International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I4P236

Volume 70 Issue 4, 418-432, April 2022

Original Article

An Efficient Approach for Shot Boundary Detection in Presence of Illumination Effects using Fusion of Transforms

Shrikant Chavate[1], Ravi Mishra[2]

[1,2]*Department of Electronics & Telecommunication Engineering, G H Raisoni University, Amravati, Maharashtra, India*

[1]spchavate@gmail.com

Received: 23 February 2022 Revised: 04 April 2022 Accepted: 19 April 2022 Published: 30 April 2022

Abstract *- For video processing applications like indexing, browsing and video retrieval, the video shot boundary detection (SBD) plays a vital role. Video is a popular mode of information sharing and thus, vast database of video is available in cyberspace. The identification of accurate shot boundary is an essential task in video retrieval and indexing. This identification still remains a challenge especially for gradual transitions in video. The proposed approach detects the abrupt and gradual transitions such as fade-in and fade-out with high accuracy. In this paper, the combination of DTCWT-WHT is proposed to extract the features. The preprocessing is applied at an early stage to remove the noise present in the frames. The proposed method implements Deep Belief Network (DBN) for accurate classification of gradual transitions. This method also detects the shots accurately even in presence of illuminations. The experiments are performed on TRECVID datasets of year 2016, 2017, 2018 and 2019. The results of proposed algorithm outperform other SBD techniques with the help of performance metrics such as, precision, recall and F1 score. In addition, under lighting effects, the adoption of early filtering techniques minimizes the number of false alarms.*

Keywords *- Deep Belief Network, SSDOA, Fast Averaging Peer Group, Gradual transition.*

1. Introduction

There has been a substantial rise in technical advancements in the domain of multimedia in recent years. There is a huge evolution of video data over internet and it seems to grow invariantly with each passing day. Video is the most popular medium of information exchange in fields including scientific research, sports, entertainment, and education, making it the most widely used data type on the internet. Owing to the excessive rise in the number of videos and the size of the repository, there is a compelling need to undertake continuous monitoring in order to effectively organize and manage this data [1]. According to the statistics, more than 500 hours of video are uploaded per minute till February 2020. Since there are many videos available, finding the right video clip from such a large database might be difficult. The manual approach of looking for the relevant contents of video and analyzing it, is quite time consuming and complex [2]. This problem motivated to carryout research in finding solution for the video retrieval [3]. The fragmentation of video into its basic elements is an important step of analyzing its structure. This consist of different levels such as scenes, shots and frames. Figure 1 illustrates the basic video structure.

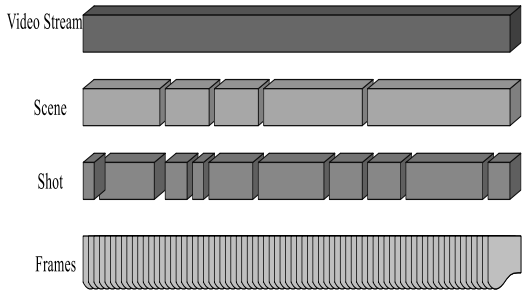

Fig. 1 Basic video structure

The basic components of a video stream are frames, which are connected together to form shots, and scenes are created by integrating many shots. These scenes come together to make a full video [4]. A single shot is defined as a sequence of interconnected successive images, taken by a single camera with the representation of continuous action in time and space [5]. Shot boundary detection (SBD) is a key stage in performing operations such as video retrieval, indexing, and browsing. The shot boundaries are of two types such as, abrupt transition (AT) and gradual transition (GT). Abrupt transition in video exhibits sudden and absolute

changes of frames, whereas in the Gradual transition, changes in frame take place step by step with variable time and velocity as per the requirement of video makers. In addition, the gradual transition exhibits special editing effects such as, fade-in, fade-out, dissolve and wipe. The process of shot boundary detection includes several steps, such as, feature extraction, continuity function construction and the detection and classification of shot boundary.

Figure 2 illustrates the concept of SBD procedure. The detection procedure may include preprocessing of extracted frames. From the state of the art, the color, edge and texture are mostly preferred features, which are then used as input for construction of continuity signal. This gives the measure of similarity /dissimilarity among the frames and with the help of threshold the detection of shot can be done.

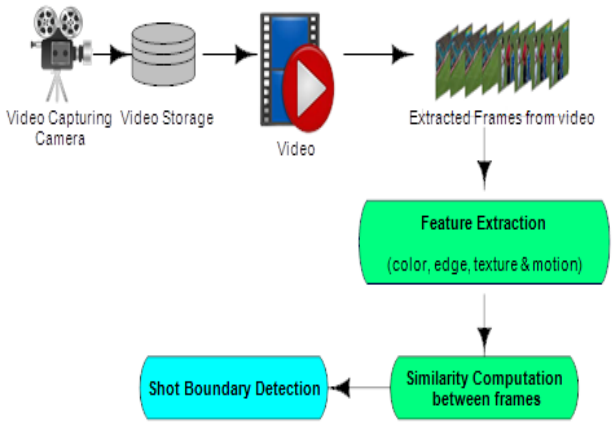

Fig. 2 Conceptual illustration of SBD

Existing literature suggested the use of approaches like pixel differencing, histogram differencing, motion and edge-based techniques for SBD. The pixel difference method is the simplest one to find the measure of dissimilarity. It computes the pixel difference among the consecutive frames and compares it, but this method is found less efficient for detection of gradual transitions. The edge-based approach finds the edges of the objects in the frames. The histogram-based methods aim in reduction of sensitivity towards object/camera motion. The motion-based approach utilizes motion vectors.

However, for the real time applications and the post production procedures, the highly accurate detection of shot boundaries are exceptionally demanded. There are certain areas like bio-medical sectors, where the task like image analysis, image denoising and its reconstruction is much needed. Thus, the preprocessing techniques are highly recommended which also improves the SBD performance. Let's discuss below the method implemented for the preprocessing.

The proposed approach applies Fast Averaging Peer Group (FAPG) filters in preprocessing for enhancing the contrast of the frames and removal of illumination noise. It helps in preserving the edge properties along with smallest information in an image during its restoration. After preprocessing, feature extraction has to be done. Here, the combination of Walsh Hadamard Transform (WHT) and Dual Tree Complex Wavelet Transform (DTCWT) is proposed. This combination is applied on each block of image for feature vector extraction. It combines the valuable properties like shift invariance from DTCWT and benefits from WHT like suboptimal, non-sinusoidal and orthogonal transform. This hybrid method for extracting feature vectors presents the advantages of robustness and less computational cost. As the shot boundaries are of two types: Abrupt and Gradual, the classification of these transitions is made using the Optimized Deep Belief Network (DBN). DBN consists of stacked Restricted Boltzmann Machines (RBMs) and this is used for extraction of features and reconstruction. The proposed methodology also promises to reduce false hits while providing precise detection of abrupt and gradual transitions specifically fade in and fade out.

2. Related Work

E. Hato, M. Abdulmunem [6] have proposed a fast detection method for SBD using Speeded-Up Robust Features (SURF), where the features were extracted from alternate frames of video. In this way, the execution time was reduced. With the help of distance function, the feature matching was conducted, and followed by this, the similarity computation among the feature vectors also taken place.

R. Mishra [5] developed an effective system for SBD by combining DTCWT and WHT. This also implemented the filtering action for the removal of illumination noise at initial stages.

P. Lakshmi, S. Domnic [7] proposed and implemented the SBD technique using Walsh Hadamard Transform (WHT) kernel and WHT matrix. In this, the feature vectors were formed by extracting color, edge, texture and motion features, and the single continuity function had been formed by combining the weighted features. Thus, the detection of cut and gradual transition was held.

L.Wu, S. Zhang, M. Jian et al [8] suggested dual stage approach for SBD, in which abrupt shot had been detected by fusing color histogram and deep features, and the gradual transitions detected using 3D-convolutional neural networks. The complete video was segmented by locating the abrupt cuts, later gradual transitions were detected which were present in the available segments. The different types of gradual transitions then classified using the neural network.
R. Mishra, C. V. Raman et al [9] proposed and implemented the SBD using the DTCWT for real time and non-real time videos.

M.Gygli [10] proposed a technique of CNN which was fully convolutional in time which claimed detection faster than 120x real time.

Chakraborty S. and Thounaojam, D.M. [11] proposed a hybrid technique for Feed Forward Neural Networks (FNN) offers the joint advantages of Gravitational Search Algorithm (GSA) and Particle Swarm Optimization (PSO). They tuned the weights of ANN using PSO, GSA, and PSOGSA, resulting in a hybrid ANN system for SBD. The algorithm was created to identify sections of video with a high likelihood of shot transitions. The next step was to confirm the location of the shot, which lowered the computational complexity and the number of false alarms.

Chakraborty S. and Thounaojam, D.M. [12] proposed an approach of SBD based on gradient and color information. Here, adaptive threshold was used to derive the probable transitions from variations in illumination and contrast structure. To detect transitions, this system used a two-stage verification process.

B. Rashmi, H. Nagendraswamy [13] developed a system of SBD, in which, the edge gradient fuzzified frame was built using the correlation of global and local features, and the block based Mean Cumulative Sum Histogram (MCSH) was retrieved from each frame. To detect the cut and gradual transitions, the relative standard deviation (RSD) statistical metric was used to the acquired MCSH, but false hits were occurred during the procedure.

R. Liang, Q. Zhu., et al [14] developed a system in which the extraction of CNN features for all the frames were carried out and further they computed the cosine similarity for a pair of frames. Then the abrupt transitions detected based on local frames and gradual transitions on the basis of window similarity with dual threshold.

Z. N. Idan, S. H. Abdulhussain et al [15] developed a technique of SBD based on active area of frames and candidate segment selection techniques. Then the active frame area was under consideration for finding the shot transitions, thus this helped in reducing the computation time. The use of adaptive threshold and inequality criteria was made to remove the non-transition frames. The machine leaning statistics based SVM was implemented to detect abrupt transitions.

H. M. Nandini, H. K. Chethan et al [16] proposed a system which could locate the abrupt transition by extracting binarized edge information from frames for texture characterization using local binary pattern (LBP). Followed by this, the histogram construction and threshold computation were used to detect the shot boundary. Later the task of keyframe extraction was done.

S. Zhou, X. Wu et al [17] designed a system of SBD which focused on fast detection procedure. Here, the candidate segment selection was made using color histogram and SURF along with uneven slice matching, which helped to carry out abrupt shot boundary detection. The gradual transition detected using motion area extraction, SIFT and even slice matching.

R. Shen, Y.Lin, T. Juang et al [18] used the HLFPN model with histogram difference to execute the predetection, the hybrid combination of HLFPN model and keypoint matching was used here. SURF method used here which helped to detect the shot boundary under instances of changes in illumination. Also, it reduced the computational cost because here all frames were not considered for matching.

A. Sulaiman, S. Mahmood, et al [19] proposed the approach of dynamic time warping. In this method, solution is provided towards shot detection under instances like movement of camera/objects.

A. Singh, D. Meitei et al [20] proposed and implemented the dual stage system for the abrupt transition which could bear the illuminations and object/motion effects. In the first stage, adaptive Wiener filter had been applied to the lighting component of frames and LBP-HF was extracted for removing the illumination effects. The second stage made the use of canny edge detection which further removes the illumination and motion effects.

3. Major Challenges That Demand Attention

The SBD process performs abrupt and gradual transitions detection. During this process, there are the chances of false detection due to the factors like illuminations or lighting effects. Apart from this, the gradual transition lasts over several frames, and these includes the special editing effects. Therefore, to locate the accurate gradual transition is also a challenge. The motion of large objects / camera may also detect the shot boundary wrongly.
To overcome the above mentioned difficuties, it is necessary to develop an efficient technique of shot boundary detection.

4. Proposed Methodology

This paper implements a hybrid approach of DTCWT and WHT. It provides special attention towards the elimination of noise in each of the frames at preprocessing stage using fitering techniques. The difficulties of occurrence of illuminations and contrast issues can be overcomed using the Fast Averaging Peer Group (FAPG) filters [21]. The FAPG filters improves the contrast and removes the illumination noise. After preprocessing, the HSV color histogram distance is calculated for each adjacent frame. For extracting the feature vectors, the combination of DTCWT and WHT is to be applied on each frame. The merged feature vector from each of the frame will be acted as an input for further matching procedure. Apart from this, the deep belief network (DBN) is

used for finding the false hits where the SSDOA has to be used for weight optimization for reducing the learning error. Thus, the shot boundaries will be located. Figure 3 shows the brief procedure of proposed system of SBD.

4.1 FAPG filtering action

Color images may contain impulsive noise which corrupt the data. This happens due to the inclusion of transmission errors in the noisy channels, poor lighting effects, aging of storage resources and faulty pixels in camera sensors [21]. The filtering of noise at early stages leads to more accurate results and reduced computation time for the detection of shot trasitions. In the proposed methodology, FAPG filters are used for the removal of illumination noise during the pre-processing stage,and this helps in enhancing the contrast of the frames. The restoration of images can also be done with FAPG filters, by preserving the edge and small image properties. The FAPG filtering action is based on the premise that the centre pixel's membership in the local neighbourhood is determined by the size of its peer group [21]. The important functions of this filtering method consist of pixel inspection and pixel replacement [5]. In the pixel inspection part, the degree of membership is to be computed for the central pixel and the local neighborhood window. In the replacement part, the replacement of pixels using the Weighted Average Filters (WAF) takes place for the outlier pixels.

The figure 4 shows the circle in which 'p1' is considered as central pixel and 'd' as a threshold (shown by dotted lines). To find out the close neighbors, the distance between central pixel and the other pixels (i.e. p2 to p7) have to be calculated. If any individual distance exceeds 'd', then it considered as outlier. And if the distance of pixel lies within the range as $0 \leq d \leq 1$, then it is considered as the close neighbors (CN). The number of close neighbors will give the size of the peer group. In figure 4, the peer size is 4. Also, when d=0, it shows the pixels are identical whereas, d=1 gives the Euclidean distance in the color space.

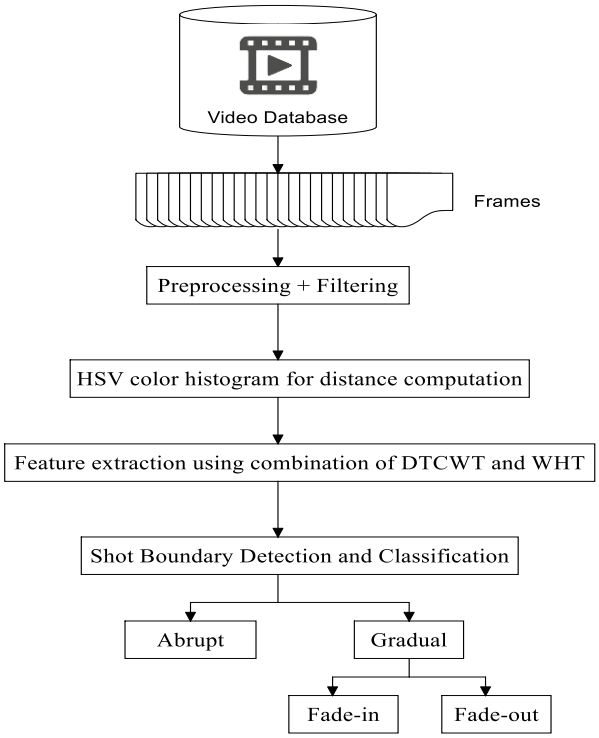

Fig. 3 Proposed methodology

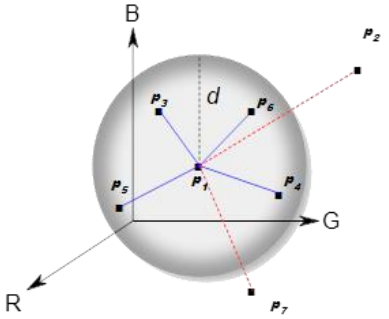

Fig. 4 Determination of size of peer group and close neighbors

The peer group size provides the measure of non corrupted pixel and corrupted pixel due to noise. When the peer group size counts low, the pixel will be treated as corrupted, otherwise not. In addition, the 'd' parameter is to be selected in proper manner, since its high value may lead to noisy pixels to add and then removal of this noise cannot be done even with this algorithm. Once the peer group size is computed, the pixel replacement task can be carried out using following ways,

Considering central pixel p_1, for the peer group size a_1, if $a_1 \leq 1$, then the pixel will be trated as outlier and has to be replaced with output of WAF which applied to the pixels that are part of the same operating window [21]. The weights w_i varies from, i=2,….,n of the corresponding pixels p_i.

$$w_i = \frac{\sigma i}{\sum_{i=2}^{n} \sigma i}, \quad \text{where } \sigma_i = a_i^{\beta} \qquad (1)$$

Where, 'n' is the window size and β is the secondary parameter which influence the result quality. The WAF output OP_1, on replacing p_1 is as follows,

$$OP_1 = \frac{1}{\sum_{i=2}^{n} wi} \sum_{i=2}^{n} w_i . pi \qquad (2)$$

The more number of CNs will offer bigger relative influence on the output of the filter. For the pixels that do not have any CN will not be considered for average. The parameter β if lies between 0 & 1 (both exclusive), then difference in peer group sizes of neighboring pixels are decreases and when β exceeds 1 (i.e. β>1) then the difference will increases. For the condition, when central pixel having two or more CNs then no changes are required and its degree of membership is enough to treat it as uncorrupted[21] . Also, in some rare instances, it may happen that within W there will be no CN found for all the available pixels, then the window size has to be increased so that atleast two uncorrupted pixels are found. After carrying out this preprocessing, the next step is a computation of HSV histograms. The figure 5, shows the output of FAPG fitering stage in which the image is cleaned of noise. In this, the image quality improves and it will be ready for the further procedure.

4.2 Color distance calculation with HSV histograms

The pixel differencing and the histogram difference are the most widely used methods[22], these are most beneficial for finding out the degree of similarity/dissimilarity among the frames. The HSV histogram offers the valuable advantage like it is easier to understand than RGB colour space [23][24]. It uses three dimensions to describe the colors.

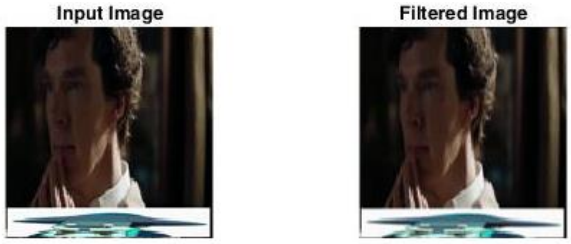

Fig. 5 Output of FAPG filter stage

The *hue* signifies the basic properties as, red, green and blue etc. *Saturation* define the fitness, higher saturation signifies pure of color and lower saturation gives greyer of color. The *value* corrosponds to brightness.
The histogram difference can be computed as,

$$Hist_{diff}[j] = \sqrt{\sum_{i=1}^{N} |m_{j,i} - m_{j-1,i}|^2} \qquad (3)$$

The color histogram for frame j with N dimension is denoted as m_j.
In figure 6, the abrupt detection using the histogram method is clearly shown on the plot generated in the form of sharp peak. This detection is an easy task since the considerable difference between the adjacent frames directly indicate the location of abrupt transition. The gradual type of transitions found difficult to locate. This is because of special editing effects lies over the frames with minor differences among them. In addition, the inclusion of flashes, illumination and lighting effects will make the detection of gradual transitions more difficult and it prone to false hits.

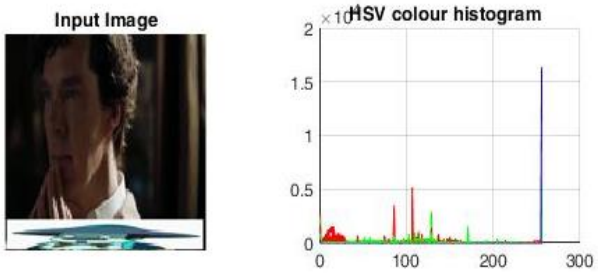

Fig. 6 Histogram estimation of image

4.3 Feature extraction based on fusing WHT-DTCWT

The feature extraction acts as a major component for producing the accurate results for any SBD algorithm. The proposed system combines DTCWT and WHT. This provides the efficient way as it combines the vital advantages of both the types of transforms. Two real DWTs are used in the DTCWT. The first one takes care of real part of transform and the second for imaginary part. The directional selectivity and shift invariance are the major advantages of DTCWT over DWT. The DTCWT also consisting of distinct sub-bands for positive and negative orientation. Figure 7 shows the analysis of the filter banks for the DTCWT and figure 8 shows the synthesis of the filter banks for the DTCWT. $h_o(n)$ and $h_1(n)$ are considered as LPF and HPF pair for upper filter bank (FB). Whereas, $g_o(n)$ and $g_1(n)$ are the LPF and HPF pair for the lower filter bank (FB)[25]. Along with DTCWT, the WHT added its advantages to the feature extraction method. The WHT is found to be attractive in implementation due to its simplicity and other important properties[5][7]. The WHT is suboptimal, nonsinusoidal, simple, fast in transformation and orthogonal.

The WHT can be constructed from WHT matrix. It is exist for N>2, since the size of the matrix is generally a power of 2. For the order 4, i.e. N=4, the WHTM is,

$$D = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \qquad (4)$$

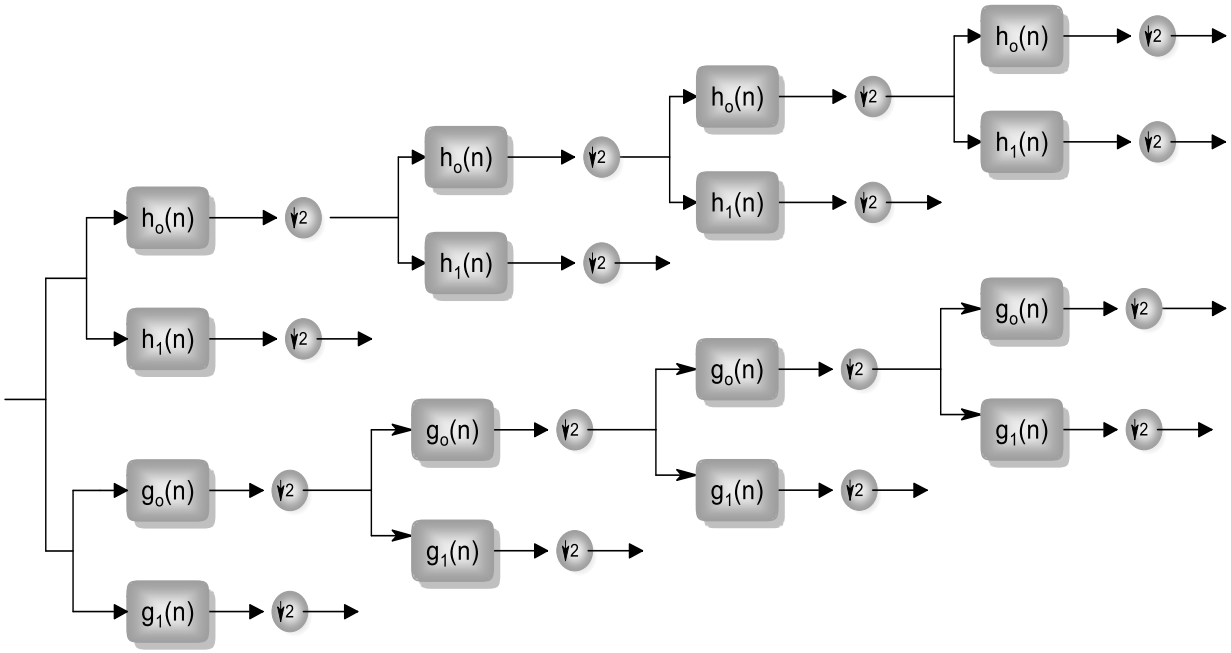

Fig. 7 Analysis of the filter banks for the DTCWT[25]

Figure 9 shows the WHT kernels for all basis vectors of WHTM of order 4, it is clear that, the kernels can be represented as basis vectors in vector space as $V = \{v_1, v_2, \dots, v_{16}\}$. In this, the average color component is to be used which are projected on basis vectors to extract the features like, color, edge and texture. The various image frequecies have also taken into consideration, where the high frequency correpsonds to the edge details and the lower frequencies corresponds to brightness of image.

Let, $F_m = \{f_1^m, f_2^m, f_3^m, f_4^m\}$, where 'm' indicates the number of blocks. F_m is the projection values of the block obtained as the inner product of m^{th} block (B_m) and V_j.

Also, the j= 1,2,3,4 respectively. The color, edge and texture features have to be computed for each frame in a block. Begining with color feature extraction, it has be carried out with the help of the following equation,

$$X = f_1^m\{B_m, v_1\} \qquad (5)$$

The next is, from each block of frame the egde features are to be extracted. This is less prone to lighting effects as well as camera movements and other operations. The magnitude of gradient vector will define the edge strength by using the equation,

$$Y = \sqrt{(f_2^m)^2 + (f_3^m)^2} \qquad (6)$$

The above equation can be modified as follows to simplify the computation,

$$Y \approx (f_2^m)^2 + (f_3^m)^2 \qquad (7)$$

The next feature extraction is of texture features, in which the vector is used for representing the finite sum. The equation 8) shown below helps to perform the block projection on vector space as,

$$F = f_1^m v_1 + f_2^m v_2 + f_3^m v_3 \qquad (8)$$

$$F = \sum_{i=1}^{3} (B_{m,} v_j) v_j \qquad (9)$$

The texture feature representation can be achieved as,

$$Z = |B_m^2 - F^2| \qquad (10)$$

Figure 10 shows the DWT walsh transform of the input image, as discussed earlier from figure 5, the input image has already been undergone through the fitering stage.

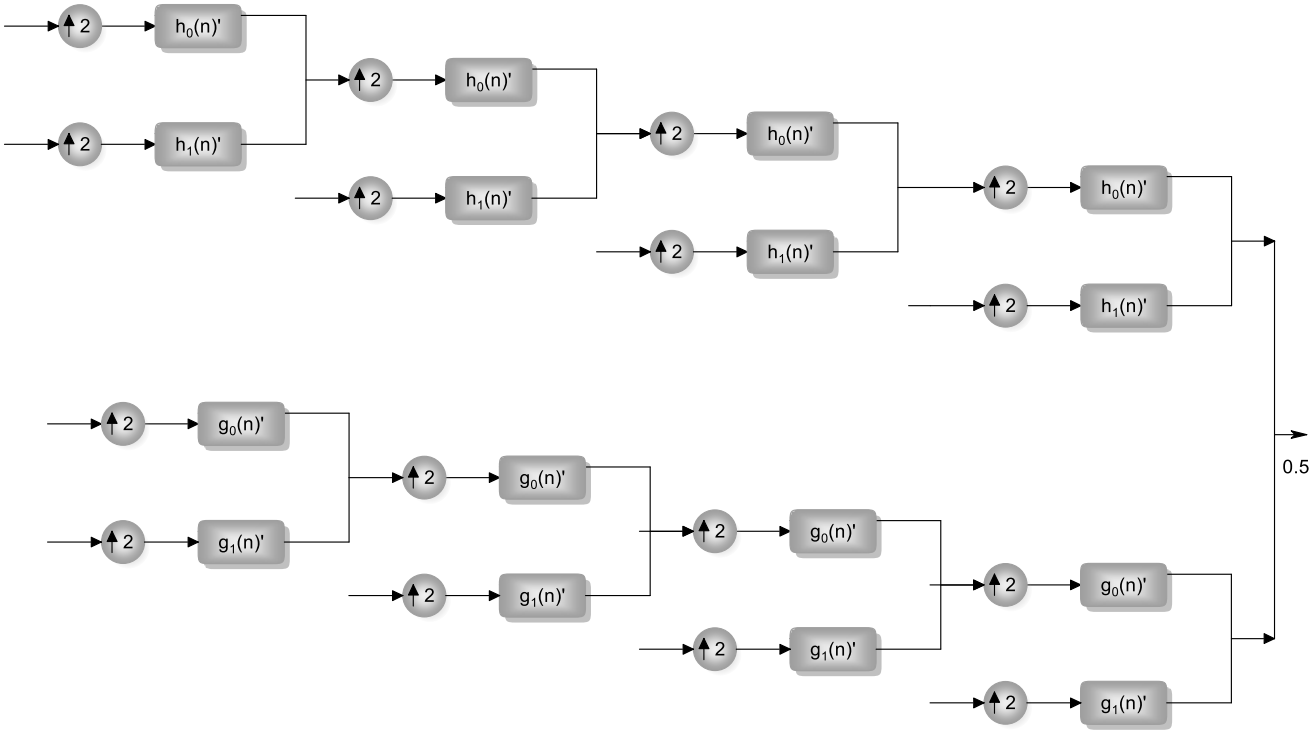

Fig. 8 Synthesis of the filter banks for the DTCWT[25]

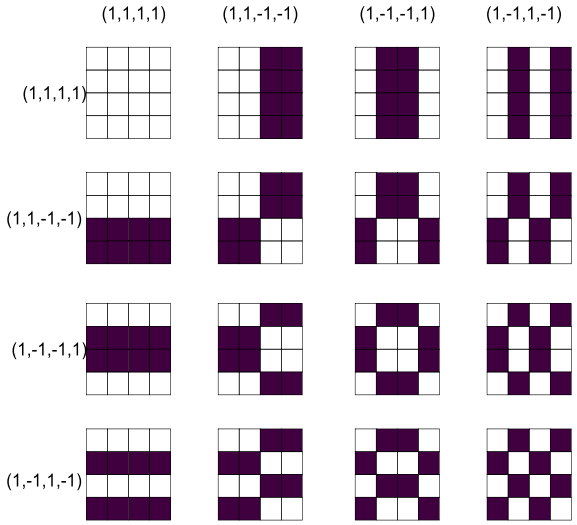

Fig. 9 WHT kernel [7]

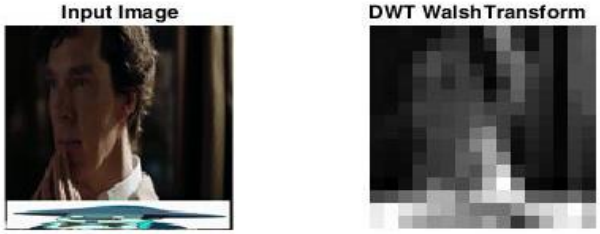

Fig. 10 DWT Walsh transform

4.4 Designing a Continuity Signal

In the previous sections, different features have been extracted which can be contributed as inputs for the determination of similarity/dissimilarity among the frames of video. The extracted feature and the distance metric will combine together to get the measure of continuity signal. It is the vital step of the proposed algorithm where the important decision for locating a boundary has to be taken place. The city block distance measure uses the features viz, color, edge and texture for the construction of continuity signal function.

$$\mu(k) = Dm(h, h+1)X = \sum_{m=1}^{n} |X_{m,h} - X_{m,h+1}| \quad (11)$$

$$\sigma(k) = Dm(h, h+1)Y = \sum_{m=1}^{n} |Y_{m,h} - Y_{m,h+1}| \quad (12)$$

$$\delta(k) = Dm(h, h+1)Z = \sum_{m=1}^{n} |Z_{m,h} - Z_{m,h+1}| \quad (13)$$

$$\gamma(k) = 10 * log_{10}\left[\frac{v^2}{\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}(O_{ij} - CmF_{ij})^2}\right] \quad (14)$$

Here, three important features that are, color, edge and texture are extracted and their values are mentioned above in equations 11), 12) and 13). The motion strength to be computed is depends upon temporal domain structure and can be calculated from consecutive frames. The measure of similarity/dissimilarity has to be computed with the help of equation 13) mentioned above. The continuity values thus obtained has to be normalized between the range of 0 and 1,

and these values further acts as an input for the detection of shot boundary. The fusion of individual continuity signals (viz,μ,σ,δ,γ) is made to get the single continuity function. However, because the characteristics may contribute various amounts to the visual representation of frames, this fusion process will not occur immediately. After this, the features will be assigned with the weights (ω1, ω2, ω3, ω4) as mentioned in the following equation,

$$\Omega = \omega_1\mu + \omega_2\sigma + \omega_3\delta + \omega_4\gamma \qquad (15)$$

Further procedure of classification is performed using Deep Belief Network (DBN).

4.5 Classification of frames using DBN

A Deep Belief Network (DBN) is a generative graphical model. It is the combination of statistics and probability with neural netwok and machine learning. The DBN structure consists of various layers, which contains the different values. Looking into history of neural networks, for the first generation, it used the perceptrons. But the perceptrons worked efficiently at basic levels only and not for advanced technology. Moving forward , in second generation, concept of back propagation was used. The back propagation helps in getting the error value on comparing the output received and the required output. Next, the evolution of DBN solved the challenges related to inference and learning issues. The DBN is composed of RBMs, in which the hidden layer of each sub-network is the visible layer for the next one. The hidden layers formed are independent with due conditions. The training of layer properties is done for obtaining the input signals directly from the pixels of an image. Every time, the addition of another layer of features is made to the deep belief network. To classify the frames, the retrieved feature vectors and the continuity function value are fed into the DBN model. The neural networks are the preferred one for the said purpose, because it offers the advantages in learning new and synthetic features. The RBM enhances the training efficiency of a model and the high-level features can be effectively extracted from the training data [26]. The back propagation also used for the cases when the overfitting issues are occurred. The energy function ENR(r,s) is used to determine the joint distribution of visible and hidden layers as,

$$P(r,s) = \frac{e^{-(ENR(r,s))}}{\sum_{r,s} e^{-(ENR(r,s))}} \qquad (16)$$

The formula for energy function is as follows,

$$ENR(r,s) = -\sum_{i=1} u_i r_i - \sum_{j=1} v_j s_j - \sum_{i,j} r_i s_j w_{ij} \qquad (17)$$

Where, w_{ij} = weights between hidden and visible layers
U_i = coefficient of visible layer
V_j = coefficient of hidden layer

Now, dealing with the output received from DBN, and deciding whether it is useful for further process or not, the difference of the received output and the desired one is to be compared. In the same view, cost function is to be formed by using Mean Square Error (MSE).
The cost function 'CF' is defined as,

$$CF = \frac{1}{M} \sum_{j=1}^{N} \sum_{i=1}^{N} (AO_j(i) - DO_j(i))^2 \qquad (18)$$

Where, N= Output layers
M= Data layers
$AO_j(i)$= actual o/p received on j^{th} unit in time (i)
$DO_j(i)$= desired o/p received on j^{th} unit in time (i)

The same procedure has to be repeated till the stopping criteria is reached. There must be an issue in weight updation and occurrence of error during the run time. This problem is taken into account and the solution is proposed by using SSDOA. The SSDOA helps in optimizing the weights and also reduces the learning error.

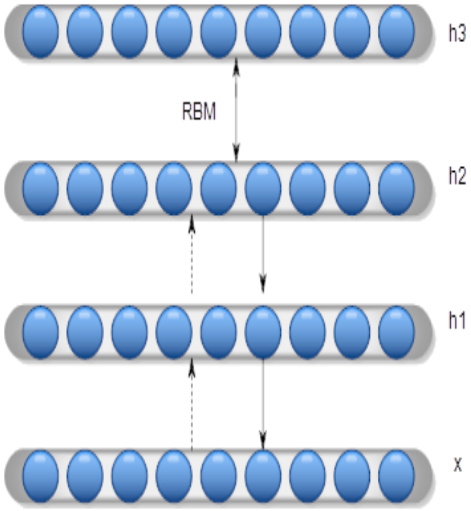

Fig. 11 Structure of RBM

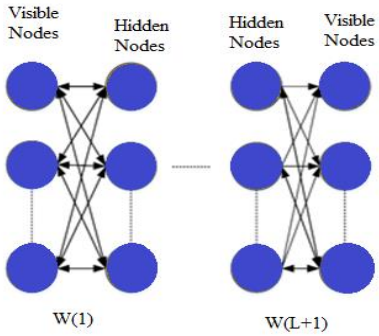

Fig. 12 Structure of Deep Belief Network

4.6 Weight updating using Social Ski Driver Optimization Algorithm (SSDOA)

The updation of the weight values received from DBN can be done through SSDOA. The Social Ski Driver (SSD) consist of many parameters like, a) Positions of the agents b) Previous best positions c) Mean Global Solution d) Velocity of the agents [27]. To calculate the objective functions, finding the positions of the agents is required. The fitness function is used to find out the fitness of all the agents, and the fitness of each agent needs is to be computed for the current position. Thus, on comparison the best position can be stored. The mean is to be computed for the best three solutions. The updation in the positions can be taken place by adding the velocity.
Considering Pi at time(t), and after an instant say at time (t+1), the value of position can be found as updated by adding a velocity V(t) as follows,

$$P_i^{t+1} = P_i^t + V_i^t \tag{19}$$

Where the velocity components can be elaborated as,

$$V_i^{t+1} = \begin{cases} c\sin(r_1)\,(B_i^t - P_i^t) + \sin(r_1)\,(M_i^t - P_i^t) & if\ r_2 \le 0.5 \\ c\cos(r_1)\,(B_i^t - P_i^t) + \cos(r_1)\,(M_i^t - P_i^t) & if\ r_2 > 0.5 \end{cases} \tag{20}$$

Where, B_i = previous best position
P_i=position of agent
M_i=mean global solution of complete population
r_1, r_2 = uniformly random generated numbers in the range of [0,1].
c= balance parameter for exploitation and exploration

The SSD searches the optimal or near-optimal solutions. The initialization of position of agents are made randomly and the number of agents is determined by the users. The update in the agent's position can be achieved with addition of velocity components to the old positions. The distance between the present position and the previous best position determines the modified velocity of the agents [27], and the distance between the current position and the mean global solution Mi.

5. Experimental Result and Analysis

This section consists of detail result analysis of the implemented algorithm. This algorithm is implemented on various videos of the TRECVID datasets from year 2016 to 2019. The details of the datasets used is mentioned below.

5.1 Description of TRECVID dataset

For a proper comparison and validation of various methodologies, the common and authenticate dataset is required. There are several government and private organizations are releasing dataset for this analysis. The majority of the approaches have been applied and their results have been validated by comparing them to the best performers in the TRECVID datasets. National Institute of Standard and Technology (NIST) provides the datasets from year 2001 to 2020. We have implemented the proposed system on different videos from TRECVID from year 2016 to 2019. The dataset is made available from http://trecvid.nist.gov/. Through open and metric-based assessment, TRECVID aims to improve the retrieval of desired content from digital video. The videos included in these datasets are in MPEG format and they are manually segmented by detecting the shot boundaries[5].
TRECVID 2016 dataset consist of 4593 video files of IACC.3 collection. The collection.xml file contains various urls, through which the videos can be collected which are of different categories.

The IACC.3 dataset contains 4593 video files of approximate duration of 600 hrs and size of nearly 144GB. The videos are of MPEG-4/H.264 format and the duration are varying between 6.5 mins to 9.5 mins. The average duration of this set of videos counts to 7.8 mins.

IACC.2.A-C contains three datasets of approximately 7300 video files of duration 600 hrs and of size 144 GB. The format is MPEG-4/H.264 and duration varying from 10s to 6.4 mins. The average duration of this set of videos counts to 5 mins.

IACC.1.A-C contains three datasets of approximately 8000 video files of duration 600 hrs and of size 160 GB. The videos are of MPEG-4/H.264 format with duration between 10s and 3.5 min.

IACC.1. tv10.training contains approximately 3200 internet achieved videos of duration 200 hrs and of size 50 GB. The format is MPEG-4/H.264 with durations ranging 3.6 mins and 4 mins. In 2017, there are the revisions made in the 5 tasks of 2016 datasets, they are AVS, INS, MED, SED, LNK.
In 2018, the TRECVID 2018 NIST works with revised task like AVS, INS, VTT.

In 2019, the videos had been segmented to form the short video shots, which are counted to nearly 1 million numbers. This dataset contains 7475 Vimeo videos with average duration of 8 hours.

5.2 Result Analysis

The proposed algorithm is implemented in MATLAB 2020a. The experimental results of the proposed system are compared with recent techniques of SBD. The findings are examined using TRECVID datasets, and the suggested system's efficiency is assessed using performance measures. The proposed system detected the Abrupt and Gradual transitions efficiently even in presence of illuminations.
The performance metrics like precision, recall rate and F1 score are used here, these three factors will determine an algorithm's performance and dependability.

$$Recall = \frac{T}{(T+M)} \tag{21}$$

$$Precision = \frac{T}{T+F} \quad (22)$$

$$F1\ Score = \frac{2*Precision*Recall}{(Precision+Recall)} \quad (23)$$

Where, T = Correct transitions detected
M = Missed transitions
F = False transitions detected
Precision refers to the relevancy of total identified frames, recall to correctly recognized relevant shots, and F1 score to the weighted average of both P and R. [31].

5.3 Abrupt Transition detection

The performance of proposed system of DBN-SSDOA for SBD is compared with different techniques that are mentioned in below tables. The efficiency of the present work is evaluated using Precision (P), Recall rate and F1 Score.

Table 1. Precision for the abrupt transition detection on different TRECVID dataset

TRECVID Dataset	Proposed Algorithm (%)	DBN (%)	RNN (%)	DNN (%)	CNN (%)
2016	94.5	93	91.26	90	88
2017	92.56	90.23	88.25	87.25	86
2018	92	90	89	88	87
2019	91.25	89.01	88.23	87.12	86.13

Table 2. Recall rate for the abrupt transition detection on different TRECVID dataset

TRECVID Dataset	Proposed Algorithm (%)	DBN (%)	RNN (%)	DNN (%)	CNN (%)
2016	93.5	90.25	89.23	88	87.23
2017	92.36	89	87.25	86.25	84.12
2018	93	88.5	87	86	85
2019	92	88	86.5	85	84

Table 3. F1 Score for the abrupt transition detection on different TRECVID dataset

TRECVID Dataset	Proposed Algorithm (%)	DBN (%)	RNN (%)	DNN (%)	CNN (%)
2016	92.5	90.12	88.23	87.15	87
2017	91.12	89.01	88.01	87.25	86.23
2018	90.23	89	87.25	86.12	86
2019	89.89	88	87	86	85

Table I, II and III shows the result analysis for the abrupt transition detection with performance metrics such as Precision, Recall and F1 score. In this analysis, the precision value obtained by proposed technique for TRECVID 2016, 2017, 2018 and 2019 datasets are 94.5%, 92.56%, 92% and 91.25% respectively.
Table II shows Recall rate obtained for TRECVID 2016, 2017, 2018 and 2019 datasets are 93.5%, 92.36%, 93% and 92% respectively.
Table III shows the F1 Score obtained for TRECVID 2016, 2017, 2018 and 2019 datasets are 92.5%, 91.12%, 90.23% and 89.89% respectively.
The performance metrics values shown above indicates the efficiency of the proposed approach in detecting the abrupt transition from the set of videos from TRECVID datasets.

5.4 Gradual Transition Detection

Figure 14 shows the result obtained for Fade-in transition detection with respect to Precision. The proposed algorithm of DBN-SSDOA produces the high value of precision as compared to DBN, RNN, DNN and CNN. The precision values obtained for the proposed system for fade-in detection are 88.79%, 87.33%, 86.48% and 85.79% for TRECVID 2016, 2017, 2018 and 2019 datasets respectively.

The figure 15 shows recall rates for fade-in detection which are 89.59%, 88.26%, 87.90% and 87.13% for TRECVID 2016, 2017, 2018 and 2019 datasets respectively.

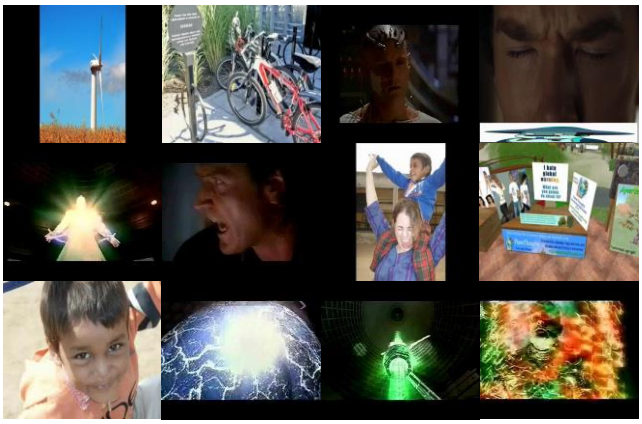

Fig. 13 Abrupt Shot transition detected in presence of illuminations

Moreover figure 16 gives the F1 score for the experiments done and they are as, 88.13%, 87.15%, 86.46% and 85.46% for TRECVID 2016, 2017, 2018 and 2019 datasets respectively.

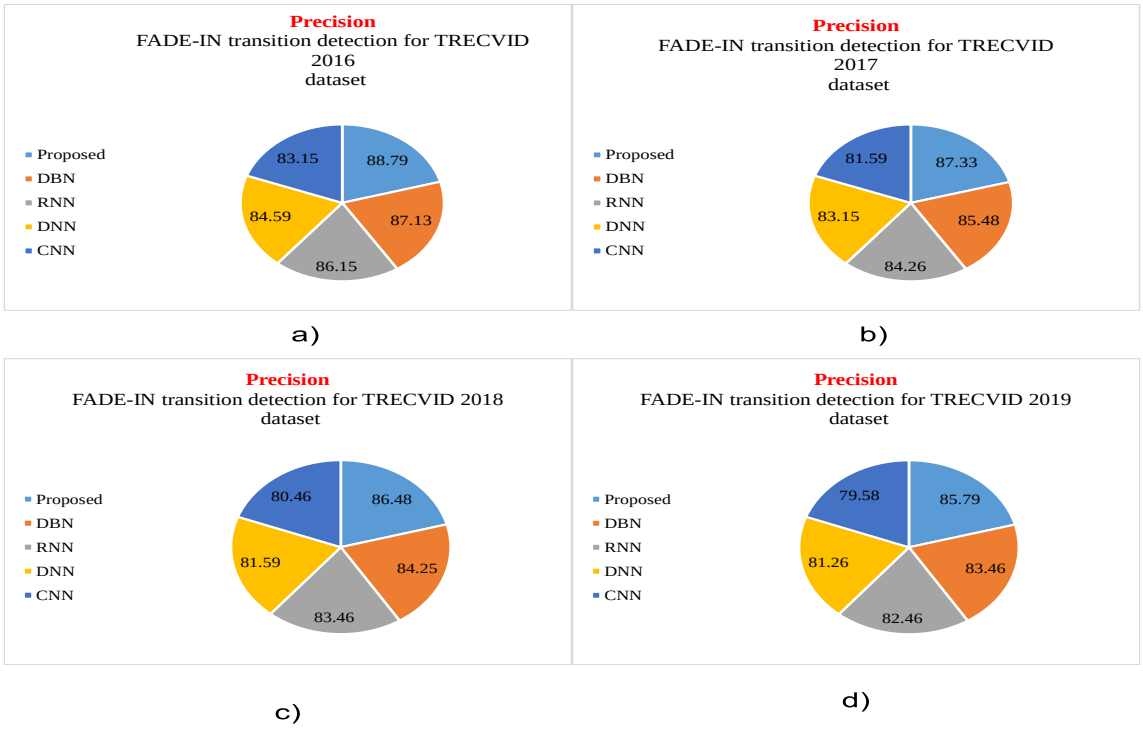

Fig. 14 a), b), c) and d) shows the precision values for Fade-in detection on different TRECVID datasets

Similarly, Figure 17, 18 and 19 shows values of the Precision, recall rate and F1 score for the Fade-out detection which taken place on the same TRECVID datasets used for Fade-in detection purpose.

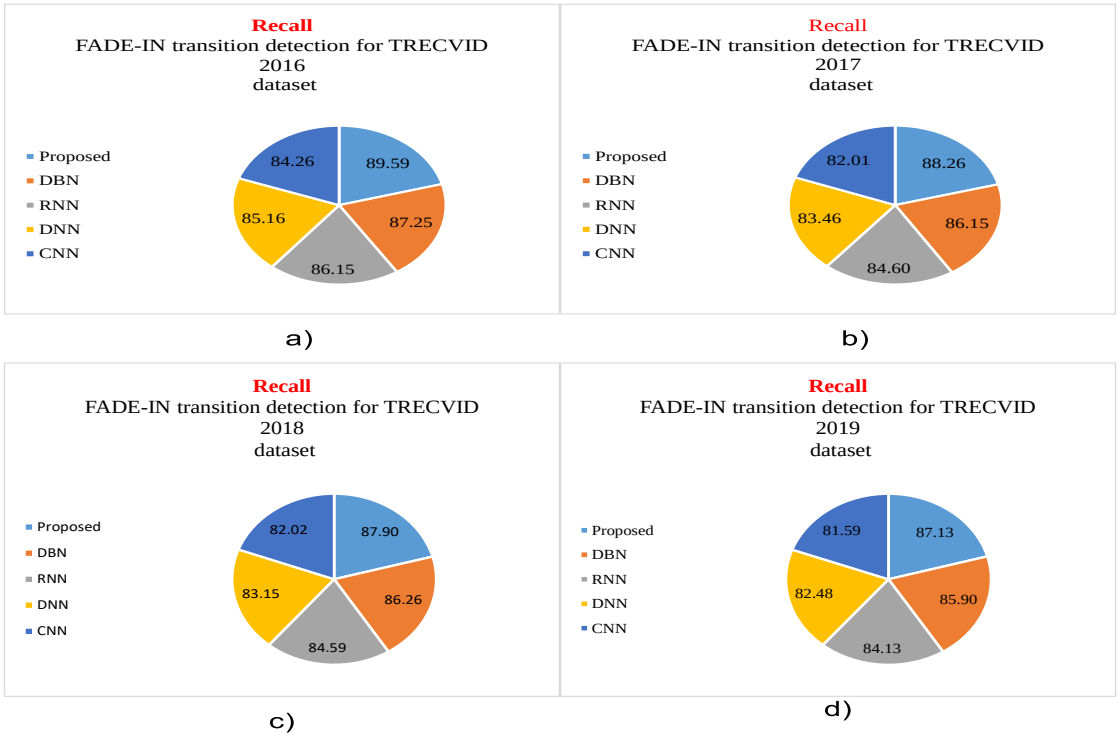

Fig. 15 a), b), c) and d) shows the Recall values for Fade-in detection for different TRECVID datasets

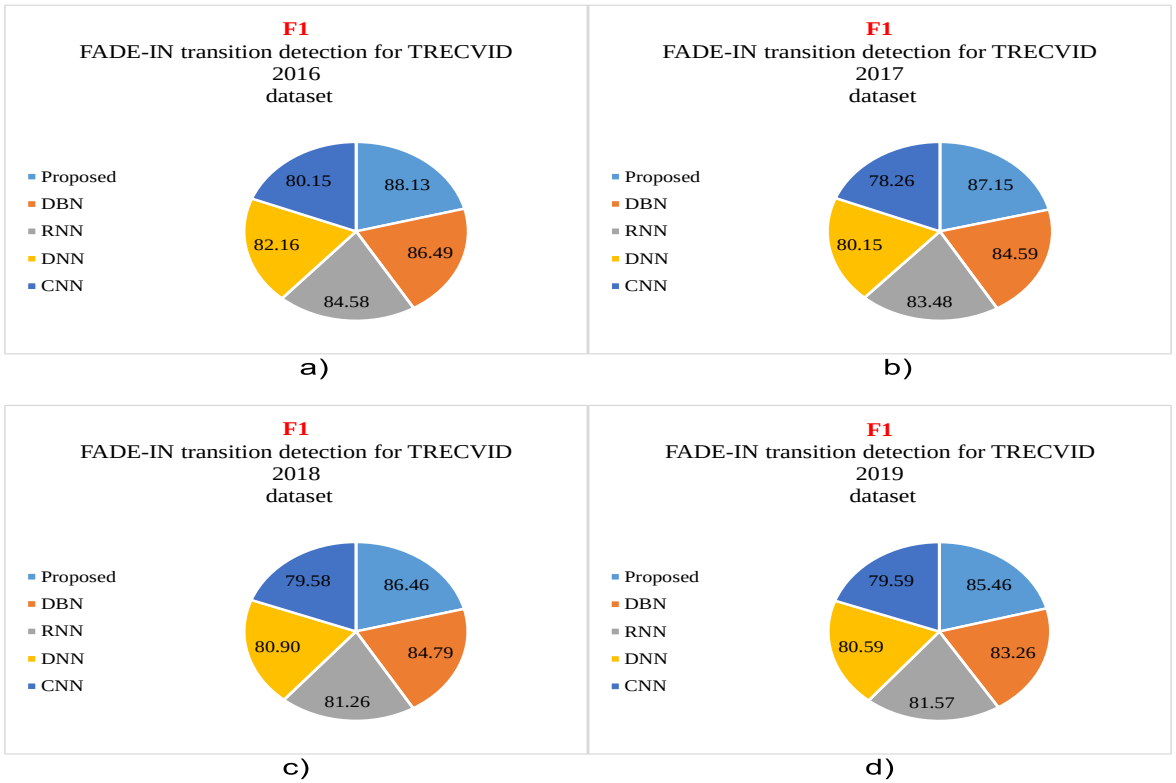

Fig. 16 a), b), c) and d) shows the F1 score for Fade-in detection for different TRECVID datasets

Fig. 17 a), b), c) and d) shows the Precision values for Fade-out detection for different TRECVID datasets

Table IV displays the comparison of proposed system with different recent approaches in terms of performance metrics, here the proposed system found to be outperformed in the SBD process.

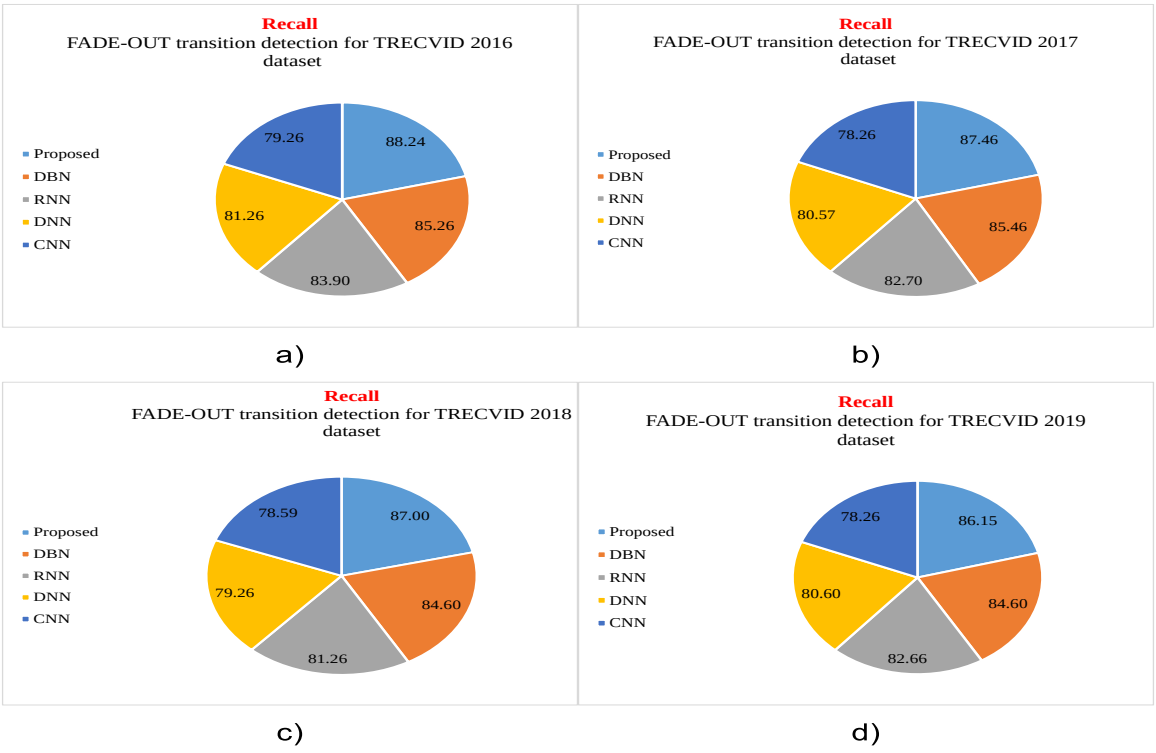

Fig. 18 a), b), c) and d) Recall values for Fade-out detection for different TRECVID datasets

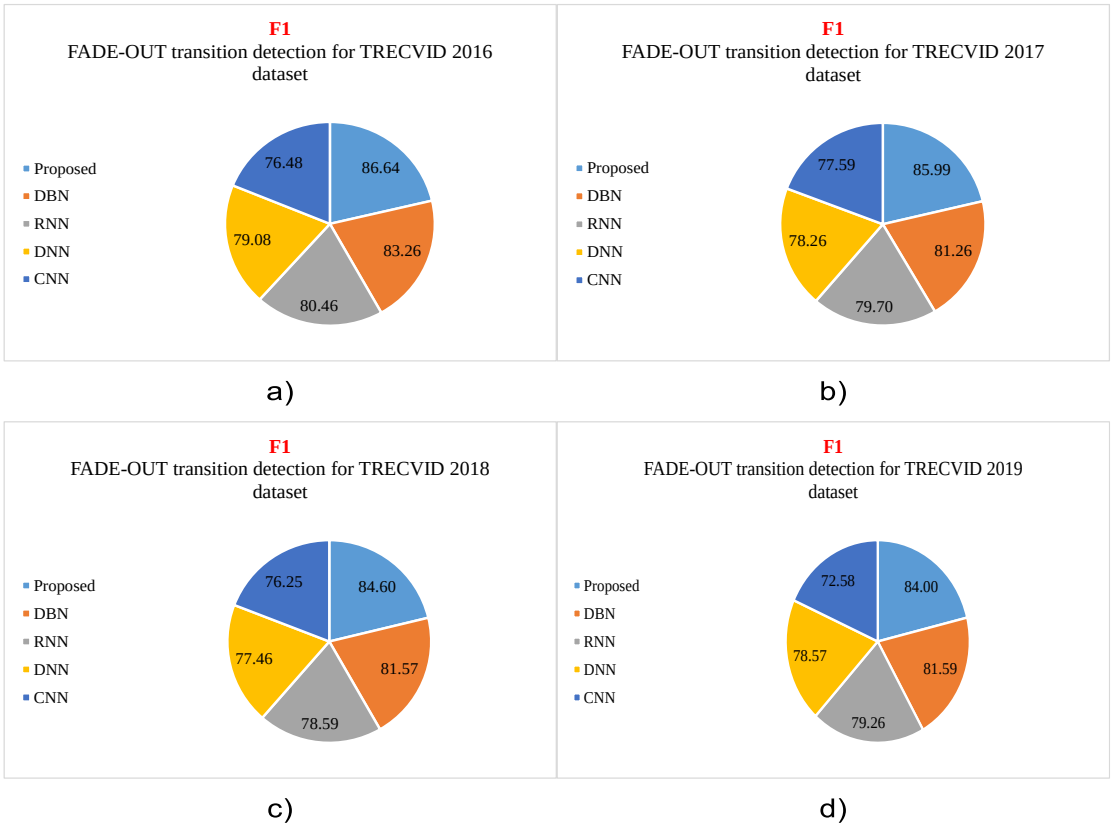

Fig. 19 a), b), c) and d) shows the F1 score for Fade-out detection for different TRECVID datasets

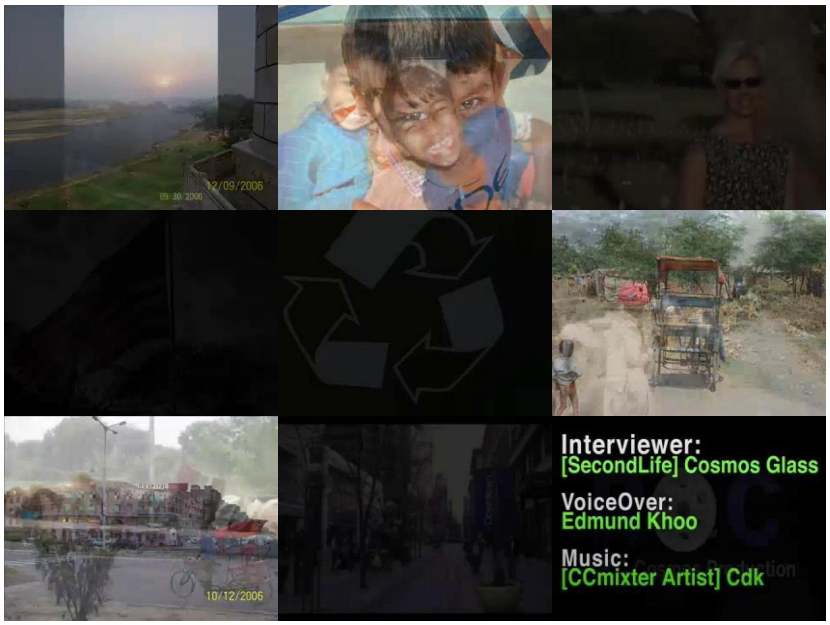

Fig. 20 Fade-in and Fade-out detection

Table 4. Comparison of performance metrics for detection of Abrupt and Gradual Transitions

Methods	Abrupt Transition			Gradual Transition		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
Proposed	93	96	94.5	90.5	91	91
WHT[7]	89	91	90	87	85	86
Perceptual Scheme[28]	72.1	88	79.2	72.3	63.3	67.5
Fuzzy Color Distribution Chart[29]	82.2	86.1	84.1	62.4	76.5	68.7
Multi-modal Visual Features[2]	91.1	94.2	92.8	73	75.1	74
SVD[30]	89.2	95.9	92.8	68.9	78.9	73.6

6. Conclusion

The proposed method is an integration of WHT and DTCWT for extraction of the features from each block of frame. In the proprocessing stage, the FAPG filtering action reduces the illumination noise intensity to much lower extent and its properties like pixel inspection and replacement helps to get restoration of desired frames. The optimized DBN classifies the gradual transtions with remarkable accuracy. The method proposed in this paper performed the experiments on latest TRECVID datasets of year 2016,2017,2018 and 2019. It demonstrates the performance comparision of proposed system with recent techniques like DNN, RNN, CNN and DBN using same datasets. In addition, a comparision of the approaches like perceptual scheme, multi-modal visual features, walsh hadamard transform and singular value decomposition with the proposed system is presented. The results obtained by proposed system claims better efficiency than the other techniques mentioned in literature. The proposed system achieved the notable values of performance metrics like precision, recall and F1 measure. The fade-in and fade-out transtions are much accurately detected with large reduction of false alarms.

Acknowledgement

The authors would like to express their gratitude to the National Institute of Standards and Technology (NIST) for sharing the TRECVID dataset, which would not have been possible without it.

References

[1] Sasithradevi A, Mohamed Mansoor Roomi S, A New Pyramidal Opponent Color-Shape Model Based Video Shot Boundary Detection, J Vis Commun Image Represent. 67 (2020) 102754. https://doi.org/10.1016/j.jvcir.2020.10275

[2] Khan MM, Chamnongthai K, Member S, Multi-modal Visual Features Based Video Shot Boundary Detection. 3536 (2017) 1–13. https://doi.org/10.1109/ACCESS.2017.2717998

[3] 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. 5 (2016) 89–104. https://doi.org/10.1007/s13735-016-0095-6

[4] Klerk MG De, Parameter Analysis of the Jensen-Shannon Diver- Gence for Shot Boundary Detection in Streaming Media Applications. 109 (2018) 171–181

[5] Mishra R, Video Shot Boundary Detection Using Hybrid Dual Tree Complex Wavelet Transform with Walsh Hadamard Transform. (2021).

[6] Hato E, Abdulmunem ME, Fast Algorithm for Video Shot Boundary Detection Using SURF features. SCCS 2019 - 2019 2nd Sci Conf Comput Sci. (2019) 81–86. https://doi.org/10.1109/SCCS.2019.8852603

[7] Lakshmi PGG, Domnic S, Walsh-Hadamard Transform Kernel-Based Feature Vector for Shot Boundary Detection, IEEE Trans Image Process. 23 (2014) 5187–5197. https://doi.org/10.1109/TIP.2014.2362652

[8] Wu L, Zhang S, Jian M, Lu Z, Wang D, Two Stage Shot Boundary Detection via Feature Fusion and Spatial-Temporal Convolutional Neural Networks, IEEE Access. 7 (2019) 77268–77276. https://doi.org/10.1109/ACCESS.2019.2922038

[9] Mishra R, Raman C V, Singhai SK, Sharma M, Real Time and Non Real Time Video Shot Boundary Detection Using Dual Tree Complex Wavelet Transform. 2015 Int Conf Ind Instrum Control ICIC. (2015) 1495–1500. https://doi.org/10.1109/IIC.2015.7150986

[10] Gygli M, Ridiculously Fast Shot Boundary Detection with Fully Convolutional Neural Networks, arXiv. (2017) 1–4

[11] Chakraborty S, Thounaojam DM, A Novel Shot Boundary Detection System Using Hybrid Optimization Technique, Appl Intell. 49 (2019) 3207–3220. https://doi.org/10.1007/s10489-019-01444-1

[12] Chakraborty S, Thounaojam DM, SBD-Duo: A Dual Stage Shot Boundary Detection Technique Robust to Motion and Illumination Effect, Multimed Tools Appl. 80 (2021) 3071–3087. https://doi.org/10.1007/s11042-020-09683-y

[13] Rashmi BS, Nagendraswamy HS, Video Shot Boundary Detection Using Block Based Cumulative Approach, Multimed Tools Appl. 80 (2021) 641–664. https://doi.org/10.1007/s11042-020-09697-6

[14] Liang R, Zhu Q, Wei H, Liao S, A Video Shot Boundary Detection Approach Based on CNN Feature. Proc - 2017 IEEE Int Symp Multimedia, ISM. (2017) 489–494. https://doi.org/10.1109/ISM.2017.97

[15] Idan ZN, Abdulhussain SH, Mahmmod BM, Al-Utaibi KA, Al-Hadad SAR, Sait SM, Fast Shot Boundary Detection Based on Separable Moments and Support Vector Machine, IEEE Access. 9 (2021) 106412–106427. https://doi.org/10.1109/ACCESS.2021.3100139

[16] Nandini HM, Chethan HK, Rashmi BS, Shot Based Keyframe Extraction Using Edge-LBP Approach, J King Saud Univ - Comput Inf Sci. (2020). https://doi.org/10.1016/j.jksuci.2020.10.031

[17] Zhou S, Wu X, Qi Y, Luo S, Xie X, Video Shot Boundary Detection Based on Multi-Level Features Collaboration, Signal, Image Video Process. 15 (2021) 627–635. https://doi.org/10.1007/s11760-020-01785-2

[18] Shen RK, Lin YN, Juang TTY, Shen VRL, Lim SY, Automatic Detection of Video Shot Boundary in Social Media Using a Hybrid Approach of HLFPN and Keypoint Matching, IEEE Trans Comput Soc Syst. 5 (2018) 210–219. https://doi.org/10.1109/TCSS.2017.2780882

[19] Sulaiman AK, Mahmood SA, Shot Boundaries Detection Based Video Summary Using Dynamic Time Warping and Mean Shift, Proc Int Conf Comput Sci Softw Eng CSASE. (2020) 278–283 . https://doi.org/10.1109/CSASE48920.2020.9142116

[20] Singh A, Meitei D, Saptarshi T, A Novel Automatic Shot Boundary Detection Algorithm : Robust to Illumination and Motion Effect. Signal, Image Video Process. (2019). https://doi.org/10.1007/s11760-019-01593-3

[21] Malinski L, Smolka B, Fast Averaging Peer Group Filter for the Impulsive Noise Removal in Color Images, J Real-Time Image Process. (2015). https://doi.org/10.1007/s11554-015-0500-z

[22] Prabavathy AK, Shree JD, Histogram Difference with Fuzzy Rule Base Modeling for Gradual Shot Boundary Detection in Video Cloud Applications, Cluster Comput. (2017). https://doi.org/10.1007/s10586-017-1201-0

[23] Liu F, Wan Y, Improving the Video Shot Boundary Detection Using the HSV Color Space and Image Subsampling. 30 (2015) 351–354

[24] Bi C, Yuan Y, Zhang J, Shi Y, Xiang Y, Wang Y, Zhang R, Dynamic Mode Decomposition Based Video Shot Detection, IEEE Access 6. (2018) 21397–21407. https://doi.org/10.1109/ACCESS.2018.2825106

[25] Selesnick IW, Baraniuk RG, Kingsbury NG, The Dual-Tree Complex Wavelet Transform ©. 123–151

[26] Prathiba T, Kumari RSS, Eagle Eye CBVR Based on Unique Key Frame Extraction and Deep Belief Neural Network. Wirel Pers Commun. 116 (2021) 411–441. https://doi.org/10.1007/s11277-020-07721-4

[27] Tharwat A, Gabel T, Parameters Optimization of Support Vector Machines for Imbalanced Data Using Social Ski Driver Algorithm, Neural Comput Appl. 32 (2020) 6925–6938. https://doi.org/10.1007/s00521-019-04159-z

[28] Birinci M, Kiranyaz S (2014) A perceptual scheme for fully automatic video shot boundary detection. Signal Process Image Commun 29:410–423 . https://doi.org/10.1016/j.image.2013.12.003

[29] Fan J, Zhou S, Siddique MA, Fuzzy Color Distribution Chart -Based Shot Boundary Detection, Multimed Tools Appl. 76 (2017) 10169–10190. https://doi.org/10.1007/s11042-016-3604-y

[30] Bendraou Y, Essannouni F, Aboutajdine D, Salam A, Shot boundary Detection via Adaptive Low Rank and SVD-Updating, Comput Vis Image Underst. 161 (2017) 20–28. https://doi.org/10.1016/j.cviu.2017.06.003

[31] S. Chavate, R. Mishra and P. Yadav, A Comparative Analysis of Video Shot Boundary Detection using Different Approaches, 10th International Conference on System Modeling & Advancement in Research Trends (SMART). (2021) 1-7. Doi: 10.1109/SMART52563.2021.9676246.