2011 Eighth International Conference Computer Graphics, Imaging and Visualization

Violence Detection in Movies

Liang-Hua Chen, Hsi-Wen Hsu
and Li-Yun Wang
Department of Computer Science
and Information Engineering
Fu Jen University, Taipei, Taiwan

Chih-Wen Su

Departmnet of Information and
Computer Engineering
Chung Yuan University, Chung-Li, Taiwan

Abstract

As violence in movies has harmful influence on children, in this paper, we propose an algorithm to detect violent scene in movies. Under our definition of violence, the task of violent scene detection is decomposed into action scene detection and bloody frame detection. While previous approaches addressed on shot level of video structure only, our approach works on more semantic-complete scene structure of video. The input video (digital movie) is first segmented into several scenes. Based on the film-making characteristics of action scene, some features of the scene are extracted to feed into the support vector machine for classification. Finally, the face, blood and motion information are integrated to determine whether the action scene has violent content. Experimental results show that the proposed approach works reasonably well in detecting most of the violent scenes. Compared with related work, our approach is computationally simple yet effective.

1 Introduction

The advances in low cost mass storage devices, higher transmission rates and improved compression techniques, have led to the widespread use and availability of digital video. Nowadays, everyone can download movies easily using home computer. However, violence in movies has harmful influence on children. It was reported that children who liked to watch violent TV programs when they were 8 years old were more likely to behave aggressively at age 18[1]. To prevent children from watching violent movies, the automatic detection of inappropriate violence in movies is of substantial importance. For content provider, the violence detection technique can be used to assist in movie-rating; for end user, it can block the violent content in client terminal devices. On the other hand, violent scenes attract attention and make viewers curious. They are usually the highlights of a movie. Therefore, violence detection would also be useful for movie skimming.

In this paper, we propose an empirically motivated approach for violence detection in movies. The task of violent scene detection is decomposed into action scene detection and bloody frame detection. Our approach is based on the integration of visual characteristics and temporal dynamics information of video. The rest of this paper is organized as follows. In the next section, we review some related works and give the motivation for our approach. An action scene detection algorithm is presented in Section 3. In Section 4, we describe how to integrate several visual features to detect violent content. The performance evaluation of our approach is reported in Section 5. Finally, some concluding remarks are given in Section 6.

2 Background and Motivation

Relatively few approaches have been proposed to the problem of violent scene detection in video. The main reason is that the definition of violence is ambiguous. It is difficult to describe this high-level concept using mathematical formulation precisely. Each related work addressed the problem by its own definition of violence. Depending on the type of video features, current techniques for violence detection can be broadly classified into three categories. The first one is based on visual cue. Using motion trajectory information and orientation information of a person's limbs, Datta et al. addressed the problem of detecting human violence in video such as fist fighting and kicking[2]. Their approach relies on the extraction of silhouette of each person from the image. Thus it works well only in presence of two persons. Mecocci and Micheli proposed to use maximum warping energy as criterion to detect violent acts among more people in crowded environments[3]. But, it is still difficult to differentiate fighting from basketball playing using this approach. It is also noted that both approaches ([2, 3]) use video data from surveillance cameras and are not suitable for movies which have large camera movement. The second category is the audio based approach. Giannakopoulos et al. used eight audio features, both from the time and frequency domain, as input to a binary classifier which decides the video content with respect to violence[4]. Then, they extended thir work to multi-

978-0-7695-4484-7/11 $26.00 © 2011 IEEE
DOI 10.1109/CGIV.2011.14

class classification problem using Bayesian networks[5]. The video content is divided into six classes. Three classes are violent: shots, fights and screams. However, their approaches assume that the audio signal has already been segmented into semantically coherent *segments*. Moreover, satisfying results may not be obtained from video with a sound track containing more than just speech (such as music and environmental sound), or video clip which is silent. In the third category, emphasis is put on the integration of visual and audio features[6, 7, 8, 9]. While different types of features may complement each other, they sometimes provide inconsistent information. Besides, it is always computationally expensive.

As the definition of violence is ambiguous, violence detection is a subjective process. In this work, we define violence from an empirical viewpoint. Our definition of violence is: a series of human actions accompanying with bleeding. Thus, the task of violence scene detection can be decomposed into two processes: action scene detection and bloody frame detection. Before describing the details of the proposed approach, we review the hierarchical structure of video. A video is physically formed by shots and semantically described by scenes. A shot is a sequence of frames that was continuously captured by the same camera. A scene is basically a story unit and consists of a small number of interrelated shots that are consecutive or not. Shots in video are analogous to words in language as they convey little semantic information in isolation. On the other hand, scenes allow event to be fully presented and reflect the dramatic and narrative structure of a video. It is noted that previous approaches on violence detection only work on shot level and they do not fully exploit the information contained in video structure. In this paper, we propose a novel technique based on the more semantic-complete scene structure of video. To be simple yet effective, only visual features of video are used. Support vector machine (SVM) is also employed to classify the extracted scenes into action scenes and non-action scenes. The proposed approach is made up of four main steps
(1) Segmentation of video into shots.
(2) Grouping of shots into scene.
(3) Action scene detection using SVM.
(4) Bloody frame detection.
where Steps (1) and (2) are accomplished by our early work[10]. The details of Steps (3) and (4) are described in Sections 3 and 4 respectively.

3 Action Scene Detection Using SVM

For each scene of the video sequence, some features are extracted. Then, a binary classifier (SVM) is employed to detect the action scene[11]. The details are described in the following two subsections.

3.1 Overview of support vector machines

Unlike traditional learning techniques such as neural networks which minimize the empirical training error, the support vector machines (SVMs) are based on the structural risk minimization principle. The basic idea is closely related to regularization[12]: for a finite set of training samples, the search for the best model or approximating function has to be constrained by an appropriately small hypothesis space. If the space is too large, functions can be found which fit exactly the training data, but they will have poor generalization capabilities on new test data. Instead, the minimization of the structural risk is equivalent to minimizing the sum of the error on the training set and the complexity of the hypothesis space, expressed in terms of VC-dimension. Consequently, the solutions obtained with SVMs are more likely to generalize well on new data points. We now go through the SVMs for a two-class classification problem. For a comprehensive and rigorous account of SVMs, please see[13].

Given a training set $S = \{(x_1, y_1), \cdots, (x_n, y_n)\}$, where feature vector $x_i \in R^d$ and label $y_i \in \{1, -1\}$, the goal of SVM is to construct a hyperplane that maximizes the margin while minimizing a quantity proportional to the misclassification error. The optimal separating hyperplane $w^* \cdot x + b^* = 0$ can be found under the following constraints:

$$\min_{w,b,\xi} \quad \frac{1}{2} w \cdot w + C \sum_{i=1}^{n} \xi_i$$

subject to

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \cdots, n$$

where C is the penalty parameter that controls the trade-off between the margin and the misclassification errors $\xi = (\xi_1, \cdots, \xi_n)$. Here training vectors x_i are mapped into a higher dimensional space by the function ϕ, and $K(x_i, x_j) \equiv \phi(x_i) \cdot \phi(x_j)$ is called the kernel function. This is a quadratic programming problem which can be solved by standard technique such as Lagrange multipliers.

In the SVM training process, 5-fold cross-validation is used to prevent the overfitting problem. The radial basis function is chosen as the kernel function of SVM:

$$K(x, y) = e^{-\gamma ||x-y||^2}$$

Finally, a grid-search procedure is employed to select the parameters of SVM[14].

3.2 Feature extraction

To extract proper features for classification, we consider the following film production rules[15]:

- In action scenes, the filmmaker often uses a series of shots with high motion activity to create tense and strong atmosphere.

- Fast edits are frequently used to build a sense of kinetic action and speed.
- Two action scenes with high film rhythm may not be juxtaposed together.

Using these guideline, the director and editor control the pace of a movie to grasp the attention of the viewers. Thus, most of the action scenes consist of a consecutive sequence of short shots with high motion activity. This type of video sequences will provide a lot of rapidly changing visual information displayed on screen to excite the viewers. Based on these action scene characteristics, the following features are extracted.
(1) Average Motion Intensity:
Motion is a visual feature which is essential to capture temporal variation of video. It also reveals the correlations between frame sequences within a video scene. To characterize the degree of motion within a scene, the *average motion intensity* is computed based on the motion vectors encoded in the MPEG-1 video stream[16]. In MPEG video, each frame is partitioned into blocks of size 16×16 pixels called macro blocks (MBs). MPEG defines motion vector as the displacement from the Target (current frame) MB to the Prediction (reference frame) MB. In MPEG format, there are three types of frames: I, P and B frames. I frames are skipped because they are intra-coded and no motion information is available. P frames have forward motion prediction and B frames have both forward and backward motion prediction. In our system, only the forward motion vectors encoded in P frames are extracted. For a given P frame, the motion intensity matrix is defined as

$$M(i,j) = \sqrt{u_{i,j}^2 + v_{i,j}^2}$$

where $(u_{i,j}, v_{i,j})$ is the motion vector associated with (i,j)th macroblock. Assuming there are $m \times n$ macroblocks in the frame, then the average motion intensity of the frame is

$$\overline{M} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} M(i,j)$$

Then, the average value of $\overline{M}$ over all frames within a scene is obtained. Finally, this value is normalized to be in the interval $[0,1]$.
(2) Camera Motion Ratio:
If a frame has less than 10% motion vectors to be zero, then this frame has camera motion. Assuming a scene S consists of m frames, and k frames have camera motion. The camera motion ratio is defined as

$$C = \frac{k}{m}$$

(3) Average Shot Length:
Assuming a scene S consists of n shots, and their corresponding shot length is $L_i, i = 1, \cdots, n$. The average shot length of S is defined as

$$\overline{L} = \frac{1}{n} \sum_{i=1}^{n} L_i$$

Likewise, $\overline{L}$ is normalized to be in the interval $[0,1]$.
(4) Shot Cut Frequency:
Assuming a scene S consists of n shots, then the shot cut frequency of S is defined as

$$F = \frac{1}{n}$$

Thus, a 4-dimensional feature vector for classification is constructed.

4 Bloody Frame Detection

For each detected action scene, we further check whether it contains the bloody frame. As scene is composed of several shots, key frames can be extracted from each shot. Key frame is the frame which can represent the salient content of the shot. Currently we choose the middle frame of each shot as key frame. Then, we determine whether one of these key frames has bloody content. We first identify the existence of human and blood in the key frame.

Face detection is the natural and convenient way to determine whether human appear in the frame image. We adopt the excellent face detection algorithm proposed by Viola and Jones[17]. This algorithm is capable of processing images extremely rapidly while achieving high detection rates. Figure 1 shows some face detection results.

There are some sophisticated techniques to detect specific color object such as flame and skin. However, for the blood pixel detection task, the easiest method is to define blood-color cluster decision boundaries for RGB color space. Ranges of threshold values for each color space component are defined and the image pixel with values that fall within these predefined ranges is detected as blood pixel. According to our observation, there are two types of blood color: bright red and dark red. The ranges of RGB color space are defined as (1) $170 > R > 80$ & $G < 5$ & $B < 5$ (2) $200 > R > 120$ & $G < 90$ & $B < 90$ & $abs(G - B) < 8$, respectively. Figure 2 shows the blood detection result.

Then, a connected components analysis is performed to group these detected blood pixels into several potential blood regions. A size filter is used to eliminate some potential blood regions which are too small or too large. Small region may result from noise and large region may correspond to some large object such as red wall. As we observe from violent movies that blood and human body always come up with fast motion, we integrate blood, face and motion informations to determine whether a key frame is a bloody frame. A key frame is detected as bloody frame if it satisfies the following conditions:

- It contains both face and blood regions.
- The distance between face and blood regions is less than a threshold.
- The average motion intensity of each blood and face region is greater than a threshold.

If an action scene contains at least one bloody frame then it is detected as a violent scene.

5 Experimental Results

Four movies in MPEG-1 format are used in our experiment: (1)"Kill Bill: Vol. 1", (2)"Gladiator", (3)"The Passion of the Christ" and (4)"First Blood 4: John Rambo". The ground truth of test movies, i.e., the decision whether a given scene is violent scene or not, is determined by human subjects. Some violent scenes of movies are shown in Figure 3.

The experimental results are shown in Table 1. The missed detection is due to two factors. One is that the actor(actress) wears mask so that the face detection algorithm fails. The other is that the face part of actor(actress) does not appear in the key frame. Although there is no false detection in our experiment, one scenario that causes false detection is a group of players in red clothing playing basketball.

The performance of violent scene detection is usually measured by the following two metrics:

$$\text{Recall} = \frac{D}{D + MD} \qquad \text{Precision} = \frac{D}{D + FD}$$

where D is the number of violent scenes detected correctly, MD is the number of missed detection and FD is the number of false detection. For performance comparison, we also implement the algorithm proposed by Lin et al.[8]. The reason for choosing Lin's work is that their definition of violence is very close to ours. To compare both approaches fairly, the parameters of Lin's algorithm are tuned to achieve the best performance. As shown in Table 2, our approach is, overall, better than Lin's approach in term of recall and precision.

6 Conclusion

We have presented an effective method for automatically detecting violent scene in the digital movies. While previous approaches addressed on shot level of video structure only, our approach works on more semantic-complete scene structure of video. Thus, more spatiotemporal information of video is exploited for analysis. Based on some features extracted from scene structure, a support vector machine is employed to detect action scene. Then, face, blood and motion informations are integrated to determine whether the action scene has violent content. Experimental results show that the proposed approach works reasonably well in detecting most of the violent scenes. Compared with the related work, our approach is promising. As the definition of violence is ambiguous, in the worst case setting, violence detection is an unsolvable problem. However, on average, there is still hope in determining an empirical basis for violence detection. The proposed approach takes a stride toward this difficult problem. Our approach can be applied directly to video abstraction and can be utilized to support high-level video indexing in movie databases.

References

[1] University of Pittsburgh Office of Child Development. TV and movie violence. http://www.education.pitt.edu/ocd/publications /parentingguides/TvAndMovieViolence.pdf.

[2] A. Datta, M. Shah, and N.D.V. Lobo. Person-on-person violence detection in video data. In *Proceedings of International Conference on Pattern Recognition*, pages 433–438, Quebec, Canada, August 2002.

[3] A. Mecocci and F. Micheli. Real-time automatic detection of violent-acts by low-level colour visual cues. In *Proceedings of International Conference on Image Processing*, pages 345–348, San Antonio, TX, October 2007.

[4] T. Giannakopoulos et al. Violence content classification using audio features. In *Proceedings of the 4th Hellenic Conference on Artificial Intelligence*, pages 502–507, Crete, Greece, May 2006.

[5] T. Giannakopoulos, A. Pikrakis, and S. Theodoridis. A multi-class audio classification method with respect to violent content in movies using bayesian networks. In *Proceedings of IEEE International Workshop on Multimedia Signal Processing*, pages 90–93, Crete, Greece, October 2007.

[6] J. Nam, M. Alghoniemy, and A.H. Tewfik. Audio-visual content-based violent scene characterization. In *Proceedings of International Conference on Image Processing*, pages 353–357, Chicago, 1998.

[7] Y. Gong et al. Detecting violent scenes in movies by auditory and visual cues. In *Proceedings of the 9th Pacific Rim Conference on Multimedia*, pages 317–326, Tainan, Taiwan, December 2008.

Table 1: Accuracy measures for four test movies.

Movie ID No.	Number of Violent Scenes	Correct Detection	Missed Detection	False Detection
(1)	17	14	3	0
(2)	10	8	2	0
(3)	18	17	1	0
(4)	12	10	2	0

Table 2: Performance comparison for violent scene detection.

Movie	Our Approach		Lin's Approach	
ID No.	Recall	Precision	Recall	Precision
(1)	82.35%	100%	82.35%	82.35%
(2)	80.00%	100%	70.00%	87.50%
(3)	94.44%	100%	88.89%	80.00%
(4)	83.33%	100%	75.00%	81.82%

[8] J. Lin and W. Wang. Weakly-supervised violence detection in movies with audio and video based co-training. In *Proceedings of the 10th Pacific Rim Conference on Multimedia*, pages 930–935, Bangkok, Thailand, December 2009.

[9] T. Giannakopoulos et al. Audio-visual fusion for detecting violent scenes in videos. In *Proceedings of the 6th Hellenic Conference on Artificial Intelligence*, pages 91–100, Athens, Greece, May 2010.

[10] L.-H. Chen, Y.-C. Lai, and H.-Y. liao. Movie scene segmentation using background information. *Pattern Recognition*, 41(3):1056–1065, March 2008.

[11] L.-H. Chen, C.-W. Su, C.-F. Weng, and H.-Y. Liao. SVM based action scene detection. In *Poster Proceedings of the 6th International Conference on Machine Learning and Data Mining*, pages 45–50, Leipzig, Germany, July 2009.

[12] T. Evgenious, M. Pontil, and T. Poggio. A unified framework for regularization networks and support vector machines. A.I. Memo 1654, MIT, Cambridge, MA, 1999.

[13] V.N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[14] C.W. Hsu, C.C. Chang, and C.J. Lin. A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University, Taipei,Taiwan, 2003.

[15] D. Arijon. *Grammar of the Film Language*. Silman-James Press, Los Angels, 1991.

[16] D.L. Gall. MPEG: A video compression standard for multimedia applications. *Communication of ACM*, 34(4):46–58, April 1991.

[17] P. Viola and M.J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, May 2004.

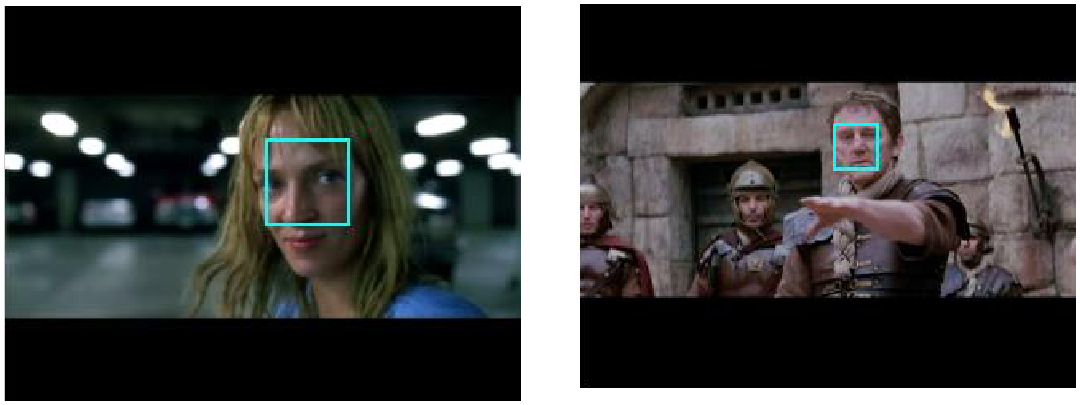
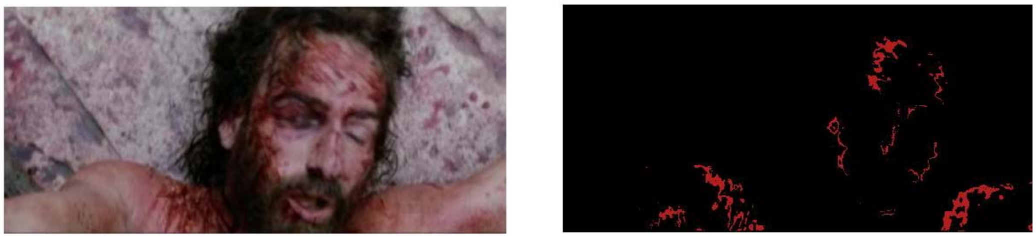


Figure 1: Some face detection results.



(a) (b)

Figure 2: (a) is the original image (b) is the detected blood pixels.

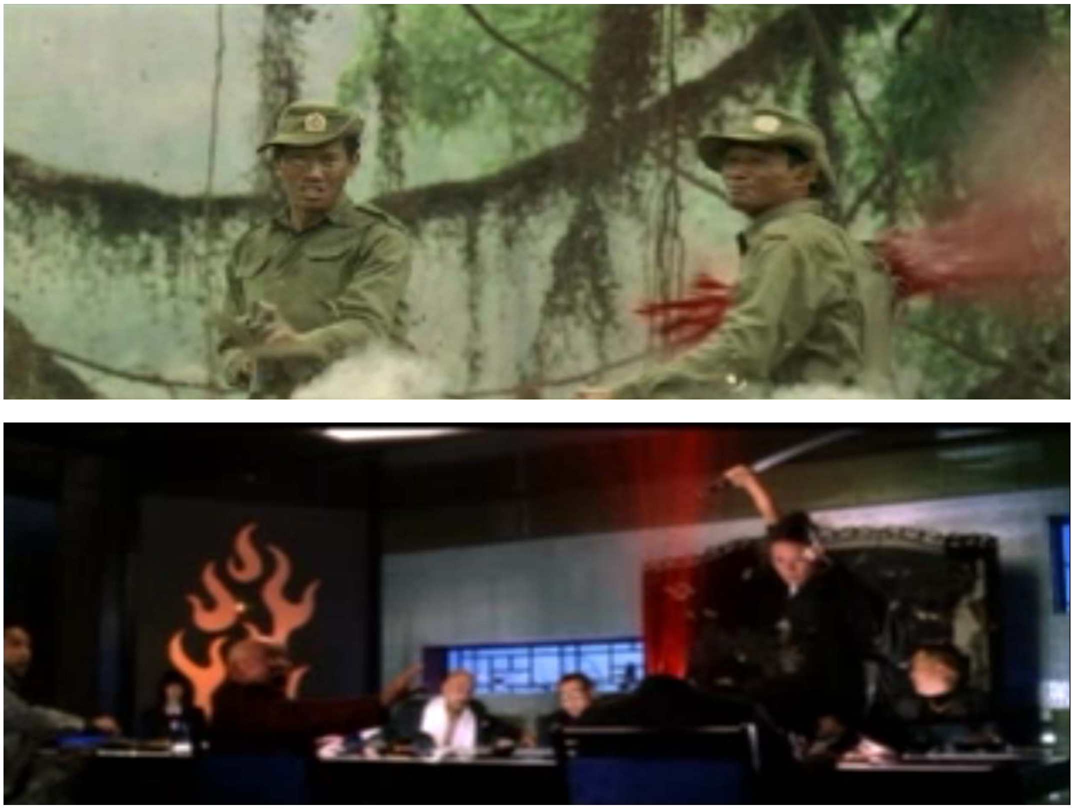


Figure 3: Some violent scenes.