2015

# Shot Boundary Detection with Graph Theory using Keypoint Features and Color Histograms

Lee, Kyoungmin; Kölsch, Mathias

IEEE

Lee, Kyoungmin, and Mathias Kolsch. "Shot boundary detection with graph theory using keypoint features and color histograms." 2015 IEEE Winter Conference on Applications of Computer Vision. IEEE, 2015.
http://hdl.handle.net/10945/63556

# Shot Boundary Detection with Graph Theory using Keypoint Features and Color Histograms

Kyoungmin Lee, and Mathias Kölsch
Naval Postgraduate School, Monterey, 93940, USA
lekomin@gmail.com; kolsch@nps.edu

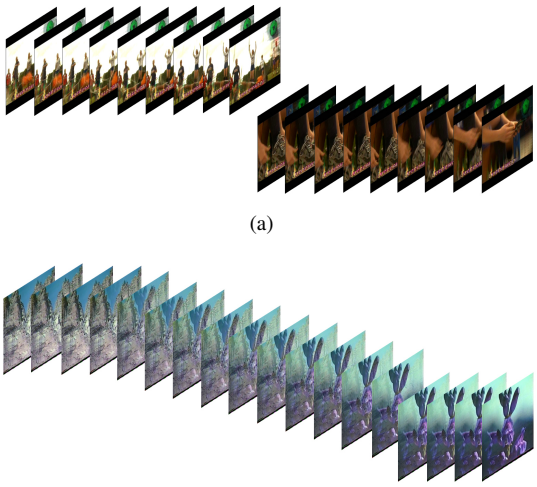
## Abstract

*The TRECVID report of 2010 [14] evaluated video shot boundary detectors as achieving "excellent performance on [hard] cuts and gradual transitions." Unfortunately, while re-evaluating the state of the art of the shot boundary detection, we found that they need to be improved because the characteristics of consumer-produced videos have changed significantly since the introduction of mobile gadgets, such as smartphones, tablets and outdoor activity-purposed cameras, and video editing software has been evolving rapidly. In this paper, we evaluate the best-known approach on a contemporary, publicly accessible corpus, and present a method that achieves better performance, particularly on soft transitions. Our method combines color histograms with keypoint feature matching to extract comprehensive frame information. Two similarity metrics, one for individual frames and one for sets of frames, are defined based on graph cuts. These metrics are formed into temporal feature vectors on which a SVM is trained to perform the final segmentation. The evaluation on said "modern" corpus of relatively short videos yields a performance of 92% recall (at 89% precision) overall, compared to 69% (91%) of the best-known method.*

## 1. Introduction

Movies and edited videos consist of scenes, such as a dialog between two people. Scenes consist of one or more shots, or consecutive frames as captured with a single camera. Locating transitions between shots, also called cuts or *shot boundaries*, is fundamental procedure for analyzing videos such as indexing videos, querying scenes, searching objects, or summarizing video contents. All shot boundaries can be classified into the following two categories:

- A **hard cut**, as shown in Fig. 1(a), is an instant transition such that frame $n$ is from shot $k$ and the very next frame $n+1$ is from the following shot $k+1$. Research for detecting hard cuts has matured as reported in [14].



(a)

(b)

Figure 1. Examples of transitions: (a) a hard cut is an instant transition from one shot to the next, and (b) a soft cut transitions gradually. Frames are arranged sequentially from left to right.

- A **soft cut** (or soft transition), as shown in Fig. 1(b), is a gradual transition over the course of multiple frames. The performance for detecting soft cuts is much worse because there is no abrupt change in the visual frame content. In particular, there are:
  - **Fade-out, fade-in** dissolves where the two shots get blended on top of each other. The details vary on aspects including transition duration, amount of blending, and possible dimming.
  - **Geometric transitions** involve wiping out or in (sliding the previous or next shot over the other), zooming in the next shot, block puzzles etc. Without an overarching description of such transformations, it will remain incredibly challenging to automatically detect them.
  - **Artistic transitions** will be nearly impossible to automatically detect, such as many transitions in the 1986 "Highlander" movie, including fading

978-1-4799-6683-7/15 $31.00 © 2015 IEEE
DOI 10.1109/WACV.2015.161

from an actor to a Mona Lisa painting or panning from a fish tank to surfacing from below a lake.

Shot boundary detection (SBD) is difficult because of the great variety of transition types and the possible similarity of the shot before and after the boundary. The TRECVID challenges from 2001 and 2007 included shot boundary detection tasks, yet Smeaton et al. [14] discontinued them due to the methods' "excellent performance."

Unfortunately, for a few years after the triumph, the characteristics of consumer-produced videos have changed significantly due to the proliferation of smartphones and video editing software. While re-evaluating the state of the art of shot boundary detection, our implementation of TRECVID's best-performing algorithm (Yuan et al. [17]) performed well on hard cuts but recall was below 50% for other boundaries: especially, when detecting artistic transitions, recall was only 3.5% (details will be shown in Section 5). From these experiences, we concluded that 1) a video corpus needs to be updated for reflecting recent change of characteristics of videos and 2) Yuan et al's approach [17] needs to be improved in terms of various similarity metrics between frames and integration of those metrics.

Based on the empirical conclusion, in this paper, we proposed a new method for shot boundary detection. Our method combines color histograms with keypoint feature matching to extract comprehensive frame information. Then, based on spectral graph theory [13], the information is used for defining two similarity metrics, one for individual frames and one for sets of frames. Finally, these metrics are formed into temporal feature vectors on which a SVM is trained to perform the final segmentation. For measuring similarity, we used spectral graph theory [13] like Yuan et al [17] that has been known as the best-performing method. Our main contributions are as follows:

1. We first collected a contemporary video corpus that reflects current consumer-grade video characteristics with respect to content, camera type, length, and editing. Care was taken to avoid copyrighted material to limit the effect of use restrictions. Table 2 presents details of this public corpus, including hyperlinks to the actual videos [1].

2. We used an efficient strategy for selecting frames with the Fibonacci sequence because using all member frames increases computation time and reduces detection performance.

3. As a new similarity measurement between frames, we proposed a new metric using the number of matching keypoints with the $exp()$ function. Applying the $exp()$ function contributes to increase robustness because it can reduce the undesirable effect of mis-matching keypoints [15].

4. For integrating two features (color block histogram and keypoint) effectively, we proposed a logically combined classifier that we call $\text{SVM}_{\text{OR}}$. In our experiment, while generating a feature vector for training a Support Vector Machine (SVM [12]), the direct combination of two features achieved inferior performance to the proposed classifier.

The next section discusses related work, followed by details of the proposed method. Section 4 presents the data set and the experiments we conducted, followed by the evaluation results and conclusions.

## 2. Related Work

According to Smeaton et al. [14], Tsinghua University's approach [17] achieved the best SBD performance in terms of speed, recall, and precision. It extracts a color block histogram for each frame and computes inter-frame similarity with correlation. Similarity between groups of frames is measured with spectral graph theory [13] and fed to a SVM classifier for SBD. The work of W. Hu et al. [6] re-emphasizes that correlation between color block histograms outperforms edge and motion features for segmenting shots. Neither of these methods considered keypoint feature matching [9] as a similarity metric.

Keypoint feature matching has been employed for SBD [10, 7, 8]. These methods classified with simple thresholds, however, and were not able to match the performance of statistics or learning-based methods [14, 6, 3]. In [2], concept of keypoint feature extraction was used for summarizing a frame in a single vector with quantizing color information. After clustering them into multiple scenes, shot boundaries were detected. Smeaton et al. [14] and Hu et al. [6] provide good summaries of the state of the art of SBD.

## 3. Proposed Approach

This section describes the proposed approach: two transition metrics that can be calculated at every frame, and how these metrics form feature vectors on which a SVM classifier is trained. The transition metrics are calculated by first extracting descriptive features (color and appearance) for each frame, then measuring the similarity between two frames, and, finally, calculating a similarity between groups of frames.

[1] In our evaluation, in addition to newly collected videos, we used old TRECVID videos (2001 and 2002) among all TRECVID videos between 2001 and 2007. Though we are still looking for TRECVID videos between 2003 and 2007, NIST (TRECVID organizing institution) and Linguistic Data Consortium (LDC) do not provide those data any more and videos only between 2001 and 2002 are still available online.

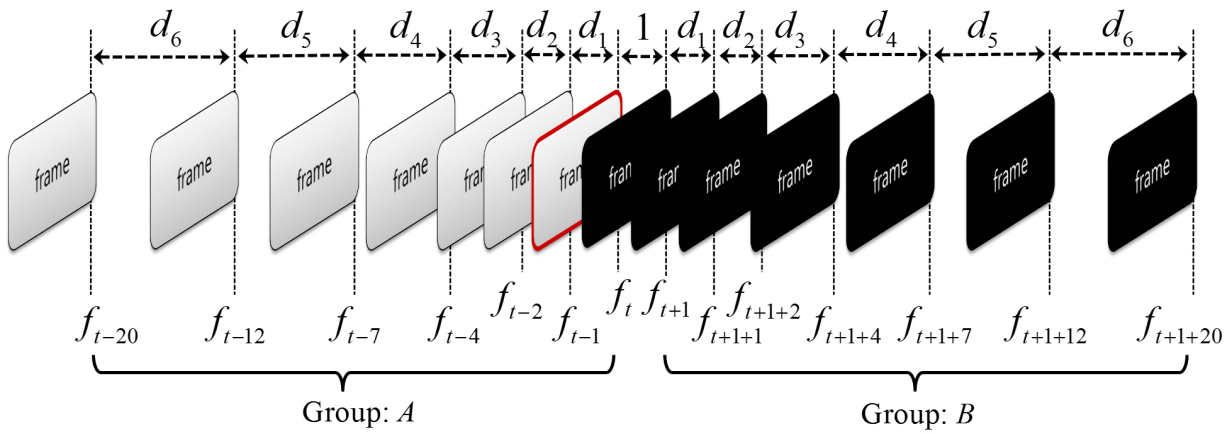



Figure 2. Two groups of frames for computing two transition metrics at frame $t$: $m^{\mathrm{S}}(t)$ in Eq. (3) and $m^{\mathrm{C}}(t)$ in Eq. (6).

### 3.1. Frame representation in feature space

We extract two types of descriptive features from each frame:

**Keypoint features** are appearance-based descriptors calculated at image interest points and designed to be robust to brightness, scale, rotation, and other image transformations. For the proposed method, the SIFT (scale-invariant feature transform) algorithm [9] selects the location of feature points and represents each with descriptors. Each video frame is represented with a set of 128-dimensional descriptor vectors.

**Color Block Histograms** (CBH) are computed in RGB space. To consider spatial information, the frame is partitioned into several blocks in which separate color histograms are computed. Each video frame is represented with a set of 2000-dimensional vectors (using five bins for each color space with 4×4 blocks). Sec. 3.4, describes how the similarity measure is defined for CBHs.

### 3.2. Frame similarity

The similarity of two frames is based on the number of matching and non-matching descriptors. Descriptor match is determined with the ratio test as proposed by Lowe [9]: two features are considered a match if the ratio of their distance to the second-closest distance is bigger than a threshold. This attempts to avoid ambiguous matches. (Our approach does not take frame size into account as it might complicate parameter handling, despite potential precision improvements [7].)

The similarity between two frames is defined as follows:

$$\mathrm{sim}_f^{\mathrm{SIFT}}(f_i, f_j) = \exp\left\{\frac{-1}{\sigma^2}\left(\frac{n(f_i)+n(f_j)-2n(f_i,f_j)}{n(f_i,f_j)+1}\right)^2\right\} \quad (1)$$

where $f_i$ is $i$-th frame, $n(f_i)$ is the total number of keypoint features in frame $i$, and $n(f_i, f_j)$ is the number of matching features between frames $i$ and $j$. For $i = j$, $sim_f(f_i, f_j) := 1$.

The numerator in Eq. (1) indicates the number of unmatched features, the denominator indicates the number of matched features. Thereby, similarity is proportional to the number of matched features and as well as to the ratio of matched-to-unmatched features. The last term $(+1)$ avoids a division by zero when there is no matching feature. Empirically, we set $\sigma = 10$.

### 3.3. Similarity of groups of frames:

Comparing a frame with its immediate neighbor frame is insufficient for detecting soft transitions because they occur over multiple frames as discussed in the Introduction and as can be seen in Fig. 3. Therefore, soft boundaries are much easier to spot when considering a sequence of frames rather than individual frames. Soft transitions can be considered a transition from one group of frames to another group of frames, and defining a *group similarity* will help identify cases where the two groups stem from different shots. Yuan et al. [17] measure group similarity by adopting spectral graph theory [13] to their purposes. Next, the similarity measure is described first, then the selection of member frames for each group.

#### 3.3.1 Measuring similarity between two groups of frames:

As in the work of Yuan et al. [17], we will adopt the min-max cut algorithm [5] for measuring similarity between two groups because it captures intra-group connectivity and inter-group disconnectivity simultaneously. Similarity between two groups of frames is defined as follows:

$$\begin{aligned}\text{sim}_g^{\text{SIFT}}(A,B) &= \frac{\text{cut}(A,B)}{\text{assoc}(A)} + \frac{\text{cut}(A,B)}{\text{assoc}(B)} \\ &= \frac{\sum_{i\in A, j\in B} \text{sim}_f^{\text{SIFT}}(f_i,f_j)}{\sum_{i\in A, j\in A} \text{sim}_f^{\text{SIFT}}(f_i,f_j)} \\ &\quad + \frac{\sum_{i\in A, j\in B} \text{sim}_f^{\text{SIFT}}(f_i,f_j)}{\sum_{i\in B, j\in B} \text{sim}_f^{\text{SIFT}}(f_i,f_j)} \quad (2)\end{aligned}$$

where $A, B$ are sets of frames. The following section describes how to select member frames for each group.

#### 3.3.2 Selecting member frames for groups:

More frame members in each group does not equate to better performance for SBD as Yuan et al. [17] showed. Instead, proper selection of member frames is directly related to detection performance. In this paper, the Fibonacci sequence ($F_n = F_{n-1} + F_{n-2}$ with seed values $F_1 = 1$ and $F_2 = 1$) dictated the distance between frames. Hence, in Fig. 2, $D = \{d_1, d_2, \cdots, d_6\}$ is set to $\{1, 1, 2, 3, 5, 8\}$. Frame selection with the Fibonacci sequence reduced computation time as well as increased performance compared to using all successive forty frames. Hence, as shown in Fig. 2, groups $A_t$ and $B_t$ are chosen as:

$$\begin{aligned}A_t &= \{f_{t-20}, f_{t-12}, f_{t-7}, f_{t-4}, f_{t-2}, f_{t-1}, f_t\} \\ B_t &= \{f_{t+1}, f_{t+1+1}, f_{t+1+2}, f_{t+1+4}, f_{t+1+7}, \\ &\quad f_{t+1+12}, f_{t+1+20}\}\end{aligned}$$

### 3.4. Calculating transition metrics at each frame

Two transition metrics are defined, one based on keypoint features, and one based on color histograms. The metric based on SIFT keypoint features is the group similarity between frame groups $A_t$ and $B_t$ (where the last frame of $A_t$ is $f_t$ and the first frame of $B_t$ is $t+1$), hence:

$$m^{\text{S}}(t) = \text{sim}_g^{\text{SIFT}}(A_t, B_t). \quad (3)$$

The transition metric for **color block histogram** (CBH) is again based on an individual frame similarity, particularly, the correlation between CBHs. A new group similarity is then defined as follows, leading to the transition metric $m^{\text{C}}(t)$. The group subscripts $t$ are dropped for legibility.

$$\text{sim}_f^{\text{CBH}}(f_i, f_j) = \text{correlation of color block histograms of two frames}(f_i, f_j) \quad (4)$$

$$\begin{aligned}\text{sim}_g^{\text{CBH}}(A,B) &= \frac{\sum_{i\in A, j\in B} \text{sim}_f^{\text{CBH}}(f_i,f_j)}{\sum_{i\in A, j\in A} \text{sim}_f^{\text{CBH}}(f_i,f_j)} \\ &\quad + \frac{\sum_{i\in A, j\in B} \text{sim}_f^{\text{CBH}}(f_i,f_j)}{\sum_{i\in B, j\in B} \text{sim}_f^{\text{CBH}}(f_i,f_j)} \quad (5)\end{aligned}$$

$$m^{\text{C}}(t) = \text{sim}_g^{\text{CBH}}(A_t, B_t) \quad (6)$$

### 3.5. Detecting shot boundaries with a SVM

Through above procedure, two metric values can be obtained for each frame that has sufficient neighbor frames. In many SBD approaches, other similarity measures have been merely thresholded to directly determine the location of SBDs [1, 16]. Here, we combine the metric values from multiple frames into a feature vector and train a Support Vector Machine (SVM [12]) on annotated videos to learn a suitable decision boundary. Two separate SVMs are trained, $\text{SVM}_{\text{SIFT}}$ and $\text{SVM}_{\text{CBH}}$, one each for feature vectors from the two transition metrics $m^{\text{S}}(t)$ and $m^{\text{C}}(t)$:

$$\begin{aligned}x_t^S &= \big[m^{\text{S}}(t-t_6), \cdots, m^{\text{S}}(t-t_1), m^{\text{S}}(t), \\ &\quad m^{\text{S}}(t+1+t_1), \cdots, m^{\text{S}}(t+1+t_6)\big]^T \quad (7) \\ x_t^C &= \big[m^{\text{C}}(t-t_6), \cdots, m^{\text{C}}(t-t_1), m^{\text{C}}(t), \\ &\quad m^{\text{C}}(t+1+t_1), \cdots, m^{\text{C}}(t+1+t_6)\big]^T. \quad (8)\end{aligned}$$

The two classifiers were also combined into one classifier ($\text{SVM}_{\text{OR}}$) with a logical OR operation (often called "late fusion") as shown in Table 1.

Table 1. Logical OR operation of two classifiers

| $\text{SVM}_{\text{OR}}$ | Positive by $\text{SVM}_{\text{SIFT}}$ | Negative by $\text{SVM}_{\text{SIFT}}$ |
|---|---|---|
| Positive by $\text{SVM}_{\text{CBH}}$ | Positive | Positive |
| Negative by $\text{SVM}_{\text{CBH}}$ | Positive | Negative |

As shown in the next section, the cooperative classifier $\text{SVM}_{\text{OR}}$ achieves superior performance over the separate classifiers $\text{SVM}_{\text{SIFT}}$ and $\text{SVM}_{\text{CBH}}$. Moreover, even a single SVM classifier ($\text{SVM}_{\text{MERGE}}$) that was trained on the *combined* feature vectors $x_t = \left[x_t^S,\ x_t^C\right]^T$ (often called "early fusion") achieved inferior performance compared to the cooperative classifier. In all four SVM classifier models a radial basis function was used as kernel.

## 4. Data and Experiments

Performance was tested on a new corpus of three types of videos (see Table 2):

Videos in the **professionally-edited** set are obtained from the TRECVID (Text REtrieval Conference Video Retrieval Evaluation[2]) challenge. This set includes broadcast news videos, NIST videos, BBC stock shots, etc. as described by Smeaton et al. [14]. Only some videos from the 2001 and 2002 TRECVID challenge are still publicly available, and of those we selected more recent videos and those free of technical issues such as de-interlacing.

[2] http://trecvid.nist.gov

Table 2. The composition of our new video corpus. Note that the titles are hyperlinks to the actual videos.

| Type | Name | # of Hard cuts | # of Soft cuts | Length (frames) |
|---|---|---|---|---|
| Pro1 | The Rio Grande | 0 | 137 | 24,550 |
| Pro2 | Desert Venture | 104 | 25 | 24,983 |
| Pro3 | NASA 25th Anniv.(seg. 9) | 39 | 63 | 12,307 |
| Pro4 | Exotic Terrane | 87 | 89 | 40,095 |
| Pro5 | NASA 25th Anniv.(seg. 5) | 38 | 28 | 11,364 |
| Pro6 | Challenge at Glen Canyon | 233 | 11 | 48,451 |
| Pro7 | Hidden Treasures | 143 | 93 | 50,823 |
| Pro8 | Wrestling with Uncertainty | 45 | 148 | 53,014 |
| Pro9 | Senses & Sensitivity (Lec.3) | 292 | 16 | 86,789 |
| Pro10 | The Technical Knockout | 605 | 108 | 105,661 |
| Pro-Total | Profession-edited videos | 1,586 | 718 | 458,037 |
| Ama1 | Best Vines | 71 | 2 | 6,712 |
| Ama2 | Colorado Snowboarding | 48 | 3 | 5,859 |
| Ama3 | Flying to Abu Dhabi | 23 | 0 | 7,744 |
| Ama4 | Freestyle Swimming | 7 | 1 | 6,797 |
| Ama5 | GoPro Montage | 39 | 1 | 7,552 |
| Ama6 | Huge Avalanche | 1 | 13 | 11,476 |
| Ama7 | Like A Flying Boss | 72 | 6 | 9,571 |
| Ama8 | New York Bound | 99 | 3 | 10,073 |
| Ama9 | Surfing Montage | 53 | 0 | 6,102 |
| Ama-Total | Amateur-edited videos | 413 | 29 | 71,886 |
| Art1 | Final Cut Pro 7 | 124 | 59 | 9,249 |
| Art2 | Final Cut Pro X | 88 | 88 | 10,920 |
| Art3 | Nor'Easter | 0 | 26 | 8,365 |
| Art-Total | Artistically edited videos | 212 | 173 | 28,534 |
| Total | Total videos | 2,211 | 920 | 558,457 |

Videos in the **amateur-edited** set reflect the trend of capturing with smartphone cameras, outdoor activity-purposed cameras s.a. the GoPro.[3] These cameras have made it possible to capture very dynamic videos of new activities, from new points of view, and with different optics compared to common video from ten years ago. SBD on this set is particularly difficult due to fast motion and dynamic viewpoint changes.

The third set of **artistically-edited** videos contains three videos that showcase an unusual variety of shot boundary types. Recent video editing tools provide a plethora of transition effects and the third category is for evaluating the performance of detecting these artistic shot boundaries.

Recall, precision, and $F_1$ score were used as performance metrics, which are defined as follows:

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{10}$$

$$F_1 \text{ score} = 2\frac{\text{recall} \cdot \text{precision}}{\text{recall} + \text{precision}} \tag{11}$$

[3] http://gopro.com

where TP is the number of correctly detected shot boundaries, FN is the number of missed shot boundaries, and FP is the number of falsely detected shot boundaries.

The experiment compared four new approaches to a baseline algorithm:

**Yuan et al. [17]** is our implementation of TRECVID's best-performing algorithm, which describes frames with a color block histogram (CBH) only, collects CBHs from successive frames into a feature vector, and classifies with one SVM each for hard and soft cuts.

**SVM$_{\text{CBH}}$** With a few modifcations, the above algorithm [17] achieved markedly better precision: with a $4 \times 4$ instead of a $2 \times 2$ block color histogram, a single SVM instead of one each for hard and soft cuts, and Fibonacci-based frame selection instead of successive frames.

**SVM$_{\text{SIFT}}$** This classifier was trained with keypoint similarity measures only, per $x_t^S$ of Eq. (7).

**SVM$_{\text{MERGE}}$** This SVM was trained on merged feature vectors from both color and keypoint similarity measures

Table 3. Performance Results [a]

| | Hard cuts + Soft cuts | | | | | | Hard[b] | | | Soft[b] | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Classifier | TP + FN | TP | TP + FP | Recall (%) | Precision (%) | $F_1$ score | TP + FN | TP | Recall (%) | TP + FN | TP | Recall (%) |
| **Profession-edited videos** | | | | | | | | | | | | |
| Yuan et al. [17] | 2,304 | 1,638 | 1,808 | 71.1 | 90.6 | 79.7 | 1,586 | 1,383 | 87.2 | 718 | 255 | 35.5 |
| $SVM_{CBH}$ | | 1,870 | 1,974 | 81.2 | 94.7 | 87.4 | | 1,507 | 95.0 | | 363 | 50.6 |
| $SVM_{SIFT}$ | | 2,066 | 2,205 | 89.7 | 93.7 | 91.6 | | 1,547 | 97.5 | | 519 | 72.3 |
| $SVM_{MERGE}$ | | 2,033 | 2,134 | 88.2 | 95.3 | 91.6 | | 1,531 | 96.5 | | 502 | 69.9 |
| $SVM_{OR}$ | | 2,134 | 2,363 | 92.6 | 90.3 | 91.5 | | 1,572 | 99.1 | | 562 | 78.3 |
| **Amateur-edited videos** | | | | | | | | | | | | |
| Yuan et al. [17] | 442 | 300 | 351 | 67.9 | 85.5 | 75.7 | 413 | 298 | 72.2 | 29 | 2 | 6.9 |
| $SVM_{CBH}$ | | 388 | 465 | 87.8 | 83.4 | 85.6 | | 376 | 91.0 | | 12 | 41.4 |
| $SVM_{SIFT}$ | | 320 | 383 | 72.4 | 83.6 | 77.6 | | 312 | 75.5 | | 8 | 27.6 |
| $SVM_{MERGE}$ | | 292 | 365 | 66.1 | 80.0 | 72.4 | | 285 | 69.0 | | 7 | 24.1 |
| $SVM_{OR}$ | | 408 | 541 | 92.3 | 75.4 | 83.0 | | 393 | 95.2 | | 15 | 51.7 |
| **Artistically edited videos** | | | | | | | | | | | | |
| Yuan et al. [17] | 385 | 218 | 223 | 56.6 | 97.8 | 71.7 | 212 | 212 | 100.0 | 173 | 6 | 3.5 |
| $SVM_{CBH}$ | | 294 | 295 | 76.4 | 99.7 | 86.5 | | 212 | 100.0 | | 82 | 47.4 |
| $SVM_{SIFT}$ | | 295 | 295 | 76.6 | 100.0 | 86.8 | | 212 | 100.0 | | 83 | 48.0 |
| $SVM_{MERGE}$ | | 313 | 318 | 81.3 | 98.4 | 89.0 | | 212 | 100.0 | | 101 | 58.4 |
| $SVM_{OR}$ | | 328 | 329 | 85.2 | 99.7 | 91.9 | | 212 | 100.0 | | 116 | 67.1 |
| **Total** | | | | | | | | | | | | |
| Yuan et al. [17] | 3,131 | 2,156 | 2,382 | 68.9 | 90.5 | 78.2 | 2,211 | 1,893 | 85.6 | 920 | 263 | 28.6 |
| $SVM_{CBH}$ | | 2,552 | 2734 | 81.5 | 93.3 | 87.0 | | 2,095 | 94.8 | | 457 | 49.7 |
| $SVM_{SIFT}$ | | 2,681 | 2,883 | 85.6 | 93.0 | 89.2 | | 2,071 | 93.7 | | 610 | 66.3 |
| $SVM_{MERGE}$ | | 2,638 | 2,817 | 84.3 | 93.6 | 88.7 | | 2,028 | 91.7 | | 610 | 66.3 |
| $SVM_{OR}$ | | 2,870 | 3,233 | 91.7 | 88.8 | 90.2 | | 2,177 | 98.5 | | 693 | 75.3 |

[a] Leave-one-out cross-validation (LOOCV) was used with a single video for evaluation and the remaining videos for training.

[b] $SVM_{CBH}$, $SVM_{SIFT}$, $SVM_{MERGE}$ and $SVM_{OR}$ approaches segment with a single SVM for both hard and soft cuts, hence they do not indicate the kind of detected cut. Therefore, recall per transition kind can be computed but precision cannot.

(Eqs. (7) and (8)).

**$\text{SVM}_{\text{OR}}$** This classifier constitutes a late fusion of the above two classifiers ($\text{SVM}_{\text{CBH}}$ and $\text{SVM}_{\text{SIFT}}$) with a logical OR operator.

## 5. Results

### 5.1. Quantitative Results:

For the videos in the professionally-edited set, the baseline method [17] performed well on both hard cuts and soft cuts as reported. While it also performed well on hard cuts in amateur-edited and artistically edited videos, there, it achieved less than 10% recall on soft cuts.

Of the evaluated methods, $\text{SVM}_{\text{OR}}$ achieves the best performance on all three video types in our corpus (see Table 3). Despite lower precision than the baseline approach [17], $\text{SVM}_{\text{OR}}$ is an overall better SBD method according to its higher $F_1$ score (the harmonic mean of precision and recall rates), and owing to much improved recall: It detects $13\%$ $\left(\frac{2177-1893}{2211}\right)$ more hard cuts and $47\%$ $\left(\frac{693-263}{920}\right)$ more soft cuts.

Note that $\text{SVM}_{\text{MERGE}}$ was anticipated to show the best performance because the SVM was expected to benefit from having access to the entire color and keypoint feature vectors. As Table 3 shows, however, early feature fusion ($\text{SVM}_{\text{MERGE}}$) was inferior to late fusion ($\text{SVM}_{\text{OR}}$) of independently classified features–in fact, its performance was similar or worse than that of each independent classifier. As discussed by Chen and Lin [4], an appropriate feature selection strategy is necessary for achieving better performance with a combined classifier than with the independent classifiers.

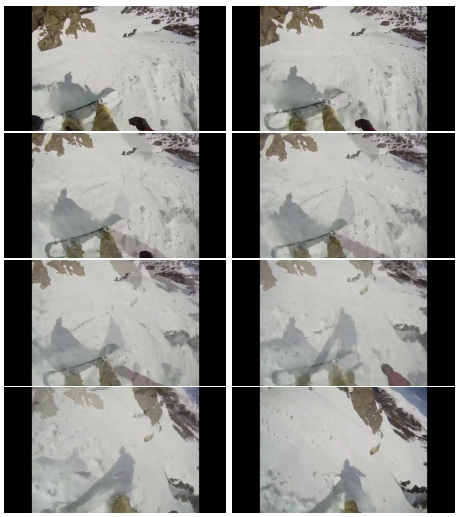


Figure 3. An examples of a missed shot boundary: there are only subtle changes in a very similar environment.

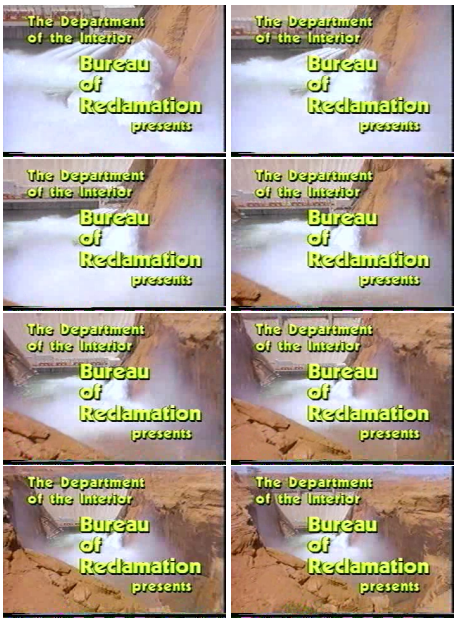
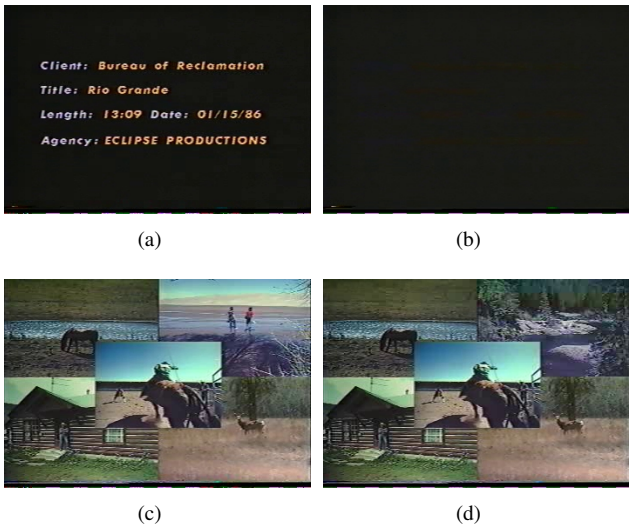



Figure 4. An example of a falsely detected shot boundary: These frames (left to right) are detected as a transition because the background is zoomed-out rapidly.




(a) (b)

(c) (d)

Figure 5. Examples of ambiguous shot boundaries: (a) - (b) are falsely detected shot boundaries because text change has not been considered a shot boundary. (c) - (d) are falsely detected ones because partial frame changes (s.a. less than one quarter) have not been considered shot boundaries.

### 5.2. Qualitative Results:

The video showing the worst performance was Ama6 (Huge Avalanche). Of 13 soft cuts, nine were missed by $\text{SVM}_{\text{OR}}$. Fig. 3 shows one missed SBD, a difficult case even for human observers because of only subtle changes of shadows and rocks in an otherwise very similar environment. $\text{SVM}_{\text{OR}}$ also failed to detect boundaries when the

camera moves very fast or the video is significantly blurred.

Fig. 4 shows falsely detected shot boundaries. Even though keypoint features (extracted predominantly from the static characters) indicate that this is not a shot boundary, the rapid zooming-out changes the background and therefore the color histograms. Fig. 5 shows two examples of ambiguous cases. They were detected as shot boundaries, but had not been annotated as such. A more precise, measurable definition would be necessary for avoiding such ambiguous cases.

## 6. Conclusions

We proposed a method for shot boundary detection (SBD) that combines two SVM classifiers; one based on color histograms, and one based on appearance features. A similarity measure between two frames was defined based on keypoint feature matching, and the similarity between two groups relied on graph theory and on selecting member frames according to the Fibonacci sequence. Finally, the two independent classifiers were combined into one classifier, $\text{SVM}_{\text{OR}}$, through a logical OR operation.

The proposed method $\text{SVM}_{\text{OR}}$ was compared with the best-known SBD (Yuan et al. [17]) on a novel video corpus. This corpus was assembled in consideration of characteristics of recent consumer-produced video and video-editing technology. Our experiments showed that $\text{SVM}_{\text{OR}}$ achieved a 12% point improvement overall and over 46% point improvement for soft cut detection.

Beyond the contribution of a contemporary corpus for evaluation and a novel method for performing SBD, we hope to revive interest in this topic as it is a core building block for video indexing, search and retrieval–increasingly important capabilities for dealing with the influx of videos from the current generation of video-capturing gadgets.

For future research, we will experiment with different keypoint features that reportedly are faster and more accurate (Rublee et al. [11]) than the SIFT algorithm.

## References


[1] A. Amiri and M. Fathy. Video shot boundary detection using generalized eigenvalue decomposition and gaussian transition detection. *Computing and Informatics*, 30(3):595–619, 2011.

[2] Y. Chang, D. J. Lee, Y. Hong, and J. Archibald. Unsupervised video shot detection using clustering ensemble with a color global scale-invariant feature transform descriptor. *J. Image Video Process.*, 2008:9:1–9:10, Jan. 2008.

[3] V. Chasanis, A. Likas, and N. Galatsanos. Simultaneous detection of abrupt cuts and dissolves in videos using support vector machines. *Pattern Recognition Letters*, 30(1):55 – 65, 2009.

[4] Y.-W. Chen and C.-J. Lin. Combining SVMs with Various Feature Selection Strategies. In I. Guyon, M. Nikravesh, S. Gunn, and L. Zadeh, editors, *Feature Extraction*, volume 207 of *Studies in Fuzziness and Soft Computing*, pages 315–324. Springer Berlin Heidelberg, 2006.

[5] C. Ding, X. He, H. Zha, M. Gu, and H. Simon. A min-max cut algorithm for graph partitioning and data clustering. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 107–114, 2001.

[6] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank. A Survey on Visual Content-Based Video Indexing and Retrieval. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 41(6):797–819, Nov 2011.

[7] C.-R. Huang, H.-P. Lee, and C.-S. Chen. Shot Change Detection via Local Keypoint Matching. *Multimedia, IEEE Transactions on*, 10(6):1097–1108, Oct 2008.

[8] G. Liu, X. Wen, W. Zheng, and P. He. Shot Boundary Detection and Keyframe Extraction Based on Scale Invariant Feature Transform. In *Computer and Information Science, 2009. ICIS 2009. Eighth IEEE/ACIS International Conference on*, pages 1126–1130, June 2009.

[9] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004.

[10] M.-H. Park, R.-H. Park, and S. W. Lee. Shot boundary detection using scale invariant feature matching. In *Proc. SPIE Visual Communications and Image Processing*, volume 6077, pages 569–577, 2006.

[11] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2564–2571, Nov 2011.

[12] B. Scholkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[13] J. Shi and J. Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, Aug 2000.

[14] A. F. Smeaton, P. Over, and A. R. Doherty. Video shot boundary detection: Seven years of TRECVid activity. *Computer Vision and Image Understanding*, 114(4):411 – 418, 2010. Special issue on Image and Video Retrieval Evaluation.

[15] Y. Tsin and T. Kanade. A correlation-based approach to robust point set registration. In T. Pajdla and J. Matas, editors, *Computer Vision - ECCV 2004*, volume 3023 of *Lecture Notes in Computer Science*, pages 558–569. Springer Berlin Heidelberg, 2004.

[16] X. Wu, P. Yuen, C. Liu, and J. Huang. Shot Boundary Detection: An Information Saliency Approach. In *Image and Signal Processing, 2008. CISP '08. Congress on*, volume 2, pages 808–812, May 2008.

[17] J. Yuan, H. Wang, L. Xiao, W. Zheng, J. Li, F. Lin, and B. Zhang. A Formal Study of Shot Boundary Detection. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(2):168–186, Feb 2007.