symmetry

MDPI

Article

The Detection of Video Shot Transitions Based on Primary Segments Using the Adaptive Threshold of Colour-Based Histogram Differences and Candidate Segments Using the SURF Feature Descriptor

Raja Suguna M. [1,*], Kalaivani A. [2] and Anusuya S. [2]

[1] Department of Computer Science and Engineering, RMK College of Engineering and Technology, Chennai 600095, India
[2] Department of Information and Technology, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai 600095, India
* Correspondence: rajasugunacse@rmkcet.ac.in or rajasugunaalagar@gmail.com

Abstract: Aim: Advancements in multimedia technology have facilitated the uploading and processing of videos with substantial content. Automated tools and techniques help to manage vast volumes of video content. Video shot segmentation is the basic symmetry step underlying video processing techniques such as video indexing, content-based video retrieval, video summarization, and intelligent surveillance. Video shot boundary detection segments a video into temporal segments called shots and identifies the video frame in which a shot change occurs. The two types of shot transitions are cut and gradual. Illumination changes, camera motion, and fast-moving objects in videos reduce the detection accuracy of cut and gradual transitions. Materials and Methods: In this paper, a novel symmetry shot boundary detection system is proposed to maximize detection accuracy by analysing the transition behaviour of a video, segmenting it initially into primary segments and candidate segments by using the colour feature and the local adaptive threshold of each segment. Thereafter, the cut and gradual transitions are fine-tuned from the candidate segment using Speeded-Up Robust Features (SURF) extracted from the boundary frames to reduce the algorithmic complexity. The proposed symmetry method is evaluated using the TRECVID 2001 video dataset, and the results show an increase in detection accuracy. Result: The F1 score obtained for the detection of cut and gradual transitions is 98.7% and 90.8%, respectively. Conclusions: The proposed symmetry method surpasses recent state-of-the-art SBD methods, demonstrating increased accuracy for both cut and gradual transitions in videos.

Keywords: shot boundary detection (SBD); cut transition; gradual transition; candidate segments; Speeded-Up Robust Feature (SURF); adaptive threshold

Citation: M., R.S.; A., K.; S., A. The Detection of Video Shot Transitions Based on Primary Segments Using the Adaptive Threshold of Colour-Based Histogram Differences and Candidate Segments Using the SURF Feature Descriptor. *Symmetry* **2022**, *14*, 2041. https://doi.org/10.3390/sym14102041

Academic Editors: Dumitru Baleanu and Chin-Ling Chen

Received: 18 August 2022
Accepted: 21 September 2022
Published: 30 September 2022

1. Introduction

Content-Based Video Indexing and Retrieval (CBVIR) offers automated video indexing, retrieval, and management. A content-based video search system provides users results pertaining to the video queries raised by searching the relevant video databases. The different categories of video content available demand different techniques for their retrieval. Video content can be broadly categorized into two groups: professional and user-generated. The former refers to the news, TV programs, documentaries, movies, sports, cartoons, videos with graphical and editing features accessible on the World Wide Web, YouTube, and digital media repositories such as Netflix. The latter refers to videos recorded by individuals at home or outdoors, covering events using a camera or smartphone, and made widely available on social media platforms and users' personal Google Drives.

CBVIR has applications in intelligent video surveillance, event management, folder browsing, summarization, and keyframe extraction. [1] Shot boundary detection is basic to CBVIR applications [2].

A video has a fundamentally hierarchical structure, and a video shot is a collection of image sequences continuously captured by a single camera with no interruptions. Video shots are combined to form meaningful scenes, and a collection of such scenes culminates in a video. Figure 1 shows a hierarchical structure of a video.

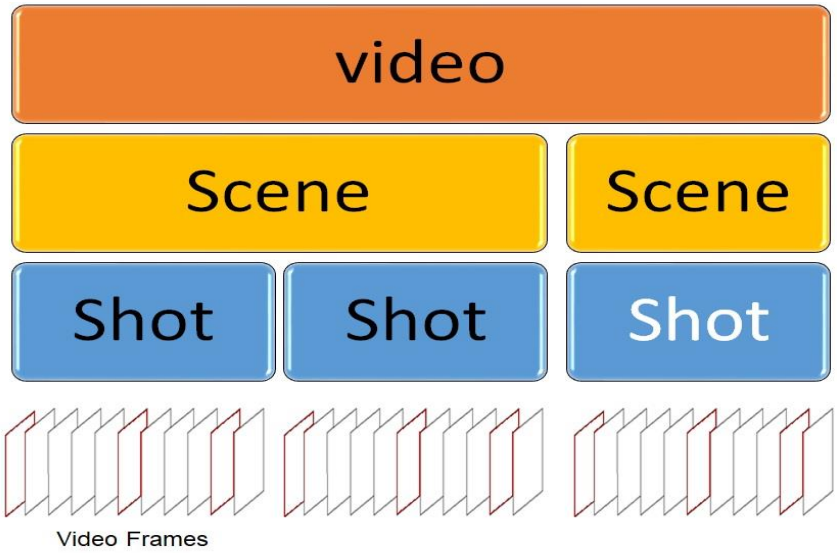

Figure 1. Hierarchical structure of a video.

Cuts or abrupt transitions occur when video shots are combined together without editing effects. A gradual transition refers to a transition sequence from one shot to another, created using editing effects. Gradual transitions include dissolve, fade in/out and wipe. Fade refers to the transition of a video frame to a single colour, white or black. Fade-in and fade-out usually occur at the commencement and end of a shot. Dissolve is a transition from one shot to another, made possible by superimposing the pixels of the two shots. Wipe uses geometrical patterns such as lines and diamonds to transition from one shot to another. Video shot boundary detection is the process of identifying the video frame in which a shot change occurs. Figure 2 shows different video shot transitions.

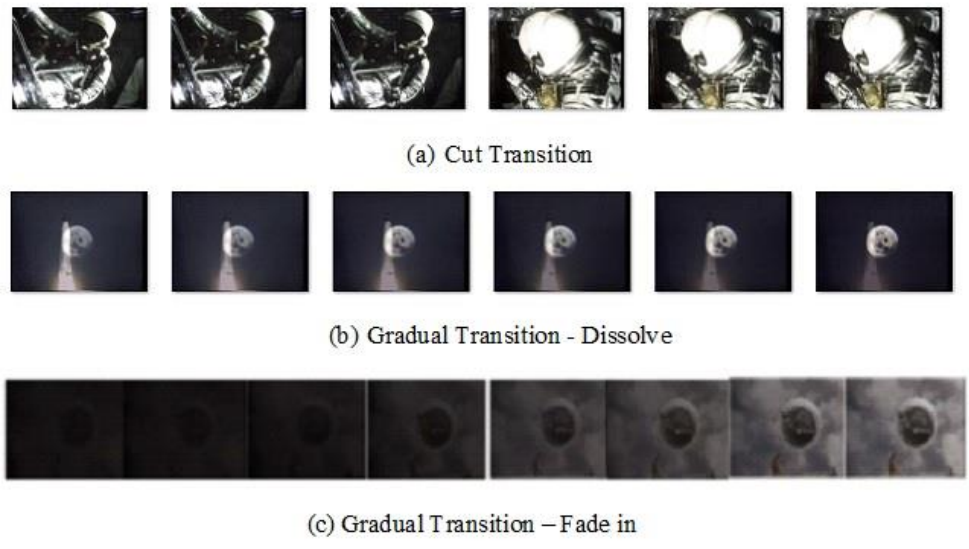

Figure 2. Types of shot transitions.

Frames within a shot share much more similar visual content than those at the boundary. The frame features lying at the transition boundary show greater change than the rest. The first step in shot boundary detection is to extract features like colour, edges, frame intensity, motion, texture and SURF from the frames. The next step is to measure feature discontinuity between consecutive frames. When the discontinuities exceed the threshold value, a shot transition is identified. For hard cut transitions, the dissimilarity values between consecutive video frames are large, whereas, for gradual transitions, the small values increase the complexity of the gradual transition detection process.

Numerous SBD methods have been proposed in the last two decades. Nevertheless, video segmentation methods face challenges in detecting gradual transitions, owing to raw video content, changes in illumination, camera and object motion, fast object motion, as well as camera operations such as panning, zooming and tilting. Researchers have used global features and local descriptors to analyse dissimilarities between video frames, and they show promising results in detecting cut and gradual transitions. However, extracting the local features of an image for every video frame, increases the time taken for feature extraction and computational complexity.

The proposed research concentrates on reducing excessive feature extraction and identifying gradual transitions by analysing the transition behaviour of the video frames. The proposed method takes every video frame from an input video and extracts the HSV colour histogram features from them. The dissimilarity value of colour characteristics is calculated using the histogram differences between successive video frames. Utilizing the mean and standard deviation of the dissimilarity colour characteristics, an adaptive threshold is constructed. When dissimilarity values exceed the adaptive threshold, primary segments are determined. Next, the adaptive threshold within each primary segment's boundaries is computed and each primary segment is examined. Either the transition is identified as a cut transition, or the method tends to divide the primary segment into candidate segments if any dissimilarity colour feature does not exceed the local adaptive threshold. Each candidate segment is then analysed further by computing the adaptive threshold local to it. If the SURF matching between the candidate border is more than or equal to 0.5, a cut transition is recognized. If the SURF matching score falls below 0.5, a gradual transition is identified. On the TRECVID 2001 video dataset, the proposed method's performance is measured by recall, precision, and F1 score, and its results are contrasted with those of the existing approaches.

2. Related Work

Boreczky et al. [3] briefly discussed a few methods to identify cut and gradual transitions in videos. It was concluded that techniques such as running dissimilarity between frames, motion vector analysis and region-based comparisons offered better results than histogram-based methods. Cotsaces et al. [4] reviewed different features and methodologies for shot transition detection, all of which were found to be inefficient for fast shot change. It was noted, however, that object detection and analysis may be explored for accurate shot change detection. Hussain et al. [5] presented a detailed survey on recent SBD methods and concluded that both algorithmic accuracy and computational complexity are mutually dependent on one another. Open challenges to be overcome include changes in illumination, camera/object motion, and camera operations.

Algorithms for shot boundary detection are categorized into two groups, compressed and uncompressed, based on the types of features extracted from video frames. Figure 3 shows the categorization of video shot segmentation methods. In the compressed domain, features extracted from video frames include motion vectors, macroblocks, DCT coefficients and motion compensation [6,7]. Shot boundary detection (SBD) algorithms in the compressed domain depend on the compression standard. Despite the quick computation time involved, system performance is compromised [8], which is a major disadvantage of the domain. In the uncompressed domain, on the other hand, the large volume of visual information present in video frames [9] helps design an SBD algorithm that works well.

In this domain, the basic features extracted from video frames include pixel differences, colour, edge detection, texture, motion, and local feature descriptors such as SURF and SIFT. The extensive feature extraction steps employed increase computation time. There is, consequently, a need to design robust SBD algorithms which are balanced in terms of computation time and accuracy in detecting cut and gradual transitions, and ones that can be used in real-time applications as well. This section reviews the work done on the uncompressed domain.

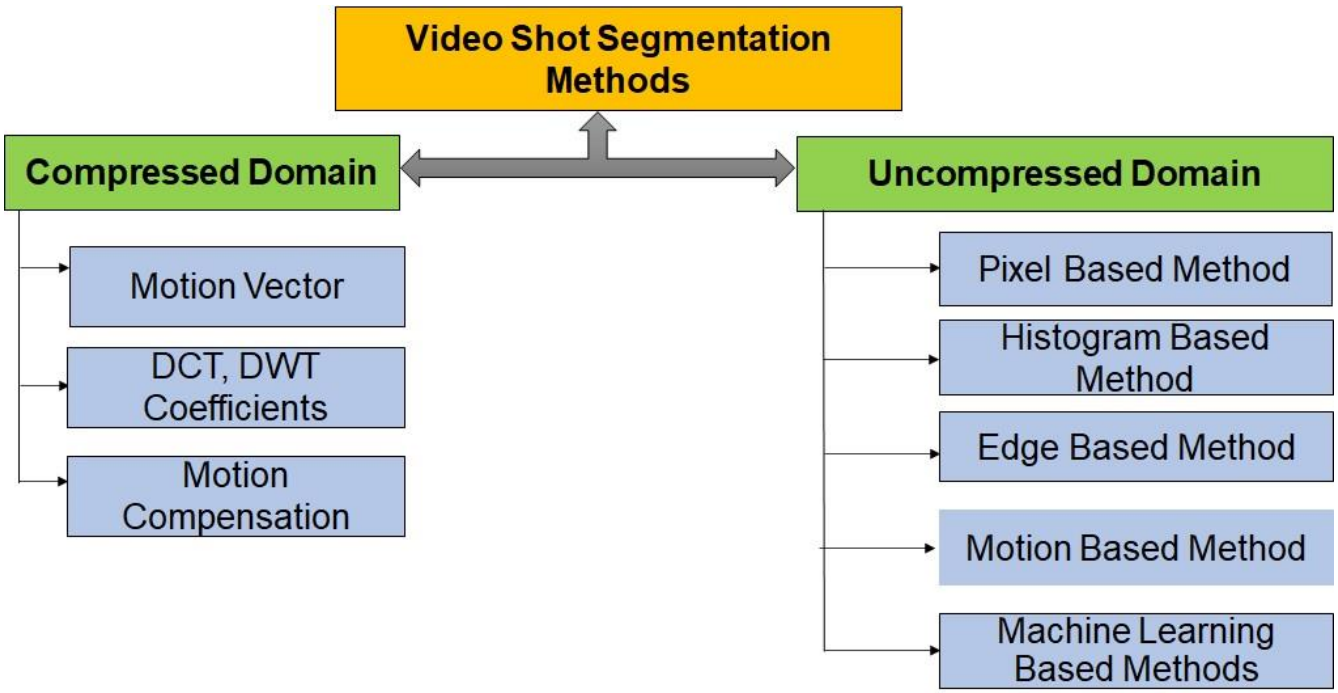

Figure 3. Categorization of Video Shot Segmentation Methods.

Pixel Differences-Based Video Shot Segmentation Methods:

Pixel-based shot boundary detection calculates the difference in pixel intensity values between consecutive video frames. Cut and gradual transitions are detected [10–12] when the difference in pixel intensities exceeds the threshold calculated. The slightest changes in illumination, camera motion and fast object motion cause large changes in pixel intensity values, resulting in false detection and misdetection of cut and gradual transitions.

Nagasaka and Tanaka et al. [10] used two thresholds for shot boundary detection. Pixel changes are identified, along with partial pixel intensity differences between consecutive video frames that exceed the first threshold. Cut transitions are detected when pixel differences exceed the second threshold value. Kikukawa et al. [11] also used a double threshold method. Their method detected a change in shot transition when the absolute differences in the pixel values exceed the first threshold. Likewise, a cut is identified when the cumulative pixel differences exceed the second threshold.

Zhang et al. [12] applied the average filter all over the video frame. The filter replaces the pixel values in the video frame with the average neighbouring pixel values, de-noises the image, and lessens the camera motion effect. Given that manual thresholds are used, even small changes in illumination, camera motion, and fast object motion show large changes in pixel intensity values, resulting in false detection and misdetection of cut and gradual transitions.

Histogram-Based Shot Segmentation Methods:

Histogram-based SBD algorithms generate a histogram of colour spaces from consecutive video frames. Thereafter, differences between the histogram of consecutive video frames are arrived at using metrics such as histogram intersection as well as histogram

differences using the chi-square distance, X^2 distance, Bhattacharya distance and cosine measures. Cuts and gradual transitions are identified when the histogram differences exceed the threshold values. Both manual and adaptive thresholds are used in the literature.

The experimental outcomes show that the adaptive threshold offers a good trade-off between the precision and recall values of transition detection [13]. Certain studies have employed twin or double thresholds for transition detection in order to reduce the incidence of false detection. A lot of work has been carried out with different colour spaces such as the grayscale [14], RGB [15,16], HSV [17,18], YSV [19], and L*a*b* [20]. However, histogram differences are still sensitive to large-object motion and camera motion such as panning, zooming and tilting, as well as to changes in illumination [21], resulting in increased false positive rates. Baber et al. [14] identified cuts in a video by ascertaining the entropy of grayscale intensity differences between consecutive frames with high and low thresholds. False cut detections were fine-tuned by the SURF feature descriptors of the respective frames. Further, fade-in and fade-out effects were identified by analysing the change in the entropy of the grey intensity values for each frame, using a run-length encoding (RLE) scheme to obtain 97.8% precision and 99.3% recall for the overall transition. Bendraou et al. [15] discovered abrupt or cut transitions in a video by determining RGB colour histogram differences between consecutive frames with the adaptive threshold. Dissimilarity values falling between the low and high adaptive thresholds were taken as candidate frames. SURF feature-matching was carried out for the frames to eliminate false cut detection, resulting in a 100% recall criterion.

Apostolidis et al. [16] arrived at a similarity score for successive video frames. The similarity score is a combination of the normalized correlation of an HSV colour histogram using the Bhattacharyya distance and the SURF descriptor matching score. Cuts were initially identified by comparing the similarity score with a small predefined threshold. Gradual transitions were identified by analysing transitions in the calculated similarity score. Tippaya et al. [18] found the similarity score for each video frame as in [15] while using the RGB colour histogram and the Pearson correlation coefficient differences. Candidate segments that were found by setting the segment length at 16 were processed for cut and gradual transitions only if the similarity score was greater than the local adaptive threshold of the segment, using an AUC analysis of the similarity values.

Hannane et al. [22] extracted a SIFT point distribution histogram from each video frame and computed the distance of the SIFT-PDH for consecutive video frames, using the adaptive threshold to detect both cut and gradual transitions. The computational complexity was reduced, and the scheme was robust to camera motion and fast object motion, as SIFT features are invariant to these conditions. Tippaya et al. [23] proposed an approach that utilized multimodal visual features based on cut and gradual shot detection. Candidate segment selection was used without the adaptive threshold to reduce computational complexity. RGB histograms, SURF features, and peak feature values from the transition behaviour of the video were extracted for cut and gradual transitions.

Edge-Based Shot Segmentation Method:

Edge-based shot detection algorithms calculate the number of edge features between consecutive frames by ascertaining the difference in edge positions between consecutive frames as the edge change ratio. Shot boundaries are detected when edge features exceed the threshold value. This method, though more expensive, does not work as well as histogram-based methods [24]. Edge-based methods are, however, invariant to changes in illumination. Consequently, edge features are used to detect flashlights in video frames [25] and filter candidate frames to spot cut and gradual transitions. Further, fade-ins and fade-outs are identified when no edges are detected in the video frames [26].

Motion-Based Video Shot Segmentation Methods:

Motion-based shot boundary algorithms compute motion vectors between consecutive video frames in the compressed domain using block-matching algorithms. The process involves dividing a video frame into blocks, and the block-matching algorithm estimates motion by comparing every block in the current frame with all the blocks in the next

frame. Motion estimation detects camera motion easily, thus increasing SBD accuracy. With this method, however, the computation cost incurred and motion vector time taken are excessive in the uncompressed domain of videos. There is, in addition, decreased accuracy with inappropriate motion vector selections by the block-matching algorithms. Priya et al. [27] calculated motion vectors from video frames using the block-matching algorithm proposed in [28], and maximized motion strength through texture, edge and colour features, using the Walshard transform weighted method. The shot boundary is detected through a change in motion strength when the threshold is exceeded.

Machine Learning-Based Shot Segmentation Method:

All of the earlier methods mentioned above use the threshold-based classification of shot boundaries as hard cut and gradual transitions. Machine learning algorithms such as support vector machines and neural networks offer an alternative classification method. Following feature extraction and estimation of feature dissimilarity from consecutive video frames, the classifiers are to be trained with the said features. It is critical that the classifier be trained with a balanced training dataset, because a training set with additional cut features will bias the classifier output in terms of only cut detection, thus decreasing SBD accuracy.

Sasithradevi et al. [29] created a balanced training set to train their bagged tree classifiers, and extracted pyramid-level opponent colour space features from video frames for candidate segment selection. Mondal et al. [30] extracted features from frames using the Non-Subsampled Contourlet Transform (NSCT) for a multiscale geometric analysis. The principal component analysis was used for feature reduction, and the least squares-support vector machine classifier was trained with a balanced dataset to reduce the class imbalance problem and enhance SBD system performance. Thounaojam et al. [31] used fuzzy logic and genetic algorithms in their work, while Yazdi et al. [32] applied the K-means clustering algorithm to identify initial cut transitions and extract features using the SVM classifier to detect gradual transitions. Xu et al. [33] used the deep learning convolutional neural network algorithm for shot transition detection. All of the machine learning and deep learning algorithm-based methods above depend on training time and training dataset quality, both of which greatly impact the performance of shot transition detection systems.

Hassanien et al. [34] proposed the DeepSBD algorithm using a three dimensional spatio-temporal convolutional neural network on a very large video dataset. The large video dataset consists of 4.7 million frames, of which 3.9 million frames are from TRECVID videos from 2001 to 2007 datasets, and the rest are synthesised consisting of both cut and gradual transition. They achieved an 11% increase in shot transition detection compared to other state-of-the-art methods.

Soucek et al. [35] proposed TransNet, an extensible architecture that uses various dilated 3D convolutional processes per layer to get a larger field of vision with fewer trainable parameters. The network accepts a series of N successive video frames as input and applies a number of 3D convolutions in order to deliver a prediction for each frame in the input. Each prediction represents the probability that a particular image is a shot boundary. When the prediction falls below a threshold, the shot begins at the first frame, and when the forecast rises over the threshold, the shot stops at the first frame. Due to the availability of a number of pre-set temporal segments, the TRECVID IACC.3 collection has been used. On a single strong GPU, the network works more than 100 times faster than real-time. Without any further post-processing and with only a small number of learnable parameters, the TransNet performs on par with the most recent state-of-the-art method.

B. Ming et al [36]. proposed a method for detecting shot boundary using image similarity with a deep residual network for gradual shots and mutation shots. For mutation shots, the image similarity feature performs better segmentation, although it performs poorly at gradual shot detection. Although the deep neural network requires more computation, it can segment gradual shots more accurately. The initial video boundary candidate points are extracted in this paper using a multitude of deep features of the video frame and the concept of image mutual information. The method is evaluated using Clipshots dataset.

For VR movies and user-generated videos on the internet, this method has better shot detection results. The importance of colourization property is discussed in [37].

Remarks on the Video Shot Segmentation Methods reviewed:

Various approaches have been mentioned in the literature for video shot segmentation which produces less F-Score for detecting cut and gradual transition. Global features and local descriptors have been used to analyse dissimilarity in video frames with promising results in the detection of cut and gradual transitions. Extracting the local features of an image for every video frame [14,18,19], increases the feature extraction time and the computational complexity. The problems that need to be addressed in order to develop efficient and robust video shot segmentation are listed below.

Problem 1: Develop an efficient video shot segmentation method that results in an improved F-score for gradual transition

- Much of the work discussed produced a satisfactory F-score in detecting cut transitions, though there was comparatively little accuracy in detecting gradual transitions.
- The detection of gradual transitions often fails, owing to changes in illumination, fast moving objects, and camera motion.

Problem 2: Reduce the computational complexity of the algorithm while simultaneously having a good trade-off for improved shot transition detection efficiency

- Every approach studied in the literature extracts bulk features from videos. The process enhances the detection of cut and gradual transitions to maximize the computational complexity of the method.
- Although video shot segmentation methods developed using deep neural networks provide higher detection accuracy, the model must be trained with a balanced dataset. For better performance, the deep network architect must be fine-tuned for each video category. This raises the cost of developing the robust shot segmentation method using deep learning methods that can be applied to all types of videos.

Hence, an efficient video shot segmentation method is needed to detect shot boundaries with an improved F-score for both cut and gradual transitions and, further, reduce the overall computational complexity involved in their detection through analysing video frame transition behaviour.

The remainder of this paper is structured as follows: Section 3 provides a detailed explanation of the proposed video shot segmentation method. Section 4 provides a detailed explanation of the experiments carried out using the proposed method. Section 5 discusses the obtained results and their comparison with existing methods. Section 6 provides the research paper's conclusion, including limitations and future improvements.

3. Proposed System

The current study aims to develop a symmetry algorithm by analysing transition behaviour through applying global feature extraction and fine-tuning cut and gradual transition detection using the local features of the frames in question.

A video that is given as input to the algorithm is converted into frames to begin with. Primary segments in the video frames are distinguished by comparing the histogram differences of consecutive video frames to the adaptive thresholds calculated for the entire length of the video. In each primary segment, candidate segments are identified when the histogram differences of consecutive video frames exceed the local adaptive threshold calculated within the boundaries of the primary segment concerned. Finally, each candidate segment is analysed by comparing the SURF feature-matching of the boundary frames with the number of peak values in the candidate segments. A cut transition is identified when the matching rate is equal to or greater than 0.5, with no peak between candidate boundaries. A gradual transition is identified when the SURF feature matching rate is less than 0.5 and there exists a local peak value between candidate boundaries. The architecture of the proposed system is shown in Figure 4.

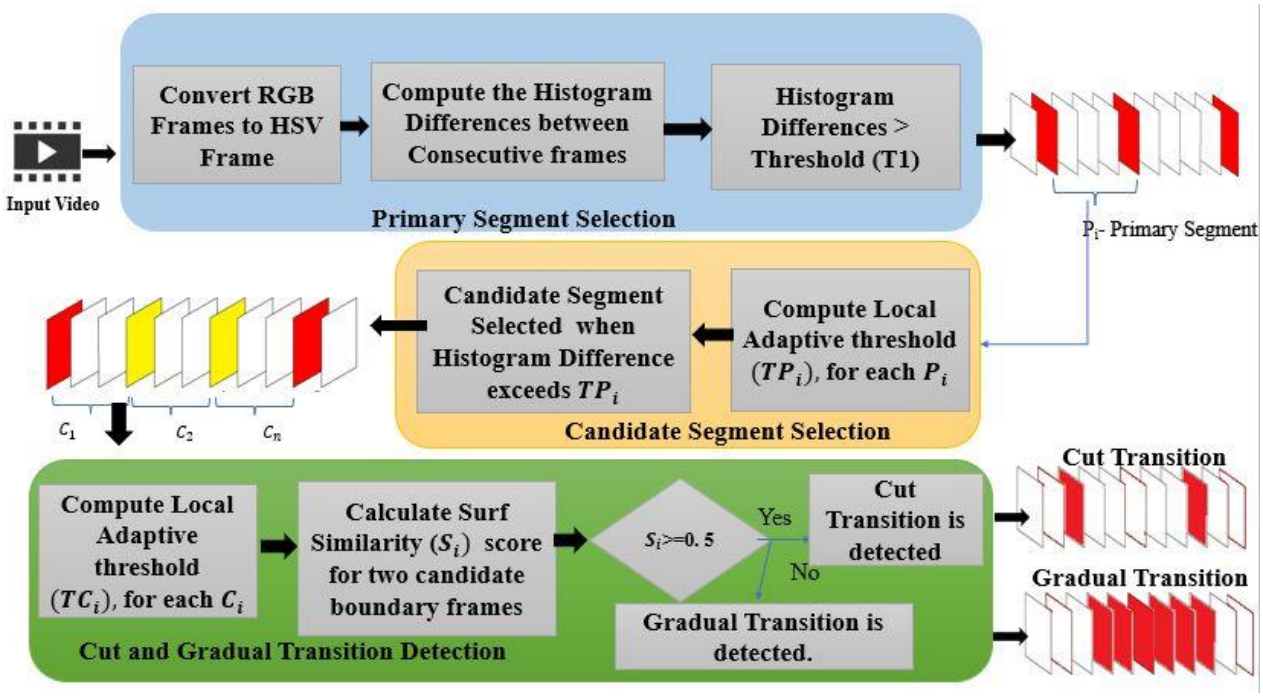

Figure 4. System Architecture.

3.1. Primary Segment Detection

The video is converted into frames and visual features are extracted from them. The global feature, the most widely used of which is the colour histogram, refers to the colour information derived uniformly from every video frame. A colour histogram represents the colour distribution in an image in which each histogram bin gives a pixel frequency representing a particular colour from a 2D or 3D colour space. The HSV colour model is chosen precisely because it replicates the way the human eye perceives colour.

The video frames are converted from the RGB to the HSV colour space. The HSV colour histogram is calculated for all the video frames. Distance measures such as the Bhattacharya, Euclidean, Manhattan and chi-square help to calculate the similarity between consecutive video frames. This work calculates the histogram difference between consecutive video frames using the chi-square (χ^2—distance) distance measure to enhance HSV colour histogram differences. The histogram distance between two frames, F_i and F_{i+1}, is given in Equation (1)

$$D(F_i, F_{i+1}) = \sum_{n=1}^{n} (H_i\,(\mathrm{n}) - H_{\,i+1}(\mathrm{n}))/H_{i(n)} \quad (1)$$

where n is the bin index of the histograms of H_i and H_{i+1} for the current and successive frames, respectively. When the histogram differences exceed the threshold value, a primary segment is identified all the way up to frame F_i. The threshold value used in this work is the adaptive threshold, which offers a good trade-off between the recall and precision values [20]. The adaptive threshold is calculated using the formula in Equation (2)

$$\text{Adaptive threshold, A} = \mu + \alpha\sigma \quad (2)$$

where μ is the mean of all the histogram differences calculated for the entire length of the video, σ is the standard deviation of the histogram differences, and α is the very small positive value that is determined using repeated experimental methods. By the end of this module, a video of length N is divided into M number of primary segments, following which each primary segment is analysed for candidate segment identification.

3.2. Candidate Segment Detection

In each primary segment, P_i with the shot boundaries, i and i+n, is analysed to identify its candidate segments. For each primary segment, the local adaptive threshold is calculated when the histogram differences of the consecutive video frames, F_i and F_{i+1}, exceed the threshold, T_i. A candidate segment is thereafter identified at frame F_i. Candidate boundaries are marked by a starting frame and the frame at which the histogram differences exceed the local threshold. In this way, a primary segment is further subdivided into N candidate segments. Another scenario is also possible, where no candidate segment is identified in the given primary segment. In such a case, the last frame of the primary segment is the frame in which a hard-cut transition is discerned.

3.3. Cut and Gradual Transitions Detection

Features specific to each video frame constitute the local features. This study utilizes the local feature, Speeded-Up Robust Feature (SURF) [38], which is invariant to rotation, blur, changes in illumination and scale, and affine transformations. SURF targets key points in an image that are invariant to the above, along with local descriptive information (also termed local descriptors) on the key points. SURF features are extracted only from selected video frames for fine-tuning cut and gradual transitions.

Cut and gradual transitions are identified by analysing each candidate segment, $C_{i,}$ for the length of the entire video. For each candidate segment, $C_{i,}$ with frame boundaries F_i and F_{i+1}, a local adaptive threshold, $T_{C,}$ is calculated. When the histogram differences of consecutive frames inside the candidate segment exceed the local adaptive threshold, a transition is identified. SURF features are extracted for the video frames F_i and F_{i+1}, where F_i is the starting frame and F_{i+1} is the frame in which a transition is identified. The transition is established as a cut or a gradual, based on the value of the similarity score falling into one of the following categories.

Case 1: If the SURF similarity score lies below 0.5, the two extreme frames are totally different and the algorithm tends to calculate, once again, a local threshold value between the extreme frames. If there is no histogram difference between consecutive video frames that exceeds the local threshold value, it is assumed that no peak values lie between the extreme frames, F_i and F_{i+1}, and the transition is declared a hard cut.

Case 2: If the SURF similarity score equals 0.5, the algorithm tends to calculate, yet again, a local threshold value between the extreme frames. Again, if the histogram difference between consecutive video frames exceeds the local threshold value, it is assumed that the peak values lie between the extreme frames. If more than five peak values lie between the extreme frames, F_i and F_{i+1} the transition is declared to be gradual. The proposed algorithm is given in Table 1.

Table 1. A description of the TRECVID 2001 video dataset.

Video Name	Total No of Frames	No of Cuts	No of Gradual	Total no of Transitions
anni005	11,364	38	27	65
anni006	16,586	41	31	72
anni009	12,307	38	65	103
anni010	31,389	98	55	153
bor03	48,451	231	11	242
bor08	50,569	380	151	531
nad31	52,405	187	55	242
nad33	49,768	189	26	215
nad53	25,783	83	75	158
nad57	12,781	45	31	76
nad58	13,648	40	45	85

4. Experimental Results

The proposed algorithm is evaluated using videos from the TRECVID 2001 video dataset, sponsored by the National Institute of Standards and Technology (NIST). The test videos are available with the ground truth used for an information retrieval project. The TRECVID 2001 video dataset is downloaded from the Open Video Project and is described in Table 1. The TRECVID-2001 6.3 GB-sized video dataset contains 84 videos with a total run time of approximatively 11.2 h. The description of the videos from VideoSeg dataset [28] is presented in Table 2. The performance of the algorithm is evaluated using the metrics of precision, recall, and F1-score

$$\text{Precision} = (N_C)/(N_C + N_F) \tag{3}$$

$$\text{Recall} = (N_C)/(N_C + N_M) \tag{4}$$

$$\text{F1-Score} = (2 \times \text{recision} \times \text{Recall})/(\text{Recall} + \text{Precision}) \tag{5}$$

where N_C is the number of transitions correctly detected, N_F the number of transitions falsely detected and N_M the number of missed transition detections. F1 score is the harmonic mean of precision and recall values. Experiments were carried out on a PC with an Intel(R) Core (TM) i5-5200U CPU with memory 6.00 GB RAM and running at 2.20 GHz, as well as a 900 GB hard disc drive, using Jupyter Notebook from the Anaconda Python distribution.

Table 2. A Description of VideoSeg Dataset

Video Name	Video Title	Frames #	Cuts #
A	**Cartoon**	**649**	**7**
B	**Action**	**957**	**8**
C	**Horror**	**1619**	**54**
D	**Drama**	**2630**	**34**
E	**Science Fiction**	**535**	**30**
F	**Commercial**	**235**	**0**
G	**Commercial**	**499**	**18**
H	**Comedy/Drama**	**5132**	**38**
I	**News/Documentary**	**478**	**4**
J	**Trailer/Action**	**871**	**87**

The dataset taken into account for the experiment is made up of challenging instances with varying lighting, length, and genre, that include video editing effects as well as camera/object motion. Uncertain shot boundaries are brought on by sudden light changes, camera/object motion, and other factors. Table 1 describes the TRECVID 2001 video dataset in detail, including the total number of frames in each video as well as the number of cut and gradual transitions. The detailed step by step process of proposed method is given in Algorithm 1: ShotBoundaryDetection_UsingColourhistogram_SURF.

Table 3 presents the precision, recall and F1-score for both the cut and gradual transitions detected by the novel proposed algorithm based on the colour histogram and SURF local feature. The average precision, recall and F1-score for the cut transition are 99.1%, 98.23% and 99%, respectively, while the same for the gradual transition are 90%, 91.9% and 90.8%, respectively.

Algorithm 1: ShotBoundaryDetection_UsingColourhistogram_SURF

Input: Video, V
Output: Cut and Gradual Transitions
Procedure: shot_boudary_detection(V)
begin
Video, V converted into HSV frames (hf1, hf2, hf3, ..., hfn)
// Calculate HSV Histogram for all frames of V
for i=1 to Len (V) do
$Hist_i \leftarrow$ HSV_Histogram(V)
// Calculate Histogram difference for consecutive frames
for i = 1 to Len $(Hist_i)$ do
$HistDiff_i \leftarrow$ Hist (hfi)– Hist (hfi+1)
TH adaptive threshold(V)
// Identify Primary Segment
for i = 1 to Len $(HistDiff_i)$ do
if $HistDiff_i > TH$ then
Declare a Primary Segment between i^{th} frame to $(i+1)^{\text{th}}$ frame
primary_segment_boundaries (v) $\leftarrow$ i /frame number
$th_i \leftarrow primary\_threshold(hf_i, hf_{i+n})$ //n=previous boundary–current
boundary
end if
end for
// Identify Candidate Segment
for each Primary Segment (hf_i, hf_{i+n}) do
if $HistDiff_i > th_i$ then
Declare a candidate segment boundary i^{th} frame to $(i+1)^{th}$ frame
candidate_segment_boundaries (v) $\leftarrow$ i /frame number
$th_i \leftarrow candidate\_threshold(hf_i, hf_{i+n})$
end if
end for
// Cut and Gradual Transition Detection
for each Candidate Segment (hf_i, hf_{i+n}) do
$th_i \leftarrow local\_candidate\_threshold(hf_i, hf_{i+n})$
Calculate SURF matching score between f_i *and* f_{i+n}
if $(\text{SURF}_{\text{score}} < 0.5)$&& $(no\ th_i\ exists)$ *then*
Detect a Cut transition at hf_{i+n} frame
else if $(\text{SURF}_{\text{score}} \geq 0.5)$&&($th_i\ exists$) then
Calculate the number of peaks lies between hf_i *and* hf_{i+n}
Analyze each peak in the candidate boundaries
Detect a Gradual transition between hf_i *and* hf_{i+n}
end if
end if
end for
end procedure

Table 3. Cut Transition and Gradual Transition Detection by the Proposed Algorithm.

TREC 2001 Video	Proposed Algorithm								
	Cut Transition			Gradual Detection			Overall		
	Precision	Recall	F1_Score	Precision	Recall	F1_Score	Precision	Recall	F1_Score
anni005	97.4	100	98.7	96	92.3	94.1	96.7	96.2	96.4
anni006	100	97.6	98.8	92.3	88.9	90.6	96.2	93.3	94.7
anni009	97.3	100	98.6	92.7	91.1	91.9	95.0	95.6	95.3
anni010	100	99	99.5	89.1	100	94.2	94.6	99.5	96.9
bor03	99.6	97.8	98.7	80	88.9	84.2	89.8	93.4	91.5
bor08	100	99.5	99.7	95.9	96.6	96.2	98.0	98.1	98.0
nad31	99.5	97.3	98.4	89.8	88	88.9	94.7	92.7	93.7
nad33	98.9	96.3	97.6	80	95.2	87	89.5	95.8	92.3
nad53	97.5	96.3	96.9	94.2	91.5	92.9	95.9	93.9	94.9
nad57	100	97.6	98.8	87.5	87.5	87.5	93.8	92.6	93.2
nad58	100	100	100	92.7	90.5	91.6	96.4	95.3	95.8
Average	99.1	98.3	**98.7**	90	91.9	**90.8**	94.6	95.1	**94.8**

5. Discussion

Using Equation 2, the proposed method computes the adaptive threshold for identifying the primary segments. The value of α is an experimentally determined constant that provides a good trade-off between detection precision and recall. Figure 5 depicts the precision/recall trade-off for various values of α for the anni005 video from the TRECVID 2000 dataset. The value is fixed with maximum precision and recall. Similarly, the α for threshold calculation for each candidate segment is determined empirically.

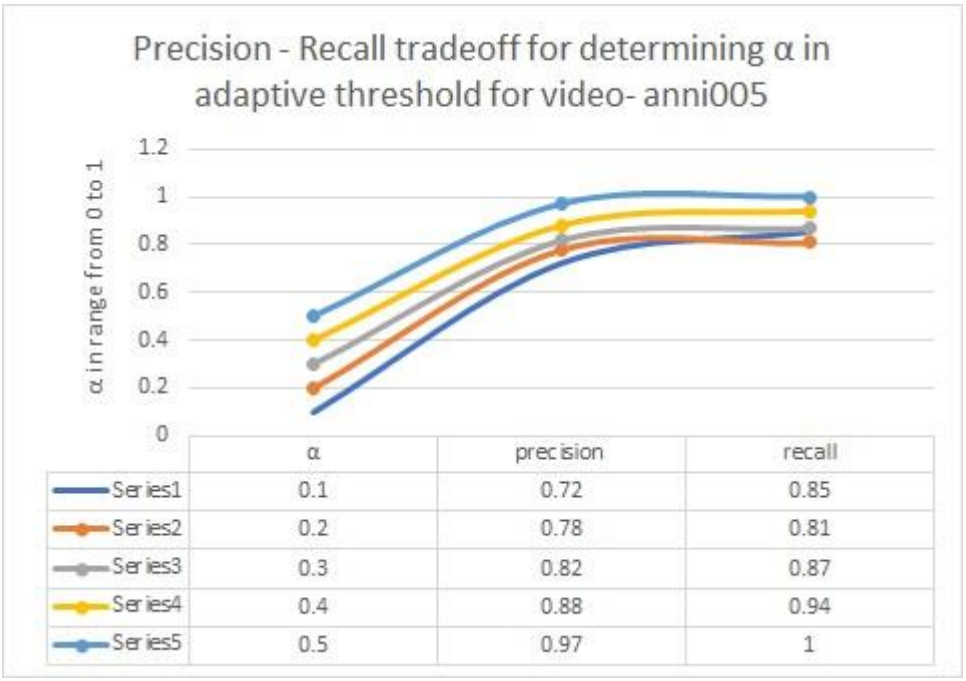

Figure 5. Precision—Recall trade-off for determining α in adaptive threshold for primary segment.

Table 4 compares the proposed method for cut transition detection on the VideoSeg Dataset to that of pyramidal opponent method [29]. The proposed method yields a high F1 score of 98.8 %, whereas [29] yields a score of 97.1%. Figure 6 depicts a graphical comparison of precision, recall, and F1 score. This demonstrates the importance of analysing

the transition behaviour of video frames in the proposed method versus the bagged tree classifiers in the existing method.

Table 4. Performance comparison of the proposed method with pyramidal opponent method for cut detection on VideoSeg Dataset.

Video	Proposed Method			Pyramidal Opponent Method		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
A	100	100	100.0	100	100	100
B	100	93.12	96.4	100	88.9	94.19
C	98.23	98.1	98.2	98.11	96.3	97.2
D	99	99.5	99.2	97.06	94.29	95.65
E	95.5	100	97.7	96.67	93.55	95.08
F	100	100	100.0	100	100	100
G	97.32	99.6	98.4	94.44	100	97.14
H	100	99	99.5	97.37	92.5	94.87
I	100	100	100.0	100	100	100
J	97.5	99.23	98.4	95.4	98.81	97.08
AVERAGE	**98.8**	**98.9**	**98.8**	97.9	96.4	97.1

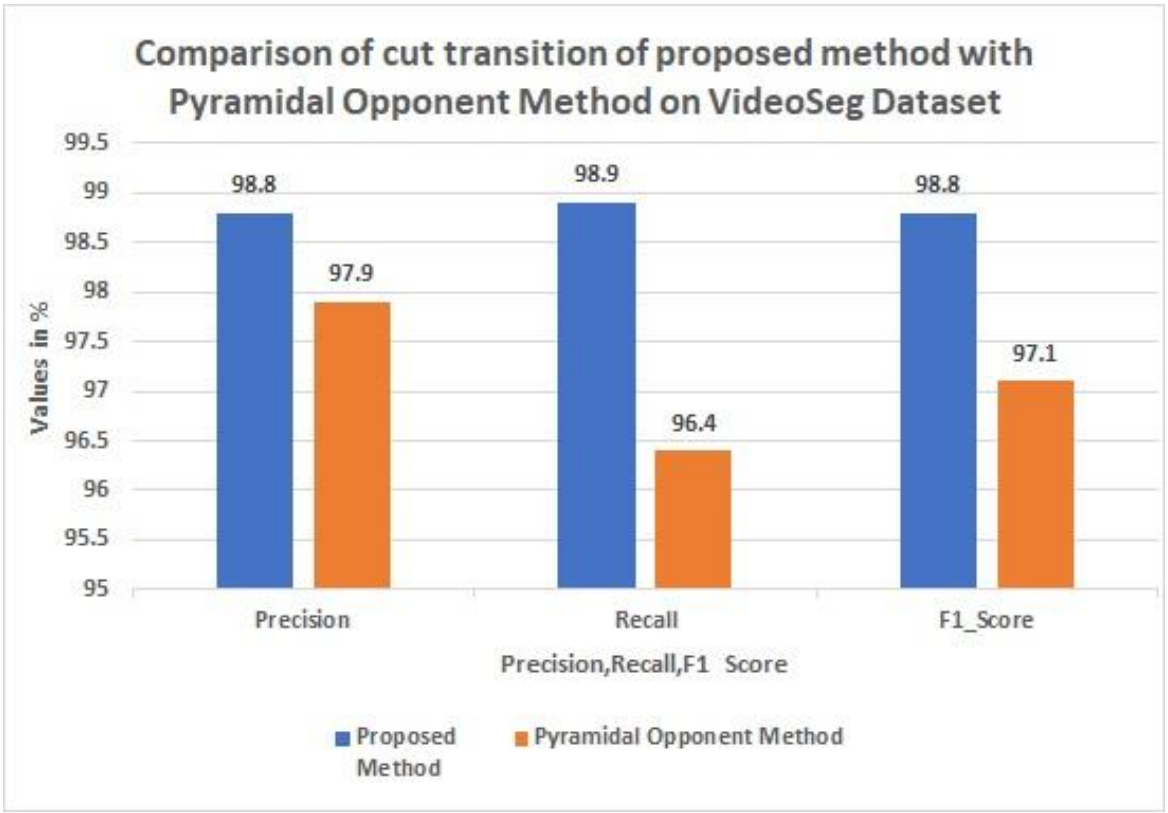

Figure 6. Comparison of precision, recall &F1 score for cut transitions by proposed algorithm with pyramidal opponent method on VideoSeg Dataset.

In Table 5, the proposed method is compared with the Multilevel Feature Collaboration Method [39] using precision, recall and F1 measures. Our proposed method offers the highest precision, recall and F1 score for both cut and gradual transitions, with the highest F1 score of 90.8% for gradual transitions. This is because the method tracks minute changes between consecutive video frames, with a recursive calculation of the adaptive threshold for each candidate segment, until it finds a group of frames with no peak value changes within the boundary. Figure 7 is the graphical comparison of precision, recall and F1 score

for detecting cut transition of the proposed method with [39]. Figure 8 is the graphical comparison of precision, recall and F1 score for detecting gradual transition of the proposed method with [39].

Table 5. A Comparison of the Proposed System with the Multilevel Feature Collaboration Method

Video	Multilevel Feature Collaboration Method						Proposed Method					
	Cut Transition			Gradual Transition			Cut Transition			Gradual Transition		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
anni005	85	85	85	84	77.8	80.8	97.4	100	98.7	96	92.3	94.1
anni009	94.3	84.6	89.2	88.5	69.7	78	97.3	100	98.6	92.7	91.1	91.9
bor08	96.3	83.6	89.5	81.7	53.8	64.9	100.0	99.5	99.7	95.9	96.6	96.2
nad53	96.4	97.6	97	84.6	76.9	80.5	97.5	96.3	96.9	94.2	91.5	92.9
nad57	100	95.6	97.7	72.4	84	77.8	100.0	97.6	98.8	87.5	87.5	87.5
nad58	100	86.5	92.8	81.4	83.3	82.4	100	100	100	92.7	90.5	91.6
Average	**95.3**	**88.8**	**91.9**	**82.1**	**74.25**	**77.4**	**98.7**	**98.9**	**98.8**	**93.2**	**91.6**	**92.4**

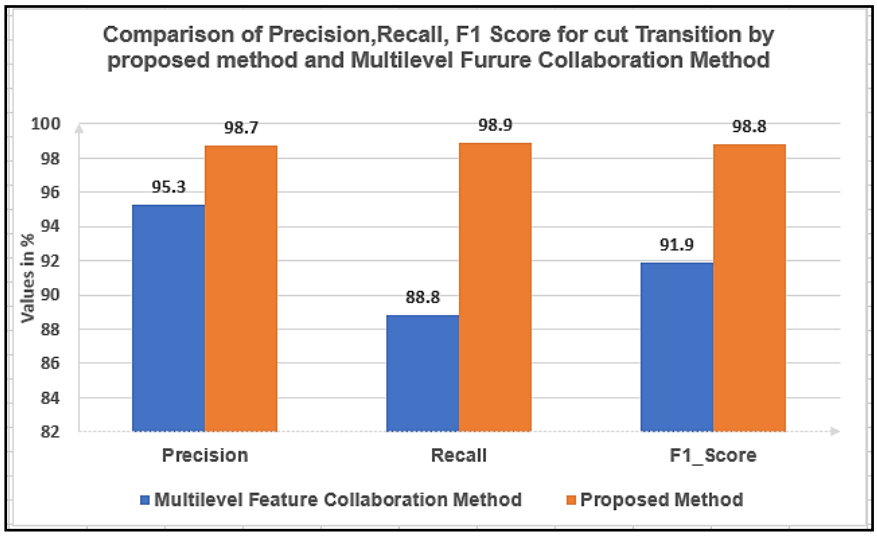

Figure 7. Comparison of Precision, Recall &F1 Score for Cut Transitions by proposed algorithm with Multilevel Feature Collaboration Method

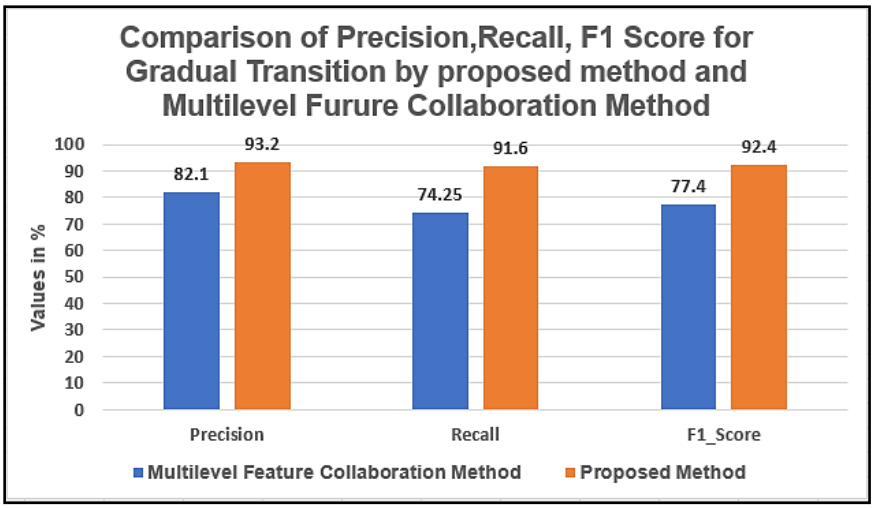

Figure 8. Comparison of Precision, Recall &F1 Score for gradual transitions by proposed algorithm with Multilevel Feature Collaboration Method

Table 6 depicts a quantitative comparison of the proposed method with the Multi-Modal Visual Features Method [23], the Visual Colour Information Method [40], and the Walsh–Hadamard Transform Kernel-based Method [27] in terms of precision, recall and F1 score. The approach applied in [23] performs cut and gradual transitions using multimodal features, with SURF features extracted for every frame, which greatly enhances algorithmic complexity whilst ignoring changes in illumination. Compared with [23], the proposed method has the highest average F1 score of 99.2% for cut and 92.1% for gradual transitions, respectively. The Visual Colour Information Method [40] records illumination changes perfectly, using the luminance component of the L*a b colour space. The method, however, lags in terms of tracking, and the features are affected by rotation and scaling, therefore showing decreased precision and recall for gradual transitions, when compared to the proposed method with average precision and recall measures of 91.7% and 92.6%, respectively. Figure 9 shows the average precision, recall and F-score graph for a performance analysis of the proposed algorithm with existing methods.

Table 6. A Quantity Analysis of the Proposed Method on the TRECVID 2001 Dataset.

Method	Video	Cut Transition			Gradual Transition		
		P (%)	R (%)	F1(%)	P (%)	R (%)	F1(%)
Proposed Method	anni006	100	97.5	98.7	92.3	88.9	90.6
	anni009	97.2	100	98.6	92.7	91.1	91.9
	anni010	100	98.9	99.4	89.1	100.0	94.2
	nad58	100	100	100	92.7	90.5	91.6
	Average	**99.3**	**99.1**	**99.2**	**91.7**	**92.6**	**92.1**
Multi-Modal Visual Features Method [23]	anni006	97.5	92.9	95.1	77.8	90.3	83.6
	anni009	100	94.9	97.4	89.6	67.2	76.8
	anni010	97.9	96.9	97.4	65.5	65.5	65.5
	nad58	100	100	100	80.4	82.2	81.3
	Average	**98.9**	**96.2**	**97.5**	**78.3**	**76.3**	**76.8**
Visual Colour Information Method [40]	anni006	76	90.5	82.6	83.3	88.2	85.7
	anni009	100	89.7	94.6	84.8	87.5	86.2
	anni010	96.8	92.9	94.8	81.7	89.1	85.2
	nad58	95.2	100	97.6	90.9	87	88.9
	Average	**92.0**	**93.3**	**92.4**	**85.2**	**88.0**	**86.5**
Walsh–Hadamard Transform Kernel-Based Method [27]	anni006	85.4	97.6	91.1	90	87.1	88.5
	anni009	86.5	82.1	84.2	88.7	85.9	87.3
	anni010	90.6	88.8	89.7	84.6	80	82.2
	nad58	93.5	95.6	94.5	88.5	88.5	88.5
	Average	**89.0**	**91.0**	**89.9**	**88.0**	**85.4**	**86.6**

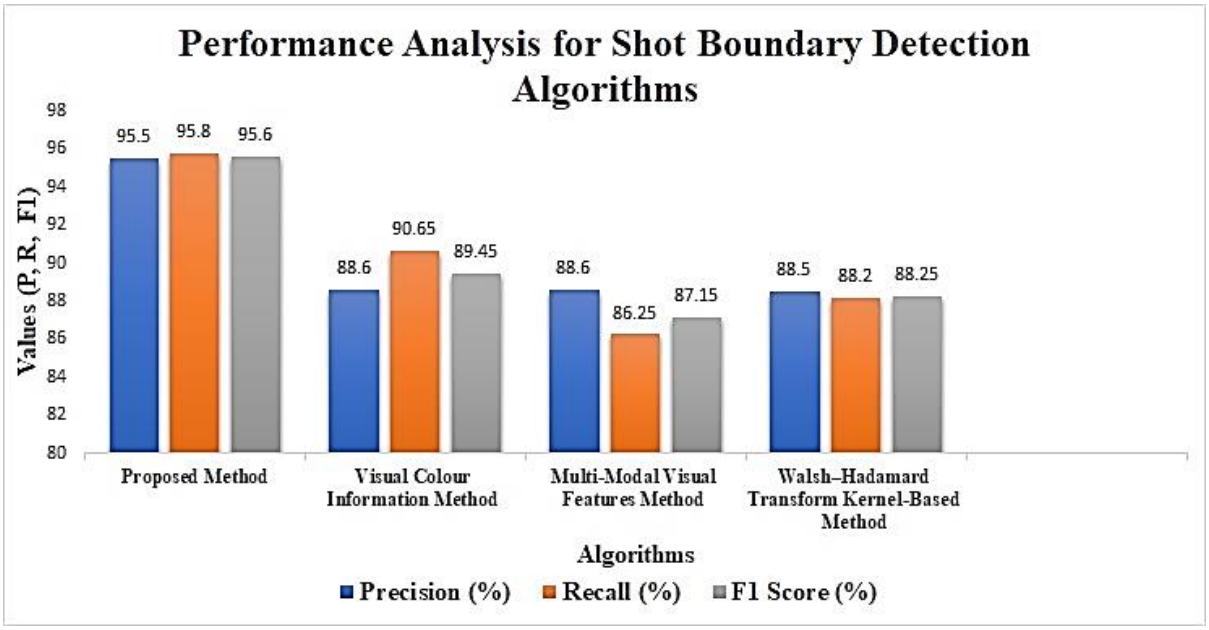

Figure 9. A Performance Analysis of the Shot Detection Algorithm.

Figures 10 and 11 depicts the quantitative comparison of the proposed method with state-of the art video shot segmentation methods from the literature for both cut and gradual transition, respectively.

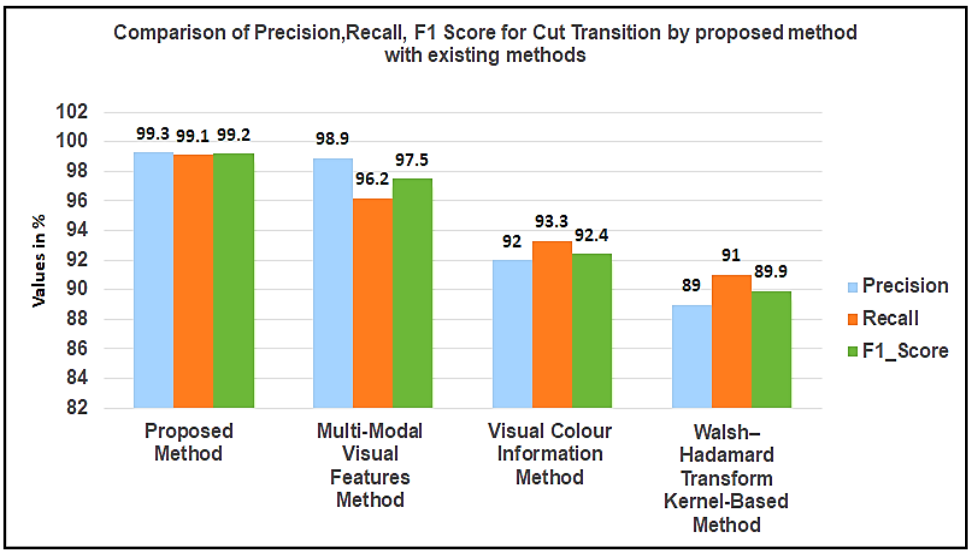

Figure 10. Comparison of Precision, Recall &F1 Score for Cut Transitions by Proposed algorithm with Existing Method.

As a result of empirical analysis on both datasets, the proposed method outperforms all sophisticated video conditions by providing a good trade-off between precision and recall values. The highest F1 score obtained demonstrates the efficacy of the proposed algorithm. The proposed method is threshold dependent, making it susceptible to camera motions such as panning and zooming, limiting the algorithm's performance. The proposed algorithm is less computationally expensive than the state-of-the-art video segmentation methods compared in the literature. The work can be improved in the future by generating automatic adaptive threshold calculations and by extending the detections of other gradual transition types such as fade-in, fade-out, and wipe.

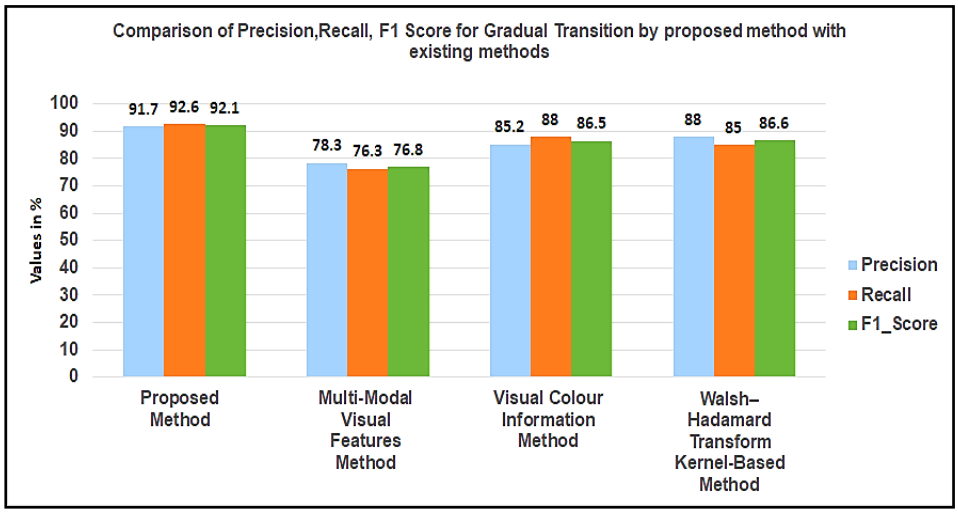

Figure 11. Comparison of Precision, Recall & F1 Score for Gradual Transitions by Proposed algorithm with Existing Method.

6. Conclusions

A simple and efficient novel shot boundary detection method has been proposed in this work. The process of analysing a full-length video is initiated by splitting it into primary segments. Global HSV colour histogram features are extracted and the adaptive threshold calculated for the entire length of the video. Each primary segment is divided into candidate segments using the HSV colour feature histogram differences and the local adaptive threshold is calculated for each primary segment boundary. Thereafter, each candidate segment is analysed by calculating the SURF matching score for candidate segment boundaries, and checked for the possibility of being split further. A candidate segment that cannot be split further is said to be a cut transition, while one that can be is considered a gradual transition. Given that the local adaptive threshold is iteratively calculated for each segment, even a minute change in the feature values is analysed. The algorithm extracts the global colour feature once for the full-length video and the SURF feature is extracted only for the candidate segment boundary frames, which greatly reduces the algorithm's complexity. The algorithm is evaluated using the standard TRECVID 2001 video dataset and compared with recent state-of-the-art SBD methods. The results show that the proposed method performs well in regard to detecting both cut and gradual transitions. The proposed work's limitation is that it only recognizes two types of transitions—cut and gradual—in the video. The current work will be extended in the future to identify other types of gradual transitions such as dissolve, fade in and fade out, and wipe.

Author Contributions: Conceptualization, R.S.M.; Visualization, R.S.M.; Writing—original draft, R.S.M.; Writing—review & editing, K.A.; Writing—review & editing, A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hu, W.; Xie, N.; Li, L.; Zeng, X.; Maybank, S. A Survey on Visual Content-Based Video Indexing and Retrieval. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2011**, *41*, 797–819.
2. Dimitrova, N.; Zhang, H.; Shahraray, B.; Sezan, I.; Huang, T.; Zakhor, A. Applications of video-content analysis and retrieval. *IEEE Multimed.* **2002**, *9*, 42–55. [CrossRef]
3. Boreczky, J.S.; Rowe, L.A. Comparison of video shot boundary detection techniques. *J. Electron. Imaging* **1996**, *5*, 122–128. [CrossRef]
4. Cotsaces, C.; Nikolaidis, N.; Pitas, I. Video shot detection and condensed representation. a review. *IEEE Signal Process. Mag.* **2006**, *23*, 28–37. [CrossRef]

5. Abdulhussain, S.H.; Ramli, A.R.; Saripan, M.I.; Mahmmod, B.M.; Al-Haddad, S.A.R.; Jassim, W.A. Methods and Challenges in Shot Boundary Detection: A Review. *Entropy* **2018**, *20*, 214. [CrossRef]
6. Pei, S.-C.; Chou, Y.-Z. Efficient MPEG compressed video analysis using macroblock type information. *IEEE Trans. Multimed.* **1999**, *1*, 321–333.
7. Nakajima, Y. A video browsing using fast scene cut detection for an efficient networked video database access. *IEICE Trans. Inf. Syst.* **1994**, *77*, 1355–1364.
8. Cooper, M.; Liu, T.; Rieffel, E. Video Segmentation via Temporal Pattern Classification. *IEEE Trans. Multimed.* **2007**, *9*, 610–618. [CrossRef]
9. Grana, C.; Cucchiara, R. Linear Transition Detection as a Unified Shot Detection Approach. *IEEE Trans. Circuits Syst. Video Technol.* **2007**, *17*, 483–489. [CrossRef]
10. Nagasaka, A.; Tanaka, Y. Automatic video indexing and full-video search for object appearances (abstract). *J. Inf. Process.* **1992**, *15*, 316.
11. Kikukawa, T.; Kawafuchi, S. Development of an automatic summary editing system for the audio-visual resources. *Trans. Electron. Inf.* **1992**, *J75-A*, 204–212.
12. Zhang, H.; Kankanhalli, A.; Smoliar, S.W. Automatic partitioning of full-motion video. *Multimed. Syst.* **1993**, *1*, 10–28. [CrossRef]
13. Park, S.; Son, J.; Kim, S.-J. Effect of adaptive thresholding on shot boundary detection performance. In Proceedings of the 2016 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia), Seoul, Korea, 26–28 October 2016; pp. 1–2.
14. Baber, J.; Afzulpurkar, N.; Dailey, M.N.; Bakhtyar, M. Shot boundary detection from videos using entropy and local descriptor. In Proceedings of the 2011 17th International Conference on Digital Signal Processing (DSP), Corfu, Greece, 6–8 July 2011; pp. 1–6.
15. Bendraou, Y.; Essannouni, F.; Aboutajdine, D.; Salam, A. Video shot boundary detection method using histogram differences and local image descriptor. In Proceedings of the 2014 Second World Conference on Complex Systems (WCCS), Agadir, Morocco, 10–12 November 2014. [CrossRef]
16. Apostolidis, E.; Mezaris, V. Fast shot segmentation combining global and local visual descriptors. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 6583–6587.
17. Zhang, Y.; Li, W.; Yang, P. Shot Boundary Detection Based on HSV Color Model. In Proceedings of the 2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP), Chongqing, China, 11–13 December 2019; pp. 1–4.
18. Tippaya, S.; Sitjongsataporn, S.; Tan, T.; Chamnongthai, K.; Khan, M. Video shot boundary detection based on candidate segment selection and transition pattern analysis. In Proceedings of the 2015 IEEE International Conference on Digital Signal Processing (DSP), Singapore, 21–24 July 2015; pp. 1025–1029.
19. Gargi, U.; Kasturi, R.; Strayer, S.H. Performance characterization of video-shot-change detection methods. *IEEE Trans. Circuits Syst. Video Technol.* **2000**, *10*, 1–13. [CrossRef]
20. Küçüktunç, O.; Güdükbay, U.; Ulusoy, Ö. Fuzzy color histogram-based video segmentation. *Comput. Vis. Image Underst.* **2010**, *114*, 125–134. [CrossRef]
21. Hanjalic, A. Shot-boundary detection: Unraveled and resolved? *IEEE Trans. Circuits Syst. Video Technol.* **2002**, *12*, 90–105. [CrossRef]
22. Hannane, R.; Elboushaki, A.; Afdel, K.; Naghabhushan, P.; Javed, M. An efficient method for video shot boundary detection and keyframe extraction using SIFT-point distribution histogram. *Int. J. Multimed. Inf. Retr.* **2016**, *5*, 89–104. [CrossRef]
23. Tippaya, S.; Sitjongsataporn, S.; Tan, T.; Khan, M.M.; Chamnongthai, K. Multi-Modal Visual Features-Based Video Shot Boundary Detection. *IEEE Access* **2017**, *5*, 12563–12575. [CrossRef]
24. Lienhart, R. Reliable transition detection in videos: A survey and practitioner's guide. *Int. J. Image Graph.* **2001**, *1*, 469–486. [CrossRef]
25. Heng, W.J.; Ngan, K.N. High accuracy flashlight scene determination for shot boundary detection. *Signal Process. Image Commun.* **2003**, *18*, 203–219. [CrossRef]
26. Zheng, J.; Zou, F.; Shi, M. An efficient algorithm for video shot boundary detection. In Proceedings of the 2004 International Symposium on Intelligent Multimedia, Video and S peech Processing, Hong Kong, China, 20–22 October 2004; pp. 266–269.
27. Lakshmi Priya, G.G.; Domnic, S. Walsh–Hadamard Transform Kernel-Based Feature Vector for Shot Boundary Detection. *IEEE Trans. Image Process.* **2014**, *23*, 5187–5197.
28. Barjatya, A. Block matching algorithms for motion estimation. *IEEE Trans. Evol. Comput.* **2004**, *8*, 225–239.
29. Sasithradevi, A.; Roomi, S.M.M. A new pyramidal opponent color-shape model based video shot boundary detection. *J. Vis. Commun. Image Represent.* **2020**, *67*, 102754. [CrossRef]
30. Mondal, J.; Kundu, M.K.; Das, S.; Chowdhury, M. Video shot boundary detection using multiscale geometric analysis of nsct and least squares support vector machine. *Multimed. Tools Appl.* **2017**, *77*, 8139–8161. [CrossRef]
31. Thounaojam, D.M.; Khelchandra, T.; Singh, K.M.; Roy, S. A Genetic Algorithm and Fuzzy Logic Approach for Video Shot Boundary Detection. *Comput. Intell. Neurosci.* **2016**, *2016*, 8469428. [CrossRef] [PubMed]
32. Yazdi, M.; Fani, M. Shot boundary detection with effective prediction of transitions' positions and spans by use of classifiers and adaptive thresholds. In Proceedings of the 2016 24th Iranian Conference on Electrical Engineering (ICEE), Shiraz, Iran, 10–12 May 2016; pp. 167–172.
33. Xu, J.; Song, L.; Xie, R. Shot boundary detection using convolutional neural networks. In Proceedings of the 2016 Visual Communications and Image Processing (VCIP), Chengdu, China, 27–30 November 2016; pp. 1–4.

34. Hassanien, A.; Elgharib, M.A.; Seleim, A.A.; Hefeeda, M.; Matusik, W. Large-scale, Fast and Accurate Shot Boundary Detection through Spatio-temporal Convolutional Neural Networks. *arXiv* **2017**, arXiv:1705.03281.
35. Soucek, T.; Moravec, J.; Lokoč, J. TransNet: A deep network for fast detection of common shot transitions. *arXiv* **2019**, arXiv:1906.03363.
36. Ming, B.; Lyu, D.; Yu, D. Shot Segmentation Method Based on Image Similarity and Deep Residual Network. In Proceedings of the 2021 IEEE 7th International Conference on Virtual Reality (ICVR), Foshan, China, 20–22 May 2021; pp. 41–45. [CrossRef]
37. Mehmood, M.; Alshammari, N.; Alanazi, S.A.; Basharat, A.; Ahmad, F.; Sajjad, M.; Junaid, K. Improved colorization and classification of intracranial tumor expanse in MRI images via hybrid scheme of Pix2Pix-cGANs and NASNet-large. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 4358–4374. [CrossRef]
38. Bay, H.; Tuytelaars, T.; van Gool, L. SURF: Speeded Up Robust Features. In *Computer Vision–ECCV 2006*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 404–417.
39. Zhou, S.; Wu, X.; Qi, Y.; Luo, S.; Xie, X. Video shot boundary detection based on multi-level features collaboration. J. VLSI Signal Process. Syst. *Signal Image Video Process.* **2020**, *15*, 627–635. [CrossRef]
40. Chakraborty, S.; Thounaojam, D.M.; Sinha, N. A Shot boundary Detection Technique based on Visual Colour Information. *Multimed. Tools Appl.* **2021**, *80*, 4007–4022. [CrossRef]