Bloody Image Classification with Global and Local Features

Song-Lu Chen, Chun Yang, Chao Zhu, and Xu-Cheng Yin(✉)

School of Computer and Communication Engineering,
University of Science and Technology Beijing, Beijing, China
chensonglu@xs.ustb.edu.cn, ych.learning@gmail.com,
{chaozhu,xuchengyin}@ustb.edu.cn

Abstract. Object content understanding in images and videos draws more and more attention nowadays. However, only few existing methods have addressed the problem of bloody scene detection in images. Along with the widespread popularity of the Internet, violent contents have affected our daily life. In this paper, we propose region-based techniques to identify a color image being bloody or not. Firstly, we have established a new dataset containing 25431 bloody images and 25431 non-bloody images. These annotated images are derived from the Violent Scenes Dataset, a public shared dataset for violent scenes detection in Hollywood movies and web videos. Secondly, we design a bloody image classification method with global visual features using Support Vector Machines. Thirdly, we also construct a novel bloody region identification approach using Convolutional Neural Networks. Finally, comparative experiments show that bloody image classification with local features is more effective.

Keywords: Bloody image classification · Violent scenes dataset · Support vector machines · Convolutional Neural Networks

1 Introduction

As we enjoy the fast and convenient access to all kinds of information via the Internet, we are also exposed to undesirable contents, such as violent images, gory videos and pornographic information. The flourishing images sharing websites like Instagram, social websites like Facebook and Twitter, search engines like Google and videos sharing websites like Youtube even enable obtaining and sharing harmful information as simple as sliding on the screen or clicking the mouse. For those who are deficient of good sense of judgement and self-control, especially teenagers, long-term exposure to such negative information can seriously affect them both physically and mentally, even may cause aggressive behaviour or crimes. For this, it's desperately needed to limit the propagation of such unhealthy contents. In this paper, we propose a fairly effective method aimed on bloody image identification.

T. Tan et al. (Eds.): CCPR 2016, Part II, CCIS 663, pp. 379–391, 2016.
DOI: 10.1007/978-981-10-3005-5_31

Compared with pornographic contents, the investigations of violence detection is still in the initial stage. Recognizing the bloody images can be considered as one category of violence detection. Following are several related works. Wang et al. [1] propose to detect suspected bloodstain in YC_gC_r color space. To accelerate bloodstain pixels detection, the transformed thresholds $(C_{g'}, C_{r'})$ are used. This approach can be used as the basis for bloodstain identification in color images, but it also works for pseudo-bloody images like red apple, red clothes etc. This method by nature is based on the idea of Dios et al. [2,3]. Yoo et al. [4] propose an efficient red-eye removal algorithm which uses an inpainting method and biometric information. Yan et al. [5] extract global features as well as local features from the detected bloody regions in an image. These features are fed into the SVM to classify. Wang et al. [6] adopt the Bag-of-Words model to discriminate violence images and non-violence images. They combine four global image features with the BoW framework for all categories of violence. Li et al. [7] propose a novel horror image recognition based on emotional attention mechanism. Guermazi et al. [8] focus on the color descriptors to recognize violent images. They utilize the MPEG7 color descriptors. Lopes et al. [9] add color information to original SIFT features to detect nude or pornographic images. This method does not rely on any skin or shape models and has considerable improvement than traditional SIFT features. Ulges et al. [10] present a visual recognition system for the detection of child pornographic image material, based on color-enhanced visual word features and SVM classifier. Violence Scenes Dataset (VSD) [11] is a public shared dataset for the detection of violent scenes in Hollywood movies and web videos. The detail of VSD has been described by Demarty et al. [12] and Schedi et al. [13]. The rest [14–19] focus on either video violence detection or audio violence detection or both.

Although there have been plenty of investigations for content-based image classification, very few of them attempt to deal with the issue of violent image detection. A big challenge is how to define a violent image. In our opinion, a violent image should be a picture which can trigger negative emotions such as panic, menace, anxiety and aggressiveness. So the range of violent images should not just include images with gory scenes, but also those embody gunshots, car chases, explosions, fire, fights as well. Anyway, ambiguity certainly exists due to subjectivity in discriminating violent images. For simplicity, in this paper we only concern about bloody image detection which means the image must contain one or more blocks of bloodstain. We define that if the proportion of blood pixels in a color image is greater than 0.5 %[1], it's bloody, otherwise, it is not. This definition can make sure a bloody image objectively instead of intuitive judgement. Thus, the issue will be converted into the following steps: segmenting all suspected bloodstain regions, filtering out all real bloodstain regions, adding up the area of all blood pixels and determining if the image being bloody or not.

Another challenge is that unlike violence detection in videos where various visual, audio and motion features can be combined together to solve the problem,

[1] We compared all bloody images and non-bloody images in our dataset and found this number is a quite reasonable threshold to distinguish these two kind of images.

detecting violence in still images can only depend on visual features. In addition, the detection task becomes rather complicated due to the wide range of scenes, backgrounds and luminance, furthermore, pseudo-bloody regions make the task more challenging. To this end, we have established a dataset of a large number of bloody and non-bloody images, which contains the samples of all aforementioned types. Then we propose a region-based method for bloody image classification because we believe that local information is more representative of the content of an image. Finally, we utilize a convolutional neural network (CNN) to classify all red-like regions. With CNN, a large number of features can be extracted automatically rather than collecting complicated color, texture or shape features manually, furthermore, we can constantly enhance the performance of CNN model based on reinforcement learning.

The remainder of this paper is organized as follows: Sect. 2 introduces the new established dataset. Section 3 presents our approach for bloody image identification from both global and local views. Section 4 shows the experimental results as well as corresponding analysis. Section 5 concludes this paper.

2 Dataset

An appropriate dataset is essential to evaluate the effectiveness of our classification method. Fortunately, there is a public shared dataset named Violent Scenes Dataset (VSD) [11] for the detection of violent scenes, which contains many Hollywood movies, web videos and corresponding annotations. These annotations mainly consist of high-level concepts for Hollywood movies. More specifically, six visual concepts and three audio concepts are annotated. They are presence of blood, fights, fire, cold arms, car chases and gory scenes for the visual modality as well as presence of gunshots, explosions and screams for the audio modality.

The VSD provides high-level concepts annotations for seventeen movies, but eventually only five representative Hollywood movies have been picked out as samples for our classification task. Sorted by contribution, they are in turn *Reservoir Dogs*, *Saving Private Ryan*, *Harry Potter 5*, *Armageddon* and *Pirates of the Caribbean 1*. Some downloaded movies over the Internet are not corresponding to the annotations, such as *Leon* and *I am Legend*, and some such as *Billy Elliot* and *Independence Day* lack bloody scenes, or even have no bloody frames, and others are too dark to be used as samples, like *Fight Club*.

Even the movies selected as samples still need further processing. Firstly, these movies all have frame offset ranged from several frames to hundreds of frames, and we can only adjust them according to movies scenes containing blood, car chases and fights etc. Secondly, official released annotations are spot-based, and each spot may contain hundreds of frames. Inevitably, there exist several non-bloody frames in some spots. Moreover, some parts of the annotations are overlapped with other parts, and some are even wrongly annotated. We have modified many frame-wise annotations manually and that's a time-consuming task. Thirdly, we extracted all sample movies to images by frame and removed the black edge of each image. Finally, we conducted many different kinds of classification experiments using features extracted from all sample movies and compared them with official released features to make sure all movies are usable. Eventually, we got five usable movies from VSD by making large adjustments of them.

In practice, the number of non-bloody images is far more than bloody images. For sample balance, we randomly selected equal number of non-bloody images from each movie. According to corresponding annotations, we collected 50862 images including 25431 bloody images and 25431 non-bloody images. For convenience, we named the dataset as PVSD (Part of Violent Scenes Dataset). They are all color images in PNG format with height larger than 500 pixels and width larger than 1000 pixels. As time goes on, we will enrich our dataset constantly with images from different sources, such as digital images, scanned photographs, software edited images, and images with large range of luminance or resolution. Figure 1 demonstrates samples of some bloody images in PVSD.

Fig. 1. Samples of some bloody images in PVSD

In addition, we have also established our training set and test set based on PVSD. Firstly, we randomly pick out 10 % images from each sample movie as training set and the other 90 % as test set. In order to distinguish them with PVSD, we called the training set TSGF (Training Set for Global Features), and named the test set TID (Test Image Dataset). TSGF contains 5088 images with 2544 bloody images and 2544 non-bloody images. TID contains 45774 images with 22887 bloody images and 22887 non-bloody images. Besides, we have established our training set for local features based on TSGF, which contains 12416 images covered with bloodstain and 11627 images covered with pseudo-blood after manual sortation. In order to test the effect of different magnitude of training set, we have set up five datasets called TSLF-1X (Training Set for Local Features 1X), TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X respectively. These five sets all contain images centered with bloodstain or pseudo-blood, but the differences lie in that the images in TSLF-1X almost exclusively contain bloodstain or pseudo-blood while images in other four sets contain more surrounding background, expanding 2 to 5 times in height and width respectively. In summary, TSGF is the training set for global features, while TSLF-1X, TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X are training sets for local features. TID is the test set for both global and local features. Figure 2 shows samples of some images

Fig. 2. Samples of some images covered with blood and pseudo-blood: the first three columns of the first row, bloody images in TSLF-1X; and the last three columns of the first row, pseudo-bloody images in TSLF-1X; and the next four rows for TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X respectively

covered with bloodstain and pseudo-blood in TSLF-1X, TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X.

3 Bloody Image Classification

In this section, the detection task can be divided into two parts: classification with global visual features as well as bloodstain segmentation and classification with local features. In the first part, three visual features extracted from images as well as their fusion are fed into SVM to classify. In the second part, we first segment all red-like regions in an image and filter out all real bloody ones using Deep Learning method.

3.1 Classification with Global Visual Features

In our view, bloodstain has distinguishing features compared with other objects, such as color, appearance and texture. Normally, bloodstains are red and scattered like irregular clusters or bands. For example, blood droplets always have little or no distortion on their peripheral edges and change smoothly from the center, while splashed blood varies largely in luminance and shape. Based on this, we apply three visual features for bloody image classification. These three features are briefly introduced as follows:

Color moments (CM), a measure to characterise color distribution in an image, contain the first three central moments of an image color distribution: mean, standard deviation and skewness.

The first color moment can be interpreted as the average color in the image, and it can be calculated by the following formula:

$$E_i = \sum_{j=1}^{N} \frac{1}{N} p_{ij} \tag{1}$$

where N is the number of pixels in the image and p_{ij} is the value of the j-th pixel of the image at the i-th color channel.

The second color moment is the standard deviation, which is obtained by taking the square root of the variance of the color distribution.

$$\sigma_i = \sqrt{(\frac{1}{N} \sum_{j=1}^{N} (p_{ij} - E_i)^2)} \tag{2}$$

where E_i is the mean value, or first color moment, for the i-th color channel of the image.

The third color moment is the skewness. It measures how asymmetric the color distribution is, and thus it gives information about the shape of the color distribution. Skewness can be computed with the following formula:

$$s_i = \sqrt[3]{(\frac{1}{N} \sum_{j=1}^{N} (p_{ij} - E_i)^3)} \tag{3}$$

Histogram of Oriented Gradients (HOG) can exploit the local object appearance and shape within an image via the distribution of edge orientations. Local Binary Patterns (LBP) is a type of texture descriptor used for classification. In some detection cases [20], LBP is often used in combination with HOG to improve the detection performance. These three visual features can be used together in our classification task.

According to the introduction document of VSD [13], VSD also extracted these three visual features for all movies besides corresponding annotations. For convenience, we apply official released CM, HOG and LBP features as well as their fusion into our task. The CM and HOG features are all 81-dimensional, and the LBP is 144-dimensional. During the classification stage, SVM with RBF kernel is chosen as the classifier. LIBSVM [21], a library for SVM, makes our experiments much easier. For the TSGF, we randomly select 60 % of the images as training set while the rest as validation set and TID is used as test set. After normalization, different combination of features are separately fed into SVM to classify.

3.2 Detection and Classification with Local Features

In this section, we first determine the color thresholds of bloodstain pixels, then generate TSLF-1X, TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X using these thresholds based on TSGF, and finally conduct classification with Deep Learning method.

Bloodstain Segmentation. According to Wang et al. [1], their approach can be used as the basis for bloodstain segmentation. Figure 3 shows outline of their algorithm. However, based on our actual conditions, we have made some adjustments, such as, no color histogram equalization for color images, no erosion and dilation to smooth binary images and effective cluster areas being raised to 400 pixels.

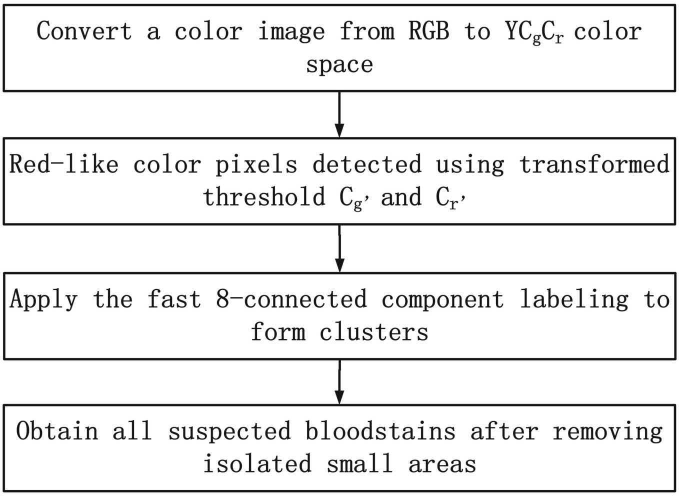

Fig. 3. The flowchart of red-like region segmentation

The YC_gC_r color space can be easily transformed from RGB color space through the following formula.

$$\begin{bmatrix} Y \\ Cg \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -81.085 & 112 & -30.915 \\ 112 & -93.768 & -18.214 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{4}$$

Four hundred images randomly selected from TSLF-1X are used as training set to simulate the bloodstain color. Figure 4 illustrates the bloodstain color distribution in the $C_g - C_r$ plane, and the bloodstain color region is concentrated in an inclined stripe area.

On the basis of Wang et al. [1], the color stripe should be rotated 36° clockwise to a transformed $C_{g'} - C_{r'}$ plane to accelerate the bloodstain pixels detection. The new color space $YC_{g'}C_{r'}$ can be transformed by the following equations.

$$C_{g'} = C_g \times cos36^\circ + C_r \times sin36^\circ \tag{5}$$

$$C_{r'} = -C_g \times sin36^\circ + C_r \times cos36^\circ \tag{6}$$

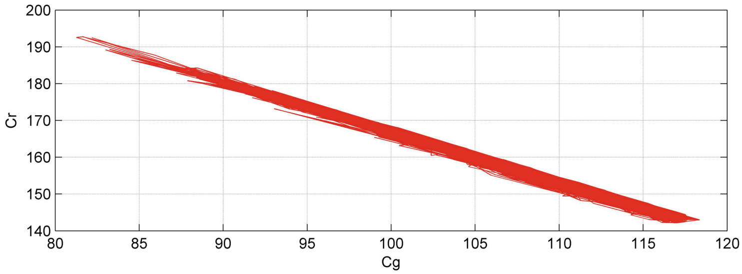

Fig. 4. Bloodstain color distribution in the $C_g - C_r$ plane (Color figure online)

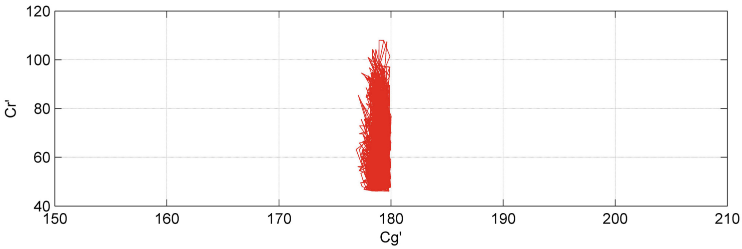

Fig. 5. Bloodstain color distribution in the $C_{g'} - C_{r'}$ plane (Color figure online)

The new bloodstain area is shown in Fig. 5, and it's perpendicular to $C_{g'}$ axis. Undoubtedly, it will make the bloodstain detection more accurate and fast.

The decision thresholds for bloodstain segmentation are determined by the minimum and maximum of $C_{g'}$ and $C_{r'}$. In contrast with Wang et al. [1], our decision thresholds are [176, 180] for $C_{g'}$ and [46, 109] for $C_{r'}$.

Classification with Local Features Using Deep Learning. So far, we have acquired the thresholds of bloodstain pixels in the transformed $YC_{g'}C_{r'}$ color space. According to the method of red-like region segmentation, we get 24043 images covered with bloodstain or pseudo-blood from TSGF and separate them into bloody and non-bloody manually[2]. By this way, we have established TSLF-1X, TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X, and images in these five datasets are all centered with bloodstain or pseudo-blood but differ in size, as shown in Fig. 2. Convolutional Neural Networks (CNN) have been widely applied to computer vision [22] and shows great effect on image detection and recognition. With CNN, effective features can be extracted automatically rather than being collected manually. Besides, MatConvNet [23], a library for CNN, provides great convenience for us.

[2] Normally, an image in TSGF contains mutiple red-like regions. Not all these regions are bloodstain and some are red car, red clothes, red light or human face etc. Thus all extracted red-like regions are mixed together at first and need manual sortation.

Similar to classification with global features, we randomly select 60 % of the images in TSLF-1X, TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X as training set while the rest as validation set and TID is used as test set. CNN requires all input images to be resized to the same size and for our network, it's 100×180[3], which is based on the mean value of height and width in TSLF-1X. Also, all images are normalized in RGB color space through the following formula:

$$p_{ij} = \frac{p_{ij} - E_i}{\sigma_i} \times 128.0 \tag{7}$$

where p_{ij} is the value of the j-th pixel of the image at the i-th color channel. E_i and σ_i represent the mean and standard deviation value for the i-th color channel respectively.

Figure 6 shows our convolutional neural network structure. At the beginning of the network, all input images should be resized to 100 × 180 in each color channel of RGB. In the middle part of it, the entire network includes five convolution layers and three pooling (subsampling) layers. Finally the input image would be classified into two types: bloody or non-bloody.

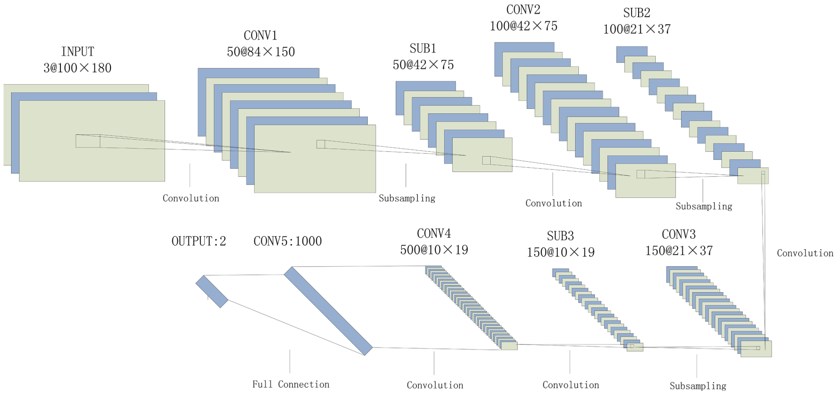

Fig. 6. Convolutional neural network structure (Color figure online)

4 Experiments

So far, we have established datasets for classification with global and local features. For classification with global features, three visual features as well as their fusion are feed into SVM to classify. For classification with local features, firstly, we train the classification model for bloodstain; secondly, all suspected bloodstain regions in an image are segmented out using transformed $YC_{g'}C_{r'}$ color space and all real bloodstain regions are filtered out using this model; finally,

[3] Height × Width.

all blood pixels are added up to determine if the image being bloody or not. As mentioned before, we define that if the proportion of blood pixels in a color image is greater than 0.5 %, it's bloody, otherwise, it is not.

Classification with Global Features. For the TSGF, we randomly select 60 % images in it as training set while the rest as validation set and TID is used as test set. During the classification stage, SVM with RBF kernel is chosen as the classifier. All visual features are normalized to [−1, 1]. Finally, different combinations of feature are separately fed into SVM to classify. The classification results with global visual features on TID are presented in Table 1.

Table 1. Classification results with global visual features on TID

Adopted features	Precision	Recall	F_1
CM	78.46 %	83.65 %	80.97 %
HOG	69.56 %	93.57 %	79.80 %
LBP	69.15 %	96.68 %	80.63 %
HOG+LBP	73.80 %	96.10 %	83.45 %
CM+LBP	75.69 %	94.37 %	84.00 %
CM+HOG+LBP	78.62 %	93.00 %	85.21 %

As we can see, CM obtains higher precision rate and lower recall rate while HOG and LBP all have higher recall rate and lower precision rate. When combined together, it shows a relatively better performance than each feature alone based on F_1, indicating that CM with HOG and LBP could provide complementary information.

Classification with Local Features. For the TSLF-1X, TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X, we also randomly select 60 % images in them as training set while the rest as validation set and TID is used as test set. We choose the network of 100th epoch[4] for five training sets respectively and training errors are all close to 0, but validation errors differ a bit, as shown in Table 2.

From the results in Table 2, we can conclude that surrounding information would enhance the bloodstain classification effect (TSLF-2X and TSLF-3X compared with TSLF-1X). However, too much surrounding information would bring extra noise and cause performance degradation (TSLF-4X and TSLF-5X compared with TSLF-2X and TSLF-3X). To sum up, TSLF-2X should be used as the dataset for local features because of its lower validation error.

Table 3 shows the comparison results of different methods on TID and results show that bloody image classification with local features is more effective based on F_1.

[4] That means the neural network passes through training set for 100 times.

Table 2. The validation errors of TSLF-1X, TSLF-2X, TSLF-3X, TSLF-4X and TSLF-5X

Classification model	Validation error
TSLF-1X	10.40 %
TSLF-2X	8.85 %
TSLF-3X	8.87 %
TSLF-4X	9.37 %
TSLF-5X	9.45 %

Table 3. The comparison results of different methods on TID

Method+(Features, Dataset)	Precision	Recall	F_1
SVM+CM+HOG+LBP	78.62 %	93.00 %	85.21 %
CNN+TSLF-2X	93.17 %	82.25 %	87.37 %

5 Conclusion

In this paper, we propose a novel region-based method for bloody image classification using Convolutional Neural Networks. At the beginning, we defined the bloody image by the proportion of blood pixels. Then we established many datasets for our task. Finally, our research is divided into two parts: classification with global visual features as well as classification with local features. In the first part, three visual features as well as their fusion are fed into SVM to classify. In the second part, we first segment all suspected bloodstain regions using transformed $YC_{g'}C_{r'}$ color space, then filter out all real bloodstain regions using Deep Learning method, finally add up the area of all blood pixels to determine if an image is bloody. The local method shows a better performance than the global one.

Future work includes a constant enlargement of our dataset with images from different sources, luminances and resolutions as well as the optimization of the convolutional neural network structure. Another future direction is to combine global and local visual features to enhance the identification performance.

References

1. Ying-Wei, W., et al.: Bloodstain segmentation in color images. In: Proceedings of the 2011 First International Conference on Robot, Vision and Signal Processing (RVSP 2011), pp. 52–55 (2011)
2. de Dios, J.J., et al.: Face detection based on a new color space YCgCr. In: Proceedings of the IEEE International Conference on Image Processing, vol. 3, pp. 902–912, September 2003
3. de Dios, J.J., Garcia, N.: Fast face segmentation in component color space. In: 2004 International Conference on Image Processing (ICIP) (IEEE Cat. No. 04CH37580), vol. 191, pp. 191–194 (2004)

4. Yoo, S., Park, R.H.: Red-Eye detection and correction using inpainting in digital photographs. IEEE Trans. Consum. Electron. **55**(3), 1006–1014 (2009)
5. Yan, G., et al.: Region-based blood color detection and its application to bloody image filtering. In: Proceedings of 2015 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR), pp. 45–50 (2015)
6. Wang, D., et al.: Baseline results for violence detection in still images. In: 2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance (AVSS), pp. 54–57 (2012)
7. Li, B., Hu, W., Xiong, W., Wu, O., Li, W.: Horror image recognition based on emotional attention. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010. LNCS, vol. 6493, pp. 594–605. Springer, Heidelberg (2011). doi:10.1007/978-3-642-19309-5_46
8. Guermazi, R., et al.: Violent web images classification based on MPEG7 color descriptors. In: 2009 IEEE International Conference on Systems, Man and Cybernetics, pp. 3106–3111 (2009)
9. Lopes, A.P.B., et al.: A bag-of-features approach based on Hue-SIFT descriptor for nude detection. In: 2009 17th European Signal Processing Conference (EUSIPCO 2009), pp. 1552–1556 (2009)
10. Ulges, A., et al.: Automatic detection of child pornography using color visual words. In: 2011 IEEE International Conference on Multimedia and Expo (2011)
11. Violent Scenes Dataset. http://www.technicolor.com/en/innovation/research-innovation/scientific-data-sharing/violent-scenes-dataset. Accessed 25 May 2016
12. Demarty, C.H., et al.: VSD, a public dataset for the detection of violent scenes in movies: design, annotation, analysis and evaluation. Multimedia Tools Appl. **74**(17), 7379–7404 (2015)
13. Schedi, M., et al.: VSD2014: a dataset for violent scenes detection in Hollywood movies and web videos. In: 2015 Proceedings of 13th International Workshop on Content-Based Multimedia Indexing (CBMI), p. 6 (2015)
14. Acar, E., et al.: Detecting violent content in Hollywood movies by mid-level audio representations. In: Czuni, L., Schoffmann, K., Sziranyi, T. (eds.) 2013 11th International Workshop on Content-Based Multimedia Indexing, pp. 73–78 (2013)
15. Gong, Y., Wang, W., Jiang, S., Huang, Q., Gao, W.: Detecting violent scenes in movies by auditory and visual cues. In: Huang, Y.-M.R., Xu, C., Cheng, K.-S., Yang, J.-F.K., Swamy, M.N.S., Li, S., Ding, J.-W. (eds.) PCM 2008. LNCS, vol. 5353, pp. 317–326. Springer, Heidelberg (2008). doi:10.1007/978-3-540-89796-5_33
16. Penet, C., et al.: Multimodal information fusion and temporal integration for violence detection in movies. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2393–2396 (2012)
17. Bermejo Nievas, E., Deniz Suarez, O., Bueno García, G., Sukthankar, R.: Violence detection in video using computer vision techniques. In: Real, P., Diaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W. (eds.) CAIP 2011. LNCS, vol. 6855, pp. 332–339. Springer, Heidelberg (2011). doi:10.1007/978-3-642-23678-5_39
18. Deniz, O., et al.: Fast violence detection in video. In: Proceedings of 9th International Conference on Computer Vision Theory and Applications (VISAP 2014), pp. 478–485 (2014)
19. Zhijie, Y., et al.: Violence detection based on histogram of optical flow orientation. In: Proceedings of the SPIE - The International Society for Optical Engineering (2013)
20. Wang, X., Han, T.X., Yan, S.: An HOG-LBP human detector with partial occlusion handling. In: Proceedings of IEEE ICCV, September 2009

21. Chang, C.C., Lin, C.J.: LIBSVM: A Library for Support Vector Machines. In: ACM Trans. Intell. Syst. Technol. **2**(3) (2011)
22. Xu-Cheng, Y., et al.: Shallow classification or deep learning: an experimental study. In: 2014 22nd International Conference on Pattern Recognition (ICPR), pp. 1904–1909 (2014)
23. MatConvNet: CNNs for MATLAB. http://www.vlfeat.org/matconvnet/. Accessed 25 May 2016