly contains non-action events, like talking and walking.

3.3. Highlighting structure of action movies

Figure 3 shows the sequence of category-label annotations for several movies. Some global trends are striking: *victory good* occurs at the end of movies; *battle* is most prevalent in the last quarter of movies; there is a pause in fast actions (*battle*, *pursuit*) around the middle of the movies. In movie script terms, this is the "midpoint" beat [Sny05], where the hero is at a temporary high or low in the

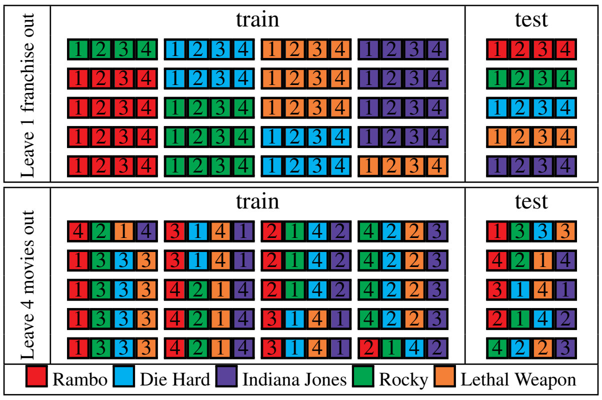

Figure 4: *The two types of split for evaluation. In addition to the train/test splits, the training videos are also split in 4* sub-folds*, that are used for cross-validation and CRF training purposes.*

story. In terms of event duration, *joy bad* and *victory bad* are short, while *pursuit* and *romance* are long. These trends can be learned by the temporal re-scoring to improve the shot classification results.

After careful analysis of the annotation, we find that *battle*, *despair good* and *pursuit* are the most prevalent categories, with 4145, 3042 and 2416 instances respectively. Since it is a semantically high level class, *despair good* is most often annotated as difficult. The co-occurrences of classes as annotations of the same shot follow predictable trends: *battle* co-occurs with *pursuit*, *battle preparation*, *victory good* and *victory bad*. Interestingly *romance* is often found in combination with *despair good*. This is typical for movies of the "Dude with a problem" type [Sny05], where the hero must prove himself.

Within each movie franchise, a shared structure may appear. For instance, in *Rocky*, the *battle preparation* occurs in the last quarter of the movie, and there is no *pursuit*.

3.4. Evaluation protocol

We propose two types of train/test splits and two performance measures for our Action Movie Franchises dataset.

Data splits

We consider two different types of splits over the 20 movies; see Figure 4. They both come in 5 folds of 16 training movies and 4 test movies. All movies appear once as a test movie. In the "leave one franchise out" setting, all movies from a single franchise are used as a test set. This simulates a viewer who is familiar with a franchise and sees a new one. In "leave 4 movies out", a single movie from each franchise is used as test. The two settings allow us to evaluate if our classifiers are specific to a franchise or generalize well across franchises.

Classification setting

In the classification setting, we evaluate the accuracy of category prediction at the shot level. Since a shot can have several labels, we adopt the following evaluation procedure. For a given shot with $n > 0$ ground-truth labels (usually $n = 1$, but the number of labels can be up to 4), we retain the best n predicted categories (out of 11, according to their confidence scores). Accuracy is then measured independently for each category as the proportion of ground-truth shots which are correctly labeled. We finally average accuracies over all categories, and report the mean and the standard deviation over the 5 cross-validation splits.

Localization setting

In the localization setting, we evaluate the temporal agreement between ground-truth and predicted events for each category. A detection, consisting of a temporal segment, a category label and a confidence score, is tagged positive if there exists a ground-truth event with an intersection-over-union score [EVGW*10] above 0.2. If the ground-truth event is tagged as "difficult" it counts neither as positive nor as negative. The performance is measured for each category in terms of average precision (AP) over all events in the test fold, and the different APs are averaged to a mAP measure.

4. Shot and event classification

The proposed approach consists of 4 stages. First, we compute high-dimensional shot descriptors for different visual and audio modalities, called *channels*. Then, we learn linear SVM classifiers for each channel. At the late fusion stage, we take the linear combination of the channel scores. Finally, predictions are refined by leveraging the temporal structure of the data and events are localized.

4.1. Descriptors extraction

For each shot from a movie, we extract different descriptors corresponding to different modalities. For this purpose, we use a state-of-the-art set of low-level descriptors [OVS13, A13]. It includes still image, face, motion and audio descriptors.
Dense SIFT [Low04] descriptors are extracted every 30^{th} frame. The SIFTs of a frame are aggregated into a Fisher vector of 256 mixture components, that is power- and L2-normalized [PJM10]. The shot descriptor is the power- and L2-normalized average of the Fisher descriptors from its frames. The output descriptor has 34559 dimensions.
Convolutional neural nets (CNN) descriptors are extracted from every 30^{th} frame. We pass the image to a CNN [KSH12] trained on Imagenet 2012, using the activations from the first fully-connected layer as description vector (FC6 with 4096 dimensions). The implementation is based on DeCAF [DJV*13], an off-the-shelf pre-trained network.
Motion descriptors are extracted for each shot. We extract improved dense trajectory descriptors [WS13]. The 4 components of the descriptor (MBHx, MBHy, HoG, HoF) are aggregated into 4 Fisher vectors that are concatenated. This output is a 108544 D vector.
Audio descriptors are based on Mel-frequency cepstrum features (MFCC) [RS07] extracted for 25ms audio chunks with a step size of 10ms. They are enhanced by first and second order temporal derivatives. The MFCCs are aggregated into a shot descriptor using Fisher aggregation resulting in a 20223D vector.

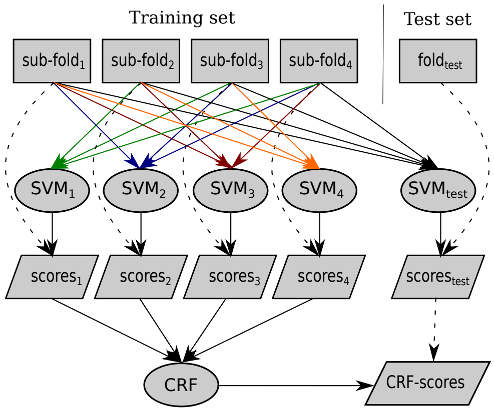

Figure 5: *Proposed training approach for one fold. In a first stage, $SVM_1 \ldots SVM_4$ are trained in leaving one sub-fold out of the training set, and are evaluated on the left-out sub-fold. In a second stage, a CRF model is trained, taking the sub-fold SVMs scores as inputs. We then use all the training videos to train the final SVM model (SVM_{test}). The final model outputs scores on the test fold, which are then refined by the CRF model. Note that each SVM training includes calibration using cross validation.*

Face descriptors are obtained by first detecting faces in each frame using the Viola-Jones detector from OpenCV [Bra00]. Following the approach from [ESZ06], we connect the detections into face tracks using the KLT tracker, allowing us to recover some missed detections. Each facial region is then described with a Fisher vector of dense SIFTs [SPVZ13] (16384 dimensions) which is power- and L2-normalized. Finally, we average-pool all face descriptors within a shot and normalize again the result to obtain the final shot descriptor.

Overall, each 2-hour movie is processed in 6 hours on a 16-core machine. We make all descriptors publicly available at `http://lear.inrialpes.fr/people/potapov/action_movies`.

4.2. Shot classification with SVMs

We now detail the time-blind detection method, that scores each shot independently without leveraging temporal structure.

Per-channel training of SVMs

The 5 descriptor channels are input separately to the SVM training. For each channel and for each category, we use all shots annotated as non-difficult as positive examples and all other shots (excluding difficult ones) as negatives to train a shot classifier. We use a linear SVM and cross-validate the C parameter, independently for each channel. We compute one classifier SVM_{test} per fold, and 4 additional classifiers $\text{SVM}_1 \ldots \text{SVM}_4$ corresponding to sub-folds, see Figure 5.

Late fusion of per-channel scores

The per-channel scores are combined linearly into a shot score. For one fold, the linear combination coefficients are estimated using the sub-fold scores. We use a random search over the 5D space of coefficients to find the one that maximizes the average precision over the sub-folds. This optimization is performed jointly over all classes (shared weights), which was found to be better to reduce the variability of the weights.

4.3. Leveraging temporal structure

We leverage the temporal structure to improve the performance of the time-blind detection/localization method, using a conditional random field (CRF) [LMP01]. We consider a CRF that takes the SVM scores as inputs. The CRF relies on a linear chain model. Unary potentials correspond to votes for the shot labels, while binary potentials model the probability of the sequences.

We model a video with a linear chain CRF. It consists of latent nodes $y_i \in \mathcal{Y}, i = 1, \ldots, n$ that correspond to shot labels. Similar to HMM, each node y_i has a corresponding input data point $x_i \in \mathbb{R}^d$. Variables x_i are always observed, whereas y_i are known only for training data. An input data point $x_i \in \mathbb{R}^d$ corresponds to the shot descriptor, which in our case is the 11-D vector of L2-normalized SVM scores for each category. The goal is to infer probabilities of shot labels for the test video. The CRF model for one video is defined as:

$$\log p(Y|X;\boldsymbol{\lambda},\boldsymbol{\mu}) = \sum_{i=1}^{n} \boldsymbol{\lambda}^T \boldsymbol{f}(y_i, X) + \sum_{i=1}^{n-1} \boldsymbol{\mu}^T \boldsymbol{g}(y_i, y_{i+1}, X),$$

where the inputs are $X = \{x_1, \ldots, x_n\}$ and the outputs $Y = \{y_1, \ldots, y_n\}$. We use the following feature (in the CRF literature sense) functions f and g:

$$f_k(y_i, X) = x_{i,k}\delta(y_i, k)$$
$$g_{k',k''}(y_i, y_{i+1}, X) = \delta(y_i, k')\delta(y_{i+1}, k'')$$

where $x_{i,k}$ is the classification score of shot i for category k, $\delta(x,y)$ is 1 when $x = y$ and 0 otherwise. Therefore, the log-likelihood becomes

$$\log p(Y|X;\boldsymbol{\lambda},\boldsymbol{\mu}) = \sum_{k \in \mathcal{Y}} \lambda_k \sum_{i=1}^{n} x_{i,k}\delta(y_i, k) + \sum_{\substack{k',k'' \in \mathcal{Y} \\ (k',k'') \neq (c,c)}} \mu_{k',k''} \sum_{i=1}^{n-1} \delta(y_i, k')\delta(y_{i+1}, k'')$$

We take x_i from SVM classifiers trained using cross validation on the training data. The CRF is learned by minimizing the negative log-likelihood in order to estimate $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$.

At test time, the CRF inference outputs marginal conditional probabilities $p(y_i|X), i = 1, \ldots, n$.

4.4. Event localization

The final step consists in localizing instances of an event in a movie, given confidence scores output by the CRF. To that aim, shots must be grouped into segments, and a score must be assigned to the segments. We create segments by joining consecutive shots for which CRF confidence is above 30% of its maximum over the movie. The segment's score is the average of these shot confidences.

	pursuit	battle	romance	victory good	victory bad	battle preparation	despair good	joy bad	good argue bad	good argue good	bad argue bad	mean accuracy
						Leave 4 movies out						
SIFT	53.8	76.4	23.9	11.7	4.4	22.1	15.0	9.5	15.1	25.5	4.0	23.76 ± 5.26
CNN	**66.4**	60.0	16.6	6.0	2.4	9.4	21.7	6.6	17.7	30.2	4.7	21.96 ± 5.91
dense trajectories	58.5	**85.2**	**38.0**	12.7	6.2	**28.0**	19.5	11.6	**18.8**	**40.4**	1.8	**29.15** ± 6.12
MFCC	28.1	56.3	4.5	**17.7**	**36.2**	3.8	**35.4**	**15.6**	17.3	26.5	0.0	21.95 ± 13.97
Face descriptors	47.9	58.1	8.6	12.7	11.4	17.3	9.3	3.2	6.2	22.3	**4.7**	18.35 ± 10.50
linear score combination	63.9	89.2	32.3	14.0	11.4	18.6	26.0	12.1	18.0	44.3	1.8	**30.15** ± 6.72
+ CRF	76.0	91.2	57.6	19.9	1.0	41.4	43.1	9.6	25.1	44.8	0.0	**37.25** ± 9.94
						Leave 1 franchise out						
linear score combination	57.8	83.6	13.0	14.9	9.6	3.8	28.0	5.2	18.2	44.3	0.0	25.32 ± 7.40
+ CRF	75.4	87.4	31.3	15.8	0.0	12.7	33.4	5.7	23.2	43.7	0.0	29.89 ± 12.11

Table 2: *Performance comparison (accuracy) for shot classification. Standard deviations are computed over folds.*

						Leave 4 movies out						
CRF + thresholding	34.6	38.9	22.6	14.6	4.4	26.7	6.4	4.6	12.2	16.9	0.6	16.59 ± 6.82
						Leave 1 franchise out						
CRF + thresholding	36.8	36.5	28.9	14.3	4.5	1.7	4.2	5.2	6.5	13.5	3.7	14.16 ± 6.84

Table 3: *Performance comparison (average precision) for event localization.*

ground truth \ predicted	pursuit	battle	romance	victory good	victory bad	preparation	despair good	joy bad	good argue bad	good argue good	bad argue bad
pursuit	194	92	0	2	2	1	10	0	0	2	0
battle	38	506	1	2	2	2	11	1	3	4	0
romance	4	7	25	3	2	0	24	2	4	15	0
victory good	11	26	1	9	1	1	6	0	1	2	0
victory bad	4	9	0	1	3	0	1	0	1	1	0
preparation	18	38	1	1	2	16	7	0	2	3	0
despair good	23	49	9	5	6	4	44	3	13	24	1
joy bad	3	10	1	2	2	0	9	5	7	5	0
good argue bad	5	14	3	2	6	1	21	2	21	43	0
good argue good	12	18	4	1	3	1	31	2	33	85	1
bad argue bad	2	1	0	0	1	0	2	0	4	6	0

Figure 6: *Confusion matrix for shot classification with SVM and linear score combination for the "leave 4 movies out" setting.*

Note that the CRF produces smoother scores over time for events that occur at a slower rhythm, see Figure 7. For example "good argue good" lasts usually longer than "joy bad", because the villain is delighted for a short time only. The CRF smoothing modulates the length of estimated segments: smoother curves produce longer segments, as expected.

5. Experiments

After validating the processing chain on a standard dataset, we report classification and localization performance.

5.1. Validation of the classification method

To make sure that the descriptor computation and classification chain is reliable, we run it on the Coffee & Cigarettes [LP07] dataset, and compare the results to the state-of-the-art method of Oneata et al. [OVS13]. For this experiment, we score fixed-size segments and use the non-maximum suppression method NMS-RS-0 of [OVS13]. We obtain 65.5 % mAP for the "drinking" action and 45.4 % mAP for "smoking", which is close to the state-of-the-art performance (63.9 % and 50.5 % resp.).

5.2. Shot classification

Table 2 shows the classification performance at the shot-level on the two types of splits. The low-level descriptors that are most useful in this context are the dense trajectories descriptors. Compared to observations made on TrecVid MED or Thumos [OAM*14, JLRZ*14], the relative performance of audio descriptors (MFCC) is high, overall the same as for e.g. CNN. This is because Hollywood action movies have well controlled soundtracks that almost continuously play music: the rhythm and tone of the music indicates the theme of the action occurring on screen. Therefore, the audio descriptors convey high-level information that is relatively easy to detect.

The face descriptor can be seen as a variant of SIFT, restricted to facial regions. The face channel classifier outperforms SIFT in three categories. Upon inspection, we noticed however that only a fraction of shots contain exploitable faces (frontal, non-blurred, non-occluded and large enough), which may explain the lower performance for other categories. The performance of the face classifier may be attributed to a rudimentary facial expression recognition: the faces of heroes arguing with other good characters can be distinguished from the grin of the villain in *joy bad* (Figure 8).

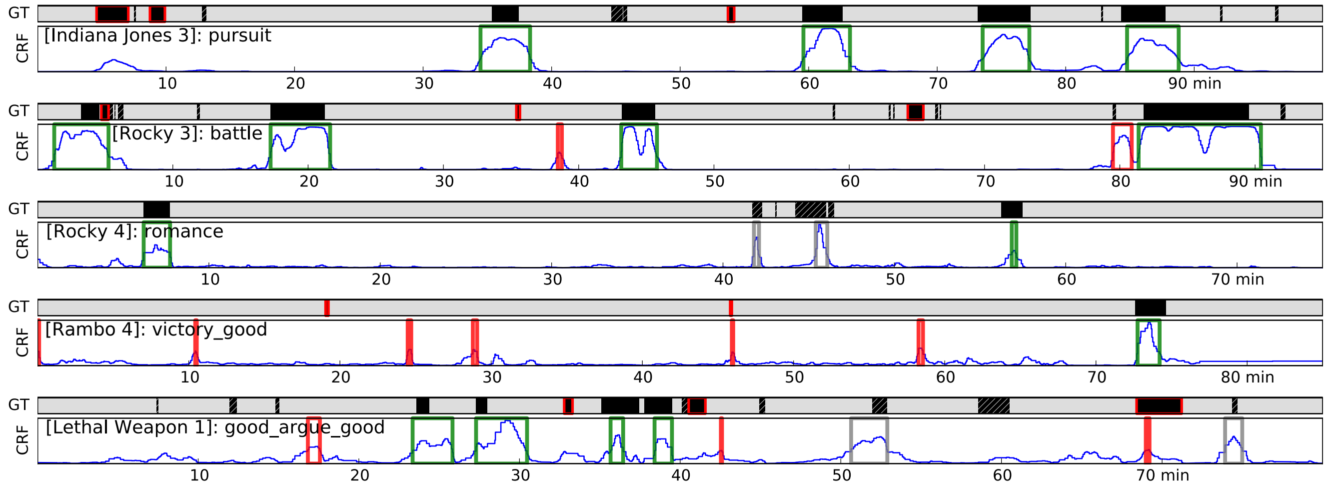

Figure 7: *Example of localization results, for several categories and movies. For each plot, detected events are indicated with bold rectangles (green/gray/red indicate correct/ignored/wrong detections). Ground-truth (GT) annotations are indicated above, missed detections are highlighted in red. Most often, occurrences of the events are rather straightforward to localize given the CRF scores.*

Figure 8: *Sample faces corresponding to shots for which the face classifier (i.e. SVM trained on faces) scored much higher than the SIFT classifier (i.e. trained on full images). Similar facial expressions can be observed within each category, which suggests that our face classifier learns to recognize human expressions to some extent.*

The 4 least ambiguous categories (*pursuit*, *battle*, *battle preparation* and *romance*) are detected most reliably. They account for more than half of the annotated shots. The other categories are typically interactions between people, which are defined by identity and speech rather than motion or music. The confusion matrix in Figure 6 shows that verbal interactions like "good argue good" and "good argue bad" are often confused.

The "leave-4-movies out" setting obtains significantly better results than "Leave-1-franchise out", indicating that having seen movies from a franchise makes it easier to recognize what is happening in a new movie of the franchise: Rambo does not fight in the same way as Rocky, even if the main actor is the same. Finally, the CRF allows to leverage temporal structure using the temporally dense annotations, improving the classification performance by 7 points.

5.3. Event localization

Table 3 gives results for event localization. We observe that the performance is low for the least frequent actions. Indeed, for 8 out of 11 categories, the performance is below 15% AP. Per-channel results are not provided due to lack of space, but their relative performance is similar to the classification ones. Figure 7 displays localization results for different categories. Categories, such as battle and pursuit, are localized reliably. Semantic categories, such as romance, victory good and good argue good are harder to detect. More advanced features could improve the results for these events. Indeed, recognition of characters, though their pose and speech, could potentially improve performance.

6. Conclusion

Despite the explosion of user-generated video content, people are watching professionally produced videos a significant amount of their time. Therefore, the analysis of this kind of footage will remain an important task. Action Movie Franchises serves as a challenging benchmark for this purpose. The annotated classes range from easy to recognize (*battle*) to very difficult as semantic (*good argue bad*). Therefore, we expect our dataset to be valuable for future work on action recognition.

Acknowledgments.

This work was partially supported by projects "Allegro" (ERC), "Khronos" (ANR-11-LABX-0025), the MSR-Inria Joint Centre, and the Moore-Sloan Data Science Environment at NYU.

References

[A13] ALY R., ...: The AXES submissions at TrecVid 2013. In *TRECVID Workshop* (2013). 5

[BBL*13] BOJANOWSKI P., BACH F., LAPTEV I., PONCE J., SCHMID C., SIVIC J.: Finding actors and actions in movies. In *ICCV* (2013). 3

[Bra00] BRADSKI G.: The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000). 6

[CG16] CHEN C.-Y., GRAUMAN K.: Efficient activity detection in untrimmed video with max-subgraph search. *PAMI* (2016). 3

[CJMT08] COUR T., JORDAN C., MILTSAKAKI E., TASKAR B.: Movie/script: Alignment and parsing of video and text transcription. In *ECCV*. 2008. 3

[DJV*13] DONAHUE J., JIA Y., VINYALS O., HOFFMAN J., ZHANG N., TZENG E., DARRELL T.: Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531* (2013). 5

[DPSG14] DEMARTY C.-H., PENET C., SOLEYMANI M., GRAVIER G.: VSD, a public dataset for the detection of violent scenes in movies: design, annotation, analysis and evaluation. *Multimedia Tools and Applications* (2014), 1–26. 3

[ESZ06] EVERINGHAM M., SIVIC J., ZISSERMAN A.: Hello! my name is... Buffy–automatic naming of characters in tv video. 3, 6

[EVGW*10] EVERINGHAM M., VAN GOOL L., WILLIAMS C. K. I., WINN J., ZISSERMAN A.: The Pascal Visual Object Classes (VOC) Challenge. *IJCV 88*, 2 (2010), 303–338. 4, 5

[GHS13] GAIDON A., HARCHAOUI Z., SCHMID C.: Temporal Localization of Actions with Actoms. *PAMI* (2013). 3

[HDlT12] HOAI M., DE LA TORRE F.: Max-margin early event detectors. In *CVPR* (2012). 3

[HLDlT11] HOAI M., LAN Z.-Z., DE LA TORRE F.: Joint segmentation and classification of human actions in video. In *CVPR* (2011). 3

[JLRZ*14] JIANG Y.-G., LIU J., ROSHAN ZAMIR A., TODERICI G., LAPTEV I., SHAH M., SUKTHANKAR R.: THUMOS challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/THUMOS14/, 2014. 7

[KJG*11] KUEHNE H., JHUANG H., GARROTE E., POGGIO T., SERRE T.: Hmdb: a large video database for human motion recognition. In *ICCV* (2011). 3

[KSH12] KRIZHEVSKY A., SUTSKEVER I., HINTON G. E.: ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS* (2012). 5

[LMP01] LAFFERTY J., MCCALLUM A., PEREIRA F. C.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML* (2001). 6

[LNK04] LI Y., NARAYANAN S., KUO C. J.: Content-based movie analysis and indexing based on audiovisual cues. *Circuits and Systems for Video Technology 14*, 8 (2004), 1073–1085. 3

[Low04] LOWE D.: Distinctive image features from scale-invariant keypoints. *IJCV 60*, 2 (2004), 91–110. 5

[LP07] LAPTEV I., PÉREZ P.: Retrieving actions in movies. In *ICCV* (2007). 3, 7

[MLD*06] MASSOUDI A., LEFEBVRE F., DEMARTY C.-H., OISEL L., CHUPEAU B.: A video fingerprint based on visual digest and local fingerprints. In *ICIP* (2006). 3

[MLS09] MARSZALEK M., LAPTEV I., SCHMID C.: Actions in context. In *CVPR* (2009). 3

[OAM*14] OVER P., AWAD G., MICHEL M., FISCUS J., SANDERS G., KRAAIJ W., SMEATON A., QUÉNOT G.: Trecvid 2014 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2014* (2014), NIST, USA. 1, 3, 7

[OVS13] ONEATA D., VERBEEK J., SCHMID C.: Action and Event Recognition with Fisher Vectors on a Compact Feature Set. In *ICCV* (2013). 1, 5, 7

[OVS14] ONEATA D., VERBEEK J., SCHMID C.: The LEAR submission at Thumos 2014. In *ECCV 2014 Workshop on the THUMOS Challenge 2014* (Zurich, Switzerland, Sept. 2014). 1

[PDHS14] POTAPOV D., DOUZE M., HARCHAOUI Z., SCHMID C.: Category-specific video summarization. In *ECCV* (2014). 3

[PJM10] PERRONNIN F., J.SÁNCHEZ, MENSINK T.: Improving the Fisher kernel for large-scale image classification. In *ECCV* (2010). 5

[RHM98] RUI Y., HUANG T. S., MEHROTRA S.: Exploring video structure beyond the shots. In *Multimedia Computing and Systems* (1998), pp. 237–240. 3

[RS07] RABINER L. R., SCHAFER R. W.: Introduction to digital speech processing. *Foundations and trends in signal processing 1*, 1 (2007), 1–194. 5

[RS13] RAPTIS M., SIGAL L.: Poselet key-framing: A model for human activity recognition. In *CVPR* (2013). 3

[RTT03] RONFARD R., TRAN-THUONG T.: A framework for aligning and indexing movies with their script. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)* (Baltimore, MD, United States, July 2003). 1

[Sny05] SNYDER B.: *Save the cat! The last book on screenwriting you'll ever need*. Michael Wiese Productions, 2005. 1, 4, 5

[SPVZ13] SIMONYAN K., PARKHI O. M., VEDALDI A., ZISSERMAN A.: Fisher Vector Faces in the Wild. In *BMVC* (2013). 6

[SZS12] SOOMRO K., ZAMIR A. R., SHAH M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012). 3

[WBTG09] WILLEMS G., BECKER J., TUYTELAARS T., GOOL L.: Exemplar-based action recognition in video. In *BMVC* (2009). 3

[WS13] WANG H., SCHMID C.: Action recognition with improved trajectories. In *ICCV* (2013). 1, 3, 5

[YGG10] YAO A., GALL J., GOOL L. V.: A hough transform-based voting framework for action recognition. In *CVPR* (2010). 3