SHOT BOUNDARY DETECTION WITH SIMPLE BOOSTED CLASSIFIERS

Michal Hradiš
Doctoral Degree Programme (1), FIT BUT
E-mail: ihradis@fit.vutbr.cz

Supervised by: Pavel Zemčík
E-mail: zemcik@fit.vutbr.cz

ABSTRACT

This paper presents an approach to detection of hard cuts in video based on simple boosted classifiers and a large set of intra-frame distance measures, which are extracted from pairs of images in small local neighborhood. The performance of the proposed approach is evaluated on the test data from TRECVID 2007 shot boundary detection task.

1. INTRODUCTION

Segmentation of video sequences into separate shots is needed in many practical applications such as content-based video retrieval and video editing. A single shot is usually defined as uninterrupted video sequence taken from single camera. Segmentation of video sequences into shots is mostly done by detecting shot boundaries [13], which are characteristic by abrupt changes in the case of hard cuts or by certain pattern of changes in the case of gradual transitions. This paper focuses solely on detection of hard cuts, which are almost the only kind of shot boundaries present in unedited video and which are also the most frequent shot boundaries in edited videos.

Many approaches to detection of hard cuts have already been proposed. The simplest approaches use some kind of difference metric derived from adjacent frames or from multiple frames in local neighborhood[4]. Value of such difference metric can be thresholded by fixed threshold or the threshold can be adapted to the content of the video to optimize tradeoff between detection rate and false positive rate. Also more advanced approaches exist which aim to for more robust or faster detection. For example in [5, 6, 8], SVM classifiers are used to verify detections. In [7], the authors use k-Nearest-Neighbor classifier with a set of inter-frame dissimilarity features. The highest possible speeds of detection are achieved by systems working with video in compressed domain or with only partially decompressed video[8, 9]; however such approaches require certain type of compression and their use is thus restricted.

We approached the hard cut detection problem purely as a pattern detection task using AdaBoost [1, 2] algorithm to create the detection classifiers and a large set features which are based on intra-frame distance measures extracted from the video in the uncompressed domain.

The set of distance measures composes of per pixel distance measures (difference of pixel intensities, squared difference and correlation) and differences of RGB histograms. All distance measures are computed on a regular grid with 4 lines and 4 columns which gives 16 values for each distance measure and frame pair. Additionally, the set of features is supplemented by mean, median, standard deviation and other statistics computed from the original 16 values. The features are extracted from a set of frame pairs in a local neighborhood of classified position, giving total 2100 features respective 4200 with temporal ranks.

2. ADABOOST CLASSIFIER

AdaBoost and derived algorithms are often used for object detection [3] in computer vision where they achieve the state-of-the-art results in both classification accuracy and classification speed. When detecting objects in images with AdaBoost, a large and an overcomplete set of features (e.g. 180 000 features in [3]) is extracted from the original data using simple filters (Haar-like features) which can be individually computed very rapidly. AdaBoost then creates a strong classifier, which is a linear combination of relatively low number of simple weak classifiers (decision stumps, decision trees). As each of the weak classifiers produces decision based only on a single feature and only the features, which are needed for classification, are computed, the resulting strong classifier can be very fast. When used in this way, AdaBoost in fact performs feature selection.

Another pleasant property of the AdaBoost algorithm is that it can cope with relatively large number of samples. The number of training samples can be further significantly increased by using importance sampling during learning. Using large number of straining samples supports generalization properties of the final classifiers.

False positive rate, which has to be very low for detection classifiers, and classification time can be significantly reduced by using some kind of attentional cascade of classifiers [3]. In such cascade, background (non-face) samples that are already classified with enough confidence are rejected after each stage. The result of this is that only the most difficult samples reach the later stages of the cascade. If there are enough samples to bootstrap, the classification function can be reliably estimated even for the very rare and difficult background samples. This way, false alarm rate can be reduced below 1e-5 in face detection task and average number of evaluated weak classifiers can be reduced to three or even less [12].

3. CLASSIFICATION AND FEATURE EXTRACTION

In the proposed approach, real AdaBoost [2] learning algorithm is used to create the detection classifiers and decision trees with four leaf nodes, which are each based on single feature, are used as the weak classifiers. A major advantage of the AdaBoost algorithm in the form we use it is that it is able to cope with very large number of features from which it selects only few features for the final classifier. This gives us an opportunity to supply the learning algorithm with any feature we can think of and leave the selection of features for the learning algorithm.

All features, which are used by the weak classifiers, are based on one of inter-frame distance measures. The measures are sum of absolute differences (1), sum of squared differences (2), correlation (3) and sum of absolute difference of RGB histograms.

$$D = \frac{1}{MN} \sum_{i=0}^{M} \sum_{j=0}^{N} |I_1(i,j) - I_2(i,j)| \tag{1}$$

$$D = \frac{\sum_{i=0}^{M} \sum_{j=0}^{N} (I_1(i,j) - I_2(i,j))^2}{\sum_{i=0}^{M} \sum_{j=0}^{N} I_1(i,j)^2 \cdot \sum_{i=0}^{M} \sum_{j=0}^{N} I_2(i,j)^2} \tag{2}$$

$$D = \frac{\sum_{i=0}^{M} \sum_{j=0}^{N} (I_1(i,j) \cdot I_2(i,j))}{\sum_{i=0}^{M} \sum_{j=0}^{N} I_1(i,j)^2 \cdot \sum_{i=0}^{M} \sum_{j=0}^{N} I_2(i,j)^2} \tag{3}$$

To extract the features, image is first divided by a regular grid with four lines and four columns into sixteen disjoint patches and the distance measures are computed for each of the patch. From these sixteen values, mean, median and standard deviation is computed. After this, the original sixteen values are sorted and divided into two halves for which mean, median and standard deviation is computed too. Finally, first and second derivation is computed for each of the 100 extracted values. This gives total 300 features extracted form a single pair of video frames.

To provide the classifier with means to distinguish abrupt non-cut events in the video like flashes, etc. from the actual hard cuts, we extract the features from multiple frame pair in local neighborhood (see Figure 1). These additional features should also provide the classifier with enough information to localize the cuts precisely. The features are extracted from three frame pairs with equal distance from classified position and from four frame pairs before and after the actual position. As the number of extracted features from a single pair of frames is 300, the total number of extracted features is 2100.

For some experiments, the set of features was additionally supplemented by ranks of the original features in small temporary window. The size of the window was set to 24 frames. The feature set with the additional temporal ranks consisted of total 4200 features.

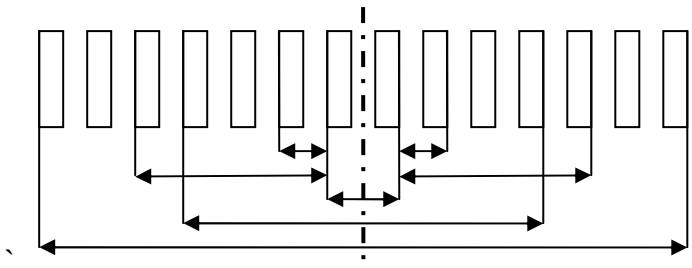

Figure 1. Frame pairs from which features are extracted.

4. TRAINING AND TESTING DATA

Six hours of hand-annotated video sequences were used for training. Five hours of the training sequences were randomly selected from the TRECVID 2007 [10] Sound and Vision data and one hour of the training data contained various material from Czech Television broadcast. The selection of video sequences for training set was not completely random, but some video sequences containing ambiguous situations (e.g. video in video) were discarded. The one hour of Czech television broadcast was added to improve generalization properties of the created classifiers. The training video sequences were hand-annotated using a video annotation tool which allowed faster then real-time annotation of shot boundaries. There are approximately 5000 cuts and 540,000 non-cuts in the training data.

The performance of created classifiers was evaluated on test data from TRECVID 2007 [10] shot boundary detection task. This data contains a total of 637,805 frames and 2,236 hard cuts and composes of news magazines, science news, news reports, documentaries, educational programming and archival video. Some of the test video sequences are poor quality and/or gray-scale. Testing was performed with the same methodology as in the TRECVID evaluation. Namely, no information about the testing dataset was used when creating the classifiers.

5. RESULTS

Four individual classifiers were created which differ in the number of selected weak classifiers, amount of non-cut samples used for training and in the feature set. The classifiers were tested with up to four different threshold settings to explore the trade-off between precision and recall. Description of the individual runs and achieved results can be seen in Table 1. This table also presents the best result (according to F-measure) for cut detection which was achieved in TRECVID 2007 evaluations and which can be used as reference. When comparing HMNR and HMWR classifiers, the temporal ranks do not seem to improve the results. On the other hand, results of the SMWR classifier and the results of the AFTER classifier suggest that longer classifiers provide better results. The AFTER classifier used importance sampling to reduce time of training. All of the classifiers were able to work in real-time even thou the algorithms were not optimized in any way.

Classifier ID	Precision	Recall	F-meaure	Negative training set size	Temporal rank features	Classifier length
HMNR_0	0,984	0,866	0,921	180000	NO	15
HMNR_1	0,967	0,955	0,961	180000	NO	15
HMNR_2	0,922	0,981	0,951	180000	NO	15
HMWR_0	0,985	0,847	0,911	170000	YES	15
HMWR_1	0,975	0,942	0,958	170000	YES	15
HMWR_2	0,944	0,972	0,958	170000	YES	15
SMWR_0	0,987	0,723	0,835	120000	YES	30
SMWR_1	0,987	0,901	0,942	120000	YES	30
SMWR_2	0,978	0,957	0,967	120000	YES	30
SMWR_3	0,96	0,976	0,968	120000	YES	30
AFTER	**0,982**	**0,973**	**0,977**	**250000**	**NO**	**60**
BRAD	0,982	0,973	0,977			

Table 1. Cut detection results of the created classifiers on the TRECVID 2007 shot boundary detection test set. The BRAD classifier is the best result (according to F-measure) for cut detection which was achieved in TRECVID 2007 evaluations (by the team from University of Bradford [11]).

6. CONCLUSION AND FUTURE WORK

Classifiers created by the AdaBoost algorithm using simple weak learners and a large set of features proved to be very suitable for detection of hard cuts in video. The best of the tested classifiers achieved precision 0,982 and recall 0,973 on the TRECVID 2007 shot boundary detection test set, which is exactly the same as the best result which has been, up to our knowledge, achieved on this dataset. Considering that these are only early experiments without any previous experience in this field, this approach to cut detection seems to be very promising.

There are many possible ways to improve the results. In the future, we plan to add more complex frame distance measures (e.g. based on motion estimation, KLT-tracking, …), we plan to use classifiers with early termination of evaluation, such as the cascade of boosted classifiers[3] or WaldBoost [12]. We also want to explore possibilities of semi-automatic annotation, which will be necessary to annotate large amount of video data, which is needed for bootstrapping.

REFERENCES

[1] Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1):119--139, August 1997

[2] Freund, Y., Schapire, R..: A short introduction to boosting. J. Japan. Soc. for Artif. Intel. 14(5), 1999, s. 771-780.

[3] Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In CVPR, 2001.

[4] Cooper, M. Adcock, J., Chen, R., Zhou, H.: FXPAL at TRECVID 2005. Proceedings of TRECVID 2005, 2005

[5] Liu, Z. et al.: AT&T research at TRECVID 2007. Proceedings of TRECVID 2007

[6] Zhao, Z. et al.: BUPT at TRECVID 2007: Shot Boundary Detection.TRECVID 2007

[7] Cooper, M. et al.: FXPAL at TRECVID 2005. Proceedings of TRECVID 2005

[8] Matsumoto, K.: Shot Boundary Detection and Low-Level Feature Extraction Experiments for TRECVID 2005. Proceedings of TRECVID 2005

[9] Zhao, Zhi-Cheng, Cai, An-Ni. Shot Boundary Detection Algorithm in Compressed Domain Based on AdaBoost and Fuzzy Theory

[10] Smeaton, A. F., Over, P., and Kraaij, W.: Evaluation campaigns and TRECVID. In Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval. MIR '06. ACM Press, New York, NY, 2006, 321-330.

[11] Ren, J. J., Chen, J.: Determination of Shot Boundary in MPEG Videos for TRECVID 2007. Proceedings of TRECVID 2007

[12] Šochman, J., Matas. J.: WaldBoost -- Learning for Time Constrained Sequential Detection. In CVPR 2005

[13] Yuan, J. et al.: A Formal Study of Shot Boundary Detection. In CirSysVideo, No. 2, February 2007, 168-186