International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 1, January 2014)

Key Frame Extraction Using Features Aggregation

B. F. Momin[1], S. B. Pawar[2]
[1]Prof. [2]Student, at Walchand College of engineering, Sangli, India.

***Abstract*— In Video Surveillance System, the surveillance of video in its different application such as performing real time online event detection, crime prevention, scene analysis and offline analysis and retrieval of interested events requires very huge computation and memory too. Key frame Extraction (KFE) is selection of frames which represents the object moves and changes in subsequent frames in the video. KFE may be used as pre-processing in surveillance application. Changes in frames are decided on the basis of frame's features. To describe frame in better way, more than one feature are used. Visual Feature Differences are found which are calculated by considering local threshold and Frame Difference is by considering global threshold. Three features are calculated for finding Frame Difference. Three feature differences are then aggregated using aggregation functions. Different Weights for different features are set according to the contribution of corresponding feature in finding frame difference to get better results.**

***Keywords*— Aggregation Function, Feature Extraction, Key frame, Thresholds, Video Summarization.**

I. INTRODUCTION

In Video Surveillance, Video analytics applications are mainly based on video processing. Videos may possess redundancy and useless data which is not required to be processed. In video processing, to reduce computation and computation time using less memory selection of cream of video is required. Among the key elements for the success of surveillance applications is how to effectively and efficiently manage and store a huge amount of visual information and providing efficient access to the stored data. To store this huge video data required memory space is very large. For the storage of this large data information in minimum space and make this data storage efficient, there is a need of abstract version of the video data which describes total scene. 'Video Abstraction' is the corresponding research area and key frame extraction is core part of it. A good video abstract will enable to gain maximum information about the video sequence in a specified time or sufficient information in the minimum time.

Two main methods of video abstraction are there; first one is 'Key Frame Extraction' and another is 'Video Skimming'.

Former is static way and its output is set of still images from the original video and is better for video summarization than later one which is dynamic and whose output is a video clip of the original video. Key frame extraction can be used as a pre-processing step in video analytics applications which suffer from the problem of processing large number of video frames. Key frames are video representative as they hold most important content of the video. Key frames helps to get maximum information in minimum time.

In the proposed method KFE is done by finding subsequent frame differences. Frame difference is calculated by finding feature differences. Different features describe frame in different way, so to find more accurate frame difference more than one feature are considered and feature differences are calculated.

This paper is structured as follows, section 2 discuss about literature review. Section 3 describes the KFE with the help of proposed methodology. Section 4 discuss about calculation of different features and feature difference. Section 5 gives the details about aggregation function used for aggregating feature differences to find frame difference. Key frame selection is discussed in Section 6. Section 7 is about experimentation and respective results. Finally, conclusion is drawn in section 8.

II. LITERATURE REVIEW

This Section deals with the various methods used and introduced in area of video abstraction.

Static and Dynamic are two kinds of video abstraction methods. As discussed above, key frame Extraction is static and it generates a set of key frames selected from video, whereas video skimming generates a short video sequence with extracted audio features.

There are different techniques for key frame selection one of them is by computing subsequent frame difference exceeds threshold. Various frame difference measures are used such as low level features measure, geometrical information measure [2]. Video is represented by a trajectory curve, and frames at discontinuous points on curve are selected as key frames [13]. Adaptive Sampling Algorithm selects the key frames by uniformly sampling the y-axis of the curve of cumulative frame difference and the resulting sampling on x-axis represents the key frame.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 1, January 2014)

Clustering is the second most important method for key frame extraction. Clustering is based on similarity measures and selects one key frame per cluster. In this method mostly researchers consider video frame as point in feature space [9]. In [12] sequential clustering method is used that assigns the current frame to existing cluster if there similarity is maximum and exceeds threshold, otherwise create a new cluster. Dominant set clustering is one another novel method for key frame extraction [9]. In [11], Clustering based on mutual information curve approach is presented. In this approach, probability density and entropy value of two successive frames are computed based on RGB color space. Frames are clustered according to the mutual information curve after building a mutual information curve for all consecutive frames.

A hybrid user attention model is illustrated in [11], which includes visual as well as audio features in video summarization. [17] Gives a max-min distortion optimization method to extract key frames from video. They address the problem by pre determined number of frames that minimizes the temporal distortion and adopted dynamic programming technique to optimize the process. Object based video abstraction, content based video abstraction are some other techniques for key frame generation [1]. Different visual attention model provides a well defined semantic of video content. [10] Gives saliency based visual attention model, where saliency map can be indication of the attention model for determination of the key frames. In intelligent video applications, the combines several different representative feature maps into single saliency map. To deal with the problem of contextual attention understanding, a well-defined contextual attention model based on human perceptual characteristics is presented in [3]. They presented a novel attention-based key frame extraction system by using both the object-based visual attention model and contextual attention-model and integrate the object-based visual attention map and the contextual on-going outcomes. This model determines the human perceptual characteristics as well as the type of interested video content effectively.

KFE based on frame difference is quiet easy method. These methods do not consider temporal information but preserves them. Clustering is similarity measure based method of KFE. Clustering generates less redundancy as compared to consecutive frame difference method. This does not consider temporal information and may not preserve temporal visual sequence. One more limitation of clustering is, cannot be implemented until all frames in video sequence are analyzed.

Some of the techniques given above use Region of Interest (ROI) based key frame extraction to find semantically relevant key frames.

III. METHODOLOGY

A standard video abstraction by Key Frame Extraction method selects salient frames and generates a set of key frame. This method is based on frame difference strategy of finding key frames. The method explained below is referred to [2] with some changes.

A video (V) to be summarized contain N frames.

$$V= \{F_{tn}\}, \quad n=0, 1, 2....N \qquad (1)$$

Where, F_t –frame in video at time t.

t – Unit time for single frame generation.

$$KF= \{KF_m\}, \quad m= 0, 1, 2,...M \text{ and } 1\leq M\leq N \qquad (2)$$

Where,

KF_m- m^{th} Selected Key Frame.

The proposed method in [2] is poured into following algorithm.

Algorithm 1:

Pre requisite is divide video into frames.

Input: Candidate frames: F_i i=1, 2, 3.....n

Key Frames: KF_j, j=1, 2, 3, ...m;

D: Frame Difference.

τ: Pre-defined threshold

Flag – set to 1 if KF is newly generated

Step 1: Select F_1 as KF_1

flag=1;

$KF_1=F_1$;

i=i+1;

Step 2: if(i≤n)

D=Frame Difference (F_i, KF_j,flag) ;

else

goto Step 5;

Step 3: if (D≥τ)

j=j+1;

$KF_j=F_i$;

i=i+1;

flag=1;

else

flag=0;

i=i+1;

goto step 3;

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 1, January 2014)

Algorithm 2: Frame Difference (F_i, KF_j,flag)

Input: Current Candidate frame (F_i)
Current Key Frame (KF_j)
Flag;
Step 1: if (Flag==0)
goto Step 2.
else
H_{kf}=histogram (KFj);
Mkf=MOI(KFi);

Step 2: Hf=histogram(Fi);
Mf=MOI(Fi);

Step 3: H=histogram_difference(Hkf,Hf);
M=MOI_difference(Mkf,Mf);
C=correlation(KFj,Fi);
Step 4: $difference = W_1H + W_2M + W_3C$; **(3)**
Return(Difference);

Algorithm1 describes the overall methodology to select key frame.

IV. Frame Difference

Every feature describes unique characteristics of image; to find frame difference single feature is not enough. Three different features are extracted from frames, viz. Histogram, Moment of inertia and Correlation, where correlation finds similarity between two frames. Former two features are widely used for finding frame difference between two frames in key frame extraction methods. Frame is first divided into "Ts (size: m×n)" non-overlapping sections while finding feature differences; consequently reduces computing complexity and time. Each feature difference is computed between each corresponding sections of two images and then finally aggregates these values to find final frame difference between two frames.

A. Colour Histogram Difference

Colour Histogram difference estimates the frame difference based on colour [2]. This measure is selected as it is simple and its robust nature towards small changes in camera motion. This difference is computed over HSV colour model instead of RGB. HSV has the ability to provide sensitive representation of colour closer to human perception.

HSV refer to the hue saturation and value, where value gives the strength of brightness, also called as HSB. This helps in finding the lightening effect change in between frames. Colour histogram frame difference is calculated in two parts, calculation of histogram and finding difference.

Histogram is calculated in **5** steps as follows:

Step1: Convert RGB values of frame into HSV by using following formulae:

$$Hue = \arctan\frac{\sqrt{3}(G-B)}{\sqrt{(2R-G-B)}} \quad (4)$$

$$Saturation = \frac{\max(R,G,B)-\min(R,G,B)}{\max(R,G,B)} \quad (5)$$

$$value = \max(R,G,B) \quad (6)$$

Step2: Draw colour histograms of each hue, saturation and value component.

Step3: colour quantization: to reduce size of the colour histogram, hue component histogram is reduced to 16 bins and other two are reduced to 8 bins each.

Step4: Normalization of HSV components in the range 0-1.

Step5: Combination of three histograms to form a single histogram of size 32 bin.

Step6: Obtain histogram difference between corresponding sections of the frame by using Euclidean distance.

$$H(f_t, f_t^{'}) = \sqrt{\sum_{i=1}^{32}\left(H_t(i) - H_t^{'}(i)\right)^2} \quad (7)$$

Where, H_t – histogram of t^{th} section of frame f.

$$H_{fd} = \frac{1}{T_s}\sum_{ts=1}^{T_s=9} H_{ts}\left(f_t, f_t^{'}\right) \quad (8)$$

B. Moment of Inertia Difference

Three moments of inertia viz. mean, variance and skewness, for each colour channel are used for calculation of 9 moments of each section of fame. For frame f, t^{th} section and colour channel c and mean saturation and skewness values are computed as in [2][16]:

$$mean(f_{t,c}) = \frac{1}{m\times n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} f_{t,c}(i,j) \quad (9)$$

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 1, January 2014)

$$\sigma^2(f_{t,c}) = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (f_{t,c}(i,j) - mean(f_{t,c}))^2 \quad (10)$$

$$\gamma(f_{t,c}) = \frac{1}{m \times n} \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (f_{t,c}(i,j) - mean(f_{t,c}))^3}{(\sigma^2(f_{t,c}))^{3/2}} \quad (11)$$

The moment of inertia feature vector (μ) of frame f is formed by combining these values. Size of this vector is $9 \times T_s$ as 9 moments of inertia of T_s sections are combined. Feature difference is calculated by using Euclidean distance and this feature difference gives moment of inertia difference between two frames f and f'.

$$M_{FD} = \mu(f, f') = \sqrt{\sum_{i=1}^{9T_s} \mu(i) - \mu'(i)} \quad (12)$$

C. Correlation Difference

The correlation coefficient computes similarity between two frames. Correlation is a pixel wise relation between two frames. First divide frames into non-overlapping sections. Instead of computing correlation coefficient for full frame in a stroke, coefficient is calculated for corresponding sections of two frames to be compared and one for each colour channel. Let f and f' be the two frames for the calculation of correlation coefficient. Each frame has been divided into ''Ts'' sections of size pxq. Then the correlation coefficient for a section ''t'' for frames and for colour channel "c" is given by [2]:

$$cor(f, f')_{t,c} = \frac{\sum_{i=0}^{p-1} \sum_{j=0}^{q-1} (f_{t,c}(i,j) - mean(f_{t,c}))(f'_{t,c}(i,j) - mean(f'_{t,c}))}{\sqrt{\sum_{i=0}^{p-1} \sum_{j=0}^{q-1} (f_{t,c}(i,j) - mean(f_{t,c}))^2 \sum_{i=0}^{p-1} \sum_{j=0}^{q-1} (f'_{t,c}(i,j) - mean(f'_{t,c}))^2}}$$

(13)

To compute over all correlation for each colour channel, mean of all sections for respective colour channel computed above is taken as [2]:

$$cor(f, f')_c = \frac{1}{T_s} \sum_{k=1}^{T_s} cor(f, f')_{k,c} \quad (14)$$

The correlation coefficient measure is calculated by taking mean of all three colour correlation coefficient.

$$C_{FD} = \frac{cor(f, f')_R + cor(f, f')_G + cor(f, f')_B}{3} \quad (15)$$

V. Key Frame Selection

Key frame selection depends on calculated frame difference. Frame difference is calculated using algorithm1 and aggregation function in eqn. (3). This function aggregates contributing values of each feature. Contributing value is computed on basis of local threshold and Feature difference computed as discussed above. Local threshold validates the feature difference and make corresponding positive contributing value. Contributing value for each feature is generated using following equations [2].

$$H = \begin{cases} 1 + |H_{FD} - \tau_h| & if H_{FD} < \tau_h \\ -|H_{FD} - \tau_h| & otherwise \end{cases} \quad (16)$$

$$M = \begin{cases} 1 + |M_{FD} - \tau_m| & if M_{FD} < \tau_m \\ -|M_{FD} - \tau_m| & otherwise \end{cases} \quad (17)$$

$$C = \begin{cases} 1 + |C_{FD} - \tau_c| & if C_{FD} < \tau_c \\ -|C_{FD} - \tau_c| & otherwise \end{cases} \quad (18)$$

τ_h, τ_m and τ_c are local thresholds for features histogram, moments, correlation respectively.

VI. Experimentation

The experimental setup consists an Intel Core i3 processor with 2GB RAM and with windows7 operating system.

Experimentation is carried out over 10 videos with different values of threshold parameter to set best value as threshold. On the basis of experimentation the probable and best suitable value for threshold is 1 to 1.5.Sometimes for different kinds of video this value may change.

Some Sample results are tabulated as shown below:

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 1, January 2014)

TABLE I
OBSERVATION TABLE

Video (Size in sec)	No. of Frame (Size)	Threshold	Key Frames	
			Manual	Extracted
1 (1sec)	30(704×576)	1	8	9
		1.25		4
		1.5		1
2 (10 sec)	250(160×120)	1	5	60
		1.5		1
		2		1
3 (60 sec)	1564 (720×480)	1	50	80
		1.5		32
		2		25
4 (218 sec)	5700 (704×576)	1	150	176
		1.5		160
		2		50

Frame size of videos used for experimentation in [2] is very small (160×120); time required for processing video is less near about some minutes depends on video size but for large frame size (720×480) videos this method takes some hours and it is insignificant.

VII. CONCLUSION

Aggregation function used here to aggregate benefits of three different features for key frame selection. Threshold value set here may have to change for different videos for getting best result, so automatic threshold value generation will be focused in future.

Future work will focus on validating the proposed approach on larger scale video datasets and also on storyboard generation for better representation of result.

Acknowledgement

This work is part of research & development project titled "Surveillance Data Mining System" under Research Promotion Scheme (RPS), AICTE New Delhi, 2013-16.

REFERENCES

[1] Q. Ji, Z. fong, Z. Xie and Z. Lu, "Video Abstraction Based on Visual Attention Model and Online Clustering", Elsevier B.V. on Signal Processing: Image Communication, 28(2013), pp.241-253.

[2] N. Ejaz, T. Tariq and S. Baik, "Adaptive Key Frame Extraction using an Aggregation Mechanism", Elsevier Inc J. Vis commun Image R. 23(2012), pp. 1031-1040.

[3] H. Shih, "A Novel Attention- Based Key-Frame Determination Method", IEEE Trans. on Broadcasting, May 2013.

[4] J. Besocs, G. Cisneros, J. Martinez and J. Menendez, "A Unified Model on Techniques on Video Shot Transition Detection", IEEE Trans. on Multimedia, Vol. 7 no. 2, pp. 293-307, 2005.

[5] J. Yu, M. Srinath, "An Efficient Method for Scene Cut Detection", Elsevier Pattern Recognition Letters, 22(13), 2001.

[6] P. Kathiriya, D. Pipalia, G. Vasani, A. Thesiya and D. Varanva, "X^2 (Chi-Square) Based Shot Boundary Detection and Key Frame Extraction for Video", International Journal of Engineering and Science, ISSN: 2278-4721, Vol. 2, pp 17-21, Jan 2013.

[7] K. Sze, K. Lam and G. Qiu, "A New Key Frame Representation for Video Segment Retrieval", IEEE Trans. on Circuits and Systems for Video technology, Vol. 15, no. 9, sept. 2005.

[8] R. Mishra and S. Singhai, "A Review on Different Methods of Video Shot Boundary Detection", International Journal of Electrical and Electronics Engineering (IJEEE), Vol.1, pp. 46-57, Aug 2012.

[9] X. Zeng, W. Hu, W. Li, X. Zhang and B. Xu, "Key-Frame Extraction using Dominant-Set Clustering", pp. 1285-1288.

[10] X. Zeng, W. Hu, W. Li, X. Zhang and B. Xu, "Key-Frame Extraction using Dominant-Set Clustering", pp. 1285-1288.

[11] Y.-F. Ma, X.-S. Hua, L. Lu, and H.-J. Zhang, "A generic framework of user attention model and its application in video summarization," IEEE Trans. Multimedia, vol. 7, no. 5, pp. 907–919, Oct. 2005.

[12] B.T. Truong, S. Venkatesh, "Video Abstraction: A Symentic Review and Classification", ACM Trans. Multimedia Comput.Commun. Appl. 3, 1, Article (Feb 2007).

[13] D. DeMenthon, V. Kobla, D. Doermann, "Video summarization by curve simplification", in: Proc. ACM International Conference on Multimedia, NewYork, USA, 1998, pp. 211–218.

[14] S.E.D. Avila, A.B.P. Lopes, L.J. Antonio, A.d.A. Araújo, "VSUMM: a mechanism designed to produce static video summaries and a novel evaluation method", Pattern Recognition Letters 32 (1) (2011) 56–68.

[15] J. Flusser, T. Suk, B. Zitove, Moments and Moment Invariants in Pattern Recognition, Wiley & Sons Ltd, 2009.

[16] J. Flusser, T. Suk, B. Zitove, Moments and Moment Invariants in Pattern Recognition, Wiley & Sons Ltd, 2009.

[17] Z. Li, G.M. Schuster, A.K. Katsaggelos, Minmax optimal video summarization, IEEE Transactions on Circuits and Systems for Video Technology 15 (10) (2005) 1245–1256.