Indonesian Journal of Electrical Engineering and Computer Science
Vol. 7, No. 3, September 2017, pp. 818 ~ 829
DOI: 10.11591/ijeecs.v7.i3.pp818-829

A meaningful Compact Key Frames Extraction in Complex Video Shots

Manar A. Mizher[1], Mei Choo Ang*[2], Ahmad A. Mazhar[3]
[1]Institute of Visual Informatics, Universiti Kebangsaan Malaysia, Malaysia
[2,3]College of Computing and Informatics, Saudi Electronic University, Saudi Arabia
Corresponding author, e-mail: manar@siswa.ukm.edu.my[1], amc@ukm.edu.my*[2], a.mazhar@seu.edu.sa[3]

Abstract

Key frame extraction is an essential technique in the computer vision field. The extracted key frames should brief the salient events with an excellent feasibility, great efficiency, and with a high-level of robustness. Thus, it is not an easy problem to solve because it is attributed to many visual features. This paper intends to solve this problem by investigating the relationship between these features detection and the accuracy of key frames extraction techniques using TRIZ. An improved algorithm for key frame extraction was then proposed based on an accumulative optical flow with a self-adaptive threshold (AOF_ST) as recommended in TRIZ inventive principles. Several video shots including original and forgery videos with complex conditions are used to verify the experimental results. The comparison of our results with the-state-of-the-art algorithms results showed that the proposed extraction algorithm can accurately brief the videos and generated a meaningful compact count number of key frames. On top of that, our proposed algorithm achieves 124.4 and 31.4 for best and worst case in KTH dataset extracted key frames in terms of compression rate, while the-state-of-the-art algorithms achieved 8.90 in the best case.

Keywords*: Optical Flow, Lucas–Kanade, Frame Differences, Blocks differential, TRIZ*

1. Introduction

Key frames extraction is the first step in detecting and segmenting video contents, video summarization, or video retrieval regardless of any pre-processing procedures. Key frames are used instead of all frames in the video for video processing and they should brief the salient content and objects with an excellent feasibility, great efficiency, and with high-level of robustness [1].

There are many visual features that may be found inside a frame used for key frame extraction. The visual features including static features (color-based, texture-based, and shape-based), objects features, motion features (camera-based and object-based) [2]. However, texture, shape and objects features are difficult to be detected and not always available in video. On the other hand, the color feature is simple, efficient to reflect human visual perception, easy to be extracted and has low computational complexity, but it is ineffective in videos with important objects or shapes. Motion features include background motion occurred by camera motions and foreground motion occurred by moving objects in a video. Camera-based motion features can be caused by various camera movements such as zooming, swing, and tilting. Camera-based motion features are useful in video indexing, but it is not effective to depict motions of important objects in video retrieval. Object-based motion features are more popular recently. Object-based motion features can be grouped into a number of classes based on statistics, trajectory, and objects relationships [2, 3]. The statistics-based motion features have low computational complexity, but they are unable to depict relations between objects and unable to capture object actions accurately. The trajectory-based motion features are used to describe object actions but the extraction of these features is a very challenging task. The objects relationship-based motion features are used to represent relationships between multiple objects but objects labeling and their positioning labeling is also a very difficult task [2].

Motion feature vector is used to supply a compact video representation while protecting the important actions of the original video [4]. Motion vector is calculated by a motion estimation technique to describe the visual contents with temporal differences inside a video. There are

Received May 9, 2017; Revised July 15, 2017; Accepted August 2, 2017

several methods for motion estimation such as feature matching, pel-recursive, deterministic model, stochastic model, and optical flow. In addition, there are several techniques which have been developed to approach the computation of optical flow such as gradient, correlation, spatio-temporal energy, and phase [5].

The Lucas–Kanade method is widely used in differential methods for optical flow estimation and computer vision by calculating the motion vector between two frames which are taken at specific time. The advantages of the Lucas-Kanade method are: fast calculation, suitability for real world tasks implementations, the accurate time derivatives on both real and synthetic image sequences, producing accurate depth maps, and good noise tolerance. The disadvantage of this technique is the rate of errors on boundaries of moving object.

Authors in [6] proposed a real time motion detection algorithm based on the integration of accumulative optical flow and twice background filtering technique. Lucas-Kanade method was employed to compute frame-to-frame optical flow to extract a 2D motion field. The accumulative optical flow method was used to cope with variations in a changing environment and to detect movement pixels in the video. The twice background filtering method was used to extract moving object from the background information. The advantages of the algorithm are: avoiding the need to learn the background model from a large number of frames, and it can handle frame variations without prior knowledge of the object shapes and sizes. The algorithm was reported to be able to detect tiny objects and even slow moving objects accurately. Many authors focused on motion estimation and analysis to extract key frames to protect important actions of original video and to provide a compact video representation. However, most of the optical flow techniques are poor due to affected by motion discontinuities and noise [7].

Generally, the technique for key frame extraction should provide a compact video representation, but it should not be a complex and time consuming process. Key frame extraction techniques can be categorized into one of these six groups: sequential comparison between frames, global comparison between frames, reference frame, clustering, curve simplification, and based on objects or events [1, 2, 8].

A variety of different key frame extraction techniques developed based on frame difference, frame blocks differential, motion estimation and clustering were improved in recently to extract key frames to segment video into shots or scenes. The simplest was the selection of key frames by calculating the color histogram difference between two consecutive frames, then computing the threshold based on the mean and the standard deviation of absolute difference. After that, comparing the difference with the threshold if it is larger then select the current frame as a keyframe. Steps were repeated till end of the video to extract all keyframes [9]. In another typical method [10], key frames were extracted by comparing the consecutive frame differences with the threshold value. The algorithm read each frame in a video and converted them into grey level, and then it calculates the differences between two consecutive frames. The algorithm then calculates the mean, standard deviation, and the threshold which is equal to standard deviation multiply by a constant number. If the difference is larger than the threshold the current frame will be saved as key frame. A key frame extraction algorithm based on frame blocks differential accumulation with two thresholds was proposed in [11]. In their algorithm, the first frame in a video is considered as a first reference frame. The remaining video frames are then partitioned into equal sized image blocks. The created image blocks are used to detect any local motion in the video. The color mean differences are computed in RGB color space of the corresponding blocks in the reference frame and the current frame. The algorithm counts the blocks changing in the current frame in relation to the block changing in the reference frame. If the count number is greater than the global threshold, this means the current frame has more changes than the reference frame. Then, the algorithm uses the current frame as key frame instead of the reference frame and similar steps will be repeated until the last frame. This algorithm was reported to show high efficiency to identify movements and extract key frames with strong robustness in different types of video.

The majority of key frame extraction algorithms was developed to segment videos into shots or scenes. There are very few researchers focused on extracting key frames within the video shots with the camera moving, shaky camera, or dynamic background. However, it is considered important in some fields which need a compact presentation of video scenes such as in video forgery detection system based on fingerprint [12], video watermarking [13], video copyright protection [14], and video summarization [15, 16]. Therefore, this paper generated meaningful compact key frames using accumulative optical flow with self-adaptive threshold inspired by the TRIZ inventive principles. The extracted key frames can be used to represent the video as a whole and summarizes the important objects and the salient events of the video. In Section 2, key frames extraction features based on TRIZ tools are analyzed, and key frames extraction algorithm within the video shot is presented; Section 3 argues the experiments and the analysis of the result. Finally, Section 4 concludes the work and suggests future work.

2. Materials and methods

The extraction of key frames is the first step in video for detecting and segmenting moving objects, detecting interesting regions, removing unwanted regions or objects, or reconstructing and repairing damaged areas. Efficient algorithms for key frame extraction for video sequences are highly desirable in the area of multimedia indexing and data retrieval due to the challenges, including: moving objects with dynamic texture background, moving camera, or shaky camera. The key frames extraction within a scene is an easier task than extracting them inside a shot. The scene has a transition between two sequential shots and different views from shot to shot, while the shot is a sequence of successive frames captured without interruption by a same camera. Derive techniques to perform key frame extraction effectively inside shots is a challenging task because video frames are attributed to many visual features such as motion and color.

2.1. Analyzing Key Frames Extraction Features Based on TRIZ Tools

TRIZ is a Russian acronym for the Theory of Inventive Problem Solving (TIPS); it was developed by Genrich Altshuller in 1946. TRIZ was applied in many disciplines and showed some promising results [17-21]. TRIZ is known to provide a systematic approach to solve innovative problems. It has many tools which include contradiction matrix, technical systems, levels of innovation, ideality, and many more [22]. In this initial investigation, we applied one of the popular TRIZ tools, namely, contradiction matrix, to study the effect of using static and motion features and techniques on extracting key frame accuracy. The outcome of the TRIZ systematic problem solving approach and the contradiction matrix will be a collection of TRIZ recommended principles. Based on the ideas inspired by the TRIZ recommended principles, we then derived a key frame extraction algorithm to perform inside a shot as it is an important issue for some systems such as detecting video forgeries.

The TRIZ process involved for this key frame extraction problem includes: problem identification, cause and effect chain analysis, contradiction matrix, ideation using TRIZ principles.

2.1.1. Problem identification

At the start, some questions were raised as follows to help us understand the key frame extraction problem:

A. Why it is difficult to choose any visual feature to extract the key frames?

Because every visual feature has its own advantages and disadvantages, and limited by the dataset environment and camera conditions.

B. Why it will be a problem when choosing a wrong visual feature to extract the key frames?

Because it will affect the results in the video processing, and it will decrease the accuracy of the algorithm.

Good features will be those that are able to detect the most meaningful keys and extract compact count number of keys. The benefit if the key frame extraction problem is solved: it will increase the algorithm accuracy of the result.

Further analysis of the advantages and disadvantages of the existing algorithms to extract the key frames based on different visual features was conducted [23]. Several key frames techniques are presented in Table 1. The extracted key frames method should compact the important action in video with efficiency, feasibility, and robustness. It should avoid computational complexity, and reduce as much redundancy as possible. The standard key frame extraction methods can be classified in four categories which are: video shot method, content analysis method, cluster method, and motion analysis method [24]. The advantages and disadvantages of the key frame extraction algorithms are highlighted in Table 1.

A common issue in these standard key frame extraction algorithms is that they were used to segment videos into shots or scenes without focusing on extracting key frames within shots. This issue is essential in some fields and it needs a compact summarization for video shots such as in video forgery detection system based on fingerprint. Falsifying motion or faked objects may occur within a single shot of the video and it can be conducted within the video shots with a moving camera, shaky camera, or dynamic background.

The key frames techniques modelling were used to illustrate the key frames extraction problem as an engineering system by defining the interaction between features and techniques. The static features and motion are the main features in the key frames techniques problem. The majority of key frame extraction techniques use these features with threshold to detect and extract key frames; these are shown in Figure 1. In the key frame techniques modelling, the main system features (mainly the color and motion) that play a role in causing accuracy improvement are surrounded by a rounded rectangles of Figure 1.

Table 1. The analysis of the key frame extraction techniques within video

Classification	Advantages	Disadvantages
Content-based Method	Very simple, select key frames according to the change of content	Unsteady key frame extraction with motion objects/background, affected by noise, not always the most representative meaning
Cluster-based Method	Accurate result	Difficult to implement, more computational cost, difficult to find general cluster parameters, needs pre-processing steps
Video shot-based Method	Temporal and spatial information, easy, simple, low computation complexity, quick extraction, representative meaning	Does not consider the complexity of content, fixed value of key frames, does not efficiently describe the motion content
Motion-based Method	Easy to implement, detect motion, select the appropriate number of key frames	Most suitable for static camera and background, high computation, low robustness (local motion and does not consider the content cumulative dynamic changes)
Selected fixed key frames	Easy to implement, fast	Missed sufficient information

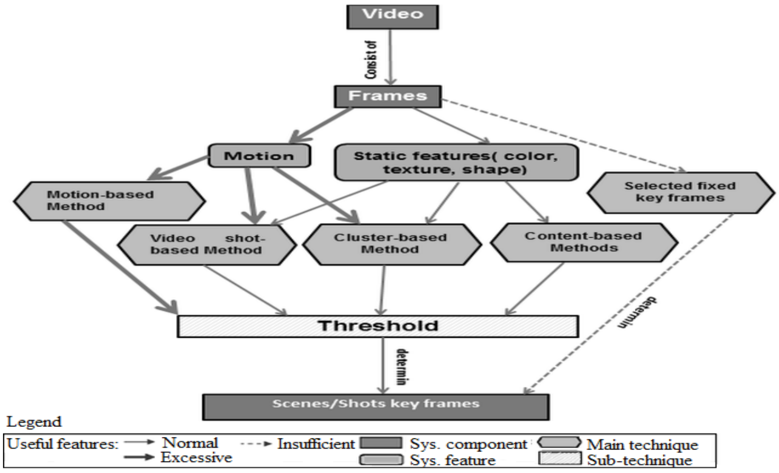

Figure 1. Techniques modelling for key frame extraction problem

2.1.2. Cause and Effect Chain Analysis

The cause and effect chain analysis will look into the possible root causes that relate to key frames detection features and extraction techniques trying to solve the accuracy problem. Based on this problem statement, the cause and effect chain analysis can be summarized in the following questions:

(a) Why does key frames extraction problem occur?

The causes that contribute to this problem are the imprecise in determining the main features (static features, motion) threshold causes low accuracy in the final result. And the difficulty in determining the threshold happened because it is very sensitive and heavily dependent on the dataset used.

(b) Why is it difficult to determine the threshold?

It is attributed to many possible causes. The static feature detection techniques are affected by noise or changing conditions during the video such as, shadow, light on/off. Motion detection technique is not suitable for moving background/camera and shaky camera. The cause and effect chain analysis can be illustrated as in Figure 2.

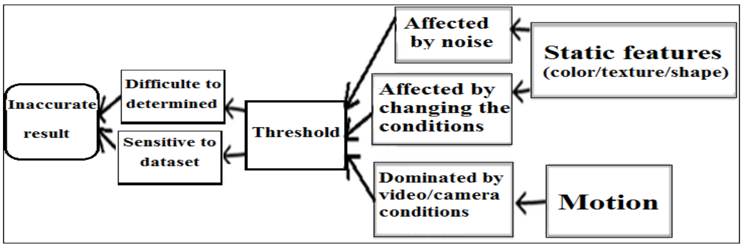
Inaccurate result
Difficulte to determined
Sensitive to dataset
Threshold
Affected by noise
Affected by changing the conditions
Dominated by video/camera conditions
Static features (color/texture/shape)
Motion

Figure 2. Cause and effect chain analysis of key frames extraction problem

2.1.3. Contradiction Atrix

The development of the solutions was based on TRIZ engineering contradictions that were derived from possible root causes. The details of these engineering contradictions including the improving and worsening features are related to each root cause. These contradictions are summarized in Table 2 where the recommended inventive principles to solve the respective root cause were determined as well.

Table 2. Summary of TRIZ tools process to solve the key frames extraction problem

Problem	Root cause level 1	Root cause level 2	Contradiction	Improving feature	Worsening feature	Inventive Principles	Solutions
Key frames extraction	Difficult to determine	Affected by noise	Static features with threshold save time, but affected by noise	Save time	Noise (37)	35- Parameter changes	- Need to change static features threshold parameters.
		Affected by changing the conditions	Static features with threshold easy to implement, but affected by changing the conditions	Easy to	Dominated by video conditions (35)	1- Segmentation 16- Partial	- Focusing on detecting static features on spatial domain. - Need to remove small regions to reduce the complexity.
	Sensitive to dataset	Dominated by video environment and camera conditions	Motion with threshold accurate result, but dominated by video conditions	Accurate	Dominated by video conditions (35)	35- Parameter changes 2- Segregation	- Need to change motion threshold parameters. - Focusing on detecting motion on temporal domain.
			Motion with threshold easy to implement, but dominated by video conditions	Easy to	Dominated by video conditions (35)	1- Segmentation 16- Partial	- Focusing on detecting motion on spatial domain. - Need to remove small motion to reduce the complexity.
			Motion with threshold save time, but dominated by video conditions	Save time	Dominated by video conditions (35)	35- Parameter changes	- Need to change motion threshold parameters.

2.1.3. Ideation using TRIZ Principles

The threshold can be classified into global, adaptive, or global and adaptive combined [2]. In addition, the motion estimation could be affected by different parameters such as the video environment conditions (static or dynamic objects with static or dynamic background and texture), video capturing conditions (static, shaky or moving camera), noise, technique used to estimate the moving objects' velocities, detection of spatial and temporal information. Based on the solutions inspired by TRIZ principles in Table 2, we have derived an algorithm, named as, Accumulative Optical Flow with Self-adaptive Threshold (AOF_ST) to detect and extract key frames on a dataset with different video environment cases and camera conditions.

The implemented TRIZ inspired solutions include:

1 Using a full self-adaptive threshold with variables: variable values could be calculated depending on motion feature to have a full dynamic system to overcome the difficulty and the sensitivity (35- parameter changes).

2 Focusing on reading motion on spatial domain using optical flow (1-segmentation).

3 For a temporal domain, using a fixed number of cumulative frames (2-segregation) will lead to reducing the effects of videos environment and camera conditions and to provide more accurate results.

Removing small motion (noise) to make it easier to implement and to reduce complexity (16-partial).

2.2. Accumulative Optical Flow with Self-Adaptive Threshold (AOF_ST)

AOF_ST is an algorithm that has two phases: pre-processing phase and key frames extraction phase. The flow chart of the proposed AOF_ST algorithm is shown in Figure 3. The details of each phase are discussed:

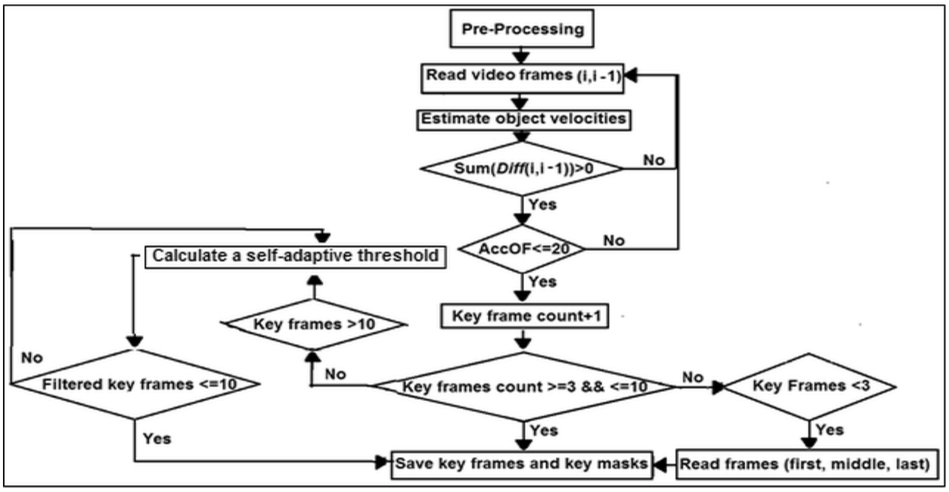
Pre-Processing
Read video frames (i,i -1)
Estimate object velocities
Sum(*Diff*(i,i -1))>0
No
Yes
AccOF<=20
No
Yes
Calculate a self-adaptive threshold
Key frames >10
Key frame count+1
No
Filtered key frames <=10
No
Key frames count >=3 && <=10
No
Key Frames <3
Yes
Yes
Yes
Save key frames and key masks
Read frames (first, middle, last)

Figure 3. Flow chart of AOF_ST algorithm

2.2.1. Preprocessing Phase

In this phase, a median filter was used to remove noise before resampling the video. The video frames were reconstructed using a specific frame rate resampling and resolution resampling size (320 W × 320 H), and a fixed rate (30 frames/second). In addition, we defined video quality equals to 100 where the quality is a number between 0 and 100. Higher quality numbers result higher video quality and larger file size. All videos were converted to .avi video format to avoid any unexpected problems through next phase implementation.

2.2.2. Key Frames Extraction Phase

This phase extracts the most important key frames which contain critical motion of objects or texture by estimating objects' velocities based on accumulative optical flow with fixed key frames between three to ten keys. For each video shot, and to have a limited number of key frames to make it easier to use them in different systems such as object detection, a video search with retrieval and generating a fingerprint is conducted. This will increase the level of security in the process of video forgery detection.

After the preprocessing phase, AOF_ST reads both consecutive frames f_i and f_{i-1} at time *i* from the resampled video shots. It converts frames to single precision and then using the optical flow to estimate objects' velocities from each frame based on Lucas-Kanade method. This process will generate coordinate points and draw lines to indicate flow. Equations 1 and 2 are used to calculate accumulative optical flow threshold depending on absolute frames difference summation.

$$FRdiff = diff(fi, fi-1) \quad (1)$$

$$AccKey = \sum_{m=1}^{c}\sum_{n=1}^{r}(FRdiff) \quad (2)$$

Where, *FRdiff* is the absolute difference frame between the two consecutive frames, f_i is the current frame, f_{i-1} is the previous frame. *AccKey* is the accumulative optical flow threshold, *c* and *r* are the *FRdiff* number of columns and rows respectively.

When *AccKey* is greater than zero, there is a noticed motion between f_i and f_{i-1} and based on *AccKey*, accumulate 20 frames for each key frame. We chose 20 as an average for accumulative key frames to be moderate and suitable for detecting moving objects with static or dynamic background, and this would be suitable with a static camera, moving camera or shaky camera videos.

The selection of key frames using AOF_ST depends on three cases. In the first case, if the key frames count is between three and ten, the key frames are saved with their masks. In the second case, if the key frames count is less than three, it means that the shot has no important motions to detect. Thus, choose the first, middle, and last frames as key frames set. In the third case, if the extracted key frames count is more than ten, it means the shot has highly dynamic texture and/or more motion objects need to be detected. In the third case, all detected key frames are flagged as temporally key frames, the self-adaptive threshold is used to filter them and delete key frames with less motion difference. Eq.3 calculates the absolute differences *Dydiff* between two consecutive temporally key frames. In Eq.4 based on *Dydiff*, the self-adaptive *DThreshold* calculates, and in Eq.5, when the *DThreshold* is greater than the initial *Dvalue*, the temporally key frame is saved as a filtered key frame. If it is less, then the *Dvalue* is increased by a constant number *con* until filtered all temporally key frames. The initial *Dvalue* and its increment were chosen by the user, and the final values determined based on implementing several comparisons. Initial *Dvalue* equals 10000 while the increment value *con* equals 5000.

$$Dydiff = diff(Mx, My) \quad (3)$$

$$DThreshold = \sum_{m=1}^{c}\sum_{n=1}^{r}(Dydiff) \quad (4)$$

$$Dvalue = Dvalue + con \text{ (5)}$$

Where, *Dydiff* is the absolute difference frame between the two consecutive temporally key frames, M_x is the current temporal key frame, M_y is the next temporally key frame. *DThreshold* is the self-adaptive threshold which compared with *Dvalue* to filtered key frames; *c* and *r* are the number of *Dydiff* columns and rows respectively. *Dvalue* is the dynamic variable increased by a *con* number, it improves the threshold because it adapts itself to the sequence statistics, which increase the performance of the extraction method [25].

3. Experiments and Result Analysis

The aim of the experiments is to see whether choosing visual features have a possible influence on the accuracy of extraction key frames or not. The experiments will focus on the color and the motion because they are the main visual features widely used to detect and extract key frames. The specific hypothesis "motion feature with self-adaptive threshold will increase the accuracy of extracting key frames more than color feature". The independent variables are the algorithms that being used to extract key frames, they are absolute color histogram difference [9], the grey level consecutive frame differences [10], RGB color space frame blocks differential accumulation [11], and motion on accumulative optical flow with self-adaptive threshold (AOF_ST).

The dependent variable is indicator of extraction key frames accuracy which based on the count number of key frames that were extracted. The possible confound was the motion based on optical flow with self-adaptive threshold could increase the key extraction accuracy more than the-state-of-the-art algorithms which based on color with local and/or global thresholds. Thus, to measure the accuracy, we implemented different algorithms based on these techniques.

Our experiments were performed on a laptop, Toshiba Satellite C850-B098 with CPU-Intel (R) core (TM) i3-2312M CPU @ 2.10 GHz, memory RAM-2 GB, system type 32-bit. Our system was programmed with Matlab2013a using Windows 7. Key frames were extracted using our AOF_ST and it was compared with the-state-of-the-art algorithms, the absolute color histogram difference [9], the grey level consecutive frame differences with threshold [10] and RGB color space frame blocks differential accumulation [11] respectively. Five videos from KTH as described in [9] were used as training set. All selected videos were represented as individual action in a single shot. Therefore, the shot detection techniques were not performed.

RGB color space frame blocks differential accumulation [11] has two thresholds as shown in Equations 6 and 7, the *mValue* is the mean differences of all blocks in each frame, *m* and *n* are the number of columns and rows respectively in the frame. Variable *a* range is [0,1] and variable *b* range is [-10,10]. We implemented RGB color space frame blocks differential accumulation [11] with different chosen values for threshold variables *a* and *b* to determine the most suitable thresholds values to have an accurate results and it had the best *CR* when a = 1 and b = - 6.

$$Threshold1 = a \times mValue \qquad (6)$$

$$Threshold2 = b + (m \times n) \times \propto \ where \ \propto = 0.6 \qquad (7)$$

The compactness measure of shots contents due to the extracted key frames was computed using the compression ratio (*CR*) to evaluate the performance of key frame extraction algorithms. The higher value of *CR* of an algorithm indicates that the algorithm is good [9]. The *CR* was calculated using the Equation 8 from [9].

$$CR = \frac{\#\, Video\ frames}{\#\, Extracted\ key\ frames} \qquad (8)$$

The *CR* results of the absolute color histogram difference were described in [9], while the *CR* results of our proposed algorithm, the grey level consecutive frame differences with threshold [10], and RGB color space frame blocks differential accumulation [11] were obtained by experiments. As shown in Table 3, our proposed algorithm had the best *CR* of extracting key frames for the five videos.

Table 3. *CR* for AOF_ST and the-state-of-the-art algorithms

Video	Frames count	Compression rate of the extracted key frames			
		Our alg.	Color his. diff.	Con. frame diff.	Frame blocks diff. (1,-6)
Person1_runing_d1	335	47.85	5.68	4.14	1.58
Person1_ runing_d2	365	33.18	8.90	2.92	1.79
Person1_ runing_d3	350	43.75	6.48	3.72	1.57
Person2_ runing_d1	314	31.4	6.83	2.51	1.35
Person2_ runing_d2	1492	124.34	7.54	1.79	1.75

To study key frame extraction under different complex environmental cases, a validation set with different 24 video shots were experimented. These were: moving objects with static camera, moving objects with a moving camera, dynamic texture with a static camera and not moving objects, dynamic texture with a static camera and moving objects, and dynamic texture with a shaky camera. The validation set videos being used are described in Table 4.

Table 4. Description of the validation set videos

Original video	Description	Faked video	Forgery Description
v1	Static camera with moving object	v2	Duplicate moving object then merge them
v3	Small shaky camera with moving objects within dynamic texture	v4	Remove static object within dynamic texture
v5	Small shaky camera with moving objects	v6	Objects motion interpolation
v7	Moving camera with moving objects	v8	Object motion interpolation
v9	Zoom out camera with dynamic texture	v10	Extended dynamic texture
v11	Small shaky camera with dynamic texture	v12	Extended dynamic texture
v13	Static camera with dynamic texture	v14	Added moving objects within dynamic texture
		v15	Extended dynamic texture
v16	Static camera with dynamic texture	v17	Extended dynamic texture
v18	Static camera with dynamic texture	v19	Removed static object within dynamic texture
v20	Static camera with moving objects within dynamic texture	v21	Added moving objects within dynamic texture
		v22	Added moving object within dynamic texture
v23	Small shaky camera with dynamic texture	v24	Extended dynamic texture and duplicated frames

Table 5 shows the statistical analysis of the validation set videos. The shot's size varied between 4.93 MB to 32.7 MB, and with a different duration length between 3 seconds to 17 seconds before pre-processing and after pre-processing between 3.07 seconds to 14.87 seconds with size (320 W × 320 H).

Table 5. Statistics of the validation set videos

	Frames count	Duration (Sec.)
Mean	241.17	8.0389
Std. Deviation	104.792	3.49300
Minimum	92	3.07
Maximum	446	14.87

To realize if there is any effect of choosing different visual features and techniques on the accuracy of key frame extraction, Table 6 shows the mean and standard deviation of implementing different algorithms based on different features on different videos conditions. From the statistical description in Table 6, we can conclude that AOF_ST had a compact count number of key frames compared to the-state-of-the-art algorithms (Mean 7.63 and Std. Dev. 2.28). The-state-of-the-art algorithms had "Mean" greater than 25 and Std. Dev. larger than 17.

Table 6. Descriptive statistics about the total count number of extracted key frames

	AOF_ST	Color his. diff.	Con. frame diff.	Frame blocks diff.		
				(0.05,-6)	(0.50,-6)	(1,-6)
Mean	7.63	25.46	111.08	193.58	193.58	180.88
Std. Deviation	2.281	17.983	125.301	106.223	106.223	113.084
Minimum	3	4	4	46	46	6
Maximum	10	60	443	444	444	441

We used the nonparametric methods (Binomial test) from the SPSS ver.20 statistical package to understand whether accuracy of generating a compact number of key frames differed based on features and techniques used. The statistical test conducted for our algorithm and the-state-of-the-art algorithms is necessary to confirm their accuracy. If gained results are near to the supposed value that we need to achieve, i.e. compact number of key frames which is equal or less than 10 key frames. We investigated whether the proportion of key frames extracted differs from 0.90 (our null hypothesis states that this proportion is 0.90 of the entire population). To define the dichotomous variable we used cut point equaling 10 to check if the algorithms generate a compact count number of key frames equal or less 10. Since our algorithm has 24 videos achieved the condition <=10 out of 24 observations, the observed proportion is (24 / 24 = 1.0). The p value denoted by Exact Sig.(1-tailed) of our algorithm is 0.08. If the proportion of key frames extraction is exactly 1.0 in the entire population, then there's only a 8 % chance of finding more than ten key frames in any extracted video. We often reject the null hypothesis if this chance is smaller than 5% (p < .05). While on the-state-of-the-art algorithms the maximum observed proportion does not increase than 0.2 and their observed proportions are smaller than the test proportion. Therefore, we accepted our algorithm results and ignore the-state-of-the-art algorithms result. As a result, the Binomial test indicated that in our algorithm the proportion of extracting key frames equal or less than ten of 1.0 was higher than the expected 0.90, p = 0.08 (1- tailed) as shown in Table 7.

Table 7. The binomial test result of extracting key frames counts number

Algorithm	Category	N	Observed Prop.	Test Prop.	Exact Sig. (1-tailed)
AOF_ST	<= 10	24	1.0	.90	.080
Color his. diff.	<= 10	5	.2	.90	.000[a]
Con. frame diff.	<= 10	2	.1	.90	.000[a]
Frame blocks diff. (0.05,-6)	<= 10	0	0	.90	.000[a]
Frame blocks diff. (0.50,-6)	<= 10	1	0	.90	.000[a]
Frame blocks diff. (1,-6)	<= 10	2	.1	.90	.000[a]

Alternative hypothesis states that the proportion of cases in the first group < 0.90

4. Conclusion

After analyzing the results using the Binomial test, we concluded that the accuracy of extracting key frames can be affected by visual features (color, motion). In general, motion features give more information about movements on video when compared to color features. As a result, motion features extracts a compact count number of key frames.

Our proposed algorithm, AOF_ST, which was developed based on motion feature, provided good and meaningful compact key frames extraction. This proposed algorithm is useful for video fingerprint generation, detecting video forgery system, video retrieval and searching system. The-state-of-the-art algorithms in the experiments, namely, the absolute color histogram difference [9], the grey level consecutive frame differences [10], RGB color space frame blocks differential accumulation [11], were not able to extract a compact count number of key frames in all condition cases which can affect the usability of their algorithms in different systems. Our proposed algorithm has a controlled number of key frames between three and ten.

Spatio-temporal changes provide significant key frames with accurate representation consequent to detect meaningful details about the salient events inside the video [26]. Our proposed algorithm, AOF_ST, focuses on estimating motion on spatial domain using optical flow, and for a temporal domain uses a fixed number of cumulative frames. Thus, our proposed algorithm has meaningful key frames, reduces the effects of videos environment and camera conditions and provides more accurate results. In comparison, the-state-of-the-art algorithms tested in our experiments were unable to describe the shots in an efficient manner due to the redundancy or very similar key frames. In this regard, we created a summation of key frames difference to show the redundancy of detecting moving objects within the key frames to give a better understanding of the meaningful key frames.

As shown in Figure 4 (video 1), the absolute color histogram difference [9] and the grey level consecutive frame differences [10] missed some salient events, while RGB color space frame blocks differential accumulation [11] extracted a set of different number of key frames based on the threshold, and missed more salient events than the absolute color histogram difference [9] and the grey level consecutive frame differences [10].

The accuracy of the key frame in Figure 4 (video 2) shows that our proposed algorithm AOF_ST is better than the-state-of-the-art algorithms in the experiments. The-state-of-the-art algorithms in the experiments had extracted redundant or very similar key frames. On the opposite, our proposed algorithm AOF_ST detected the important salient events in the two videos with compact number of key frames.

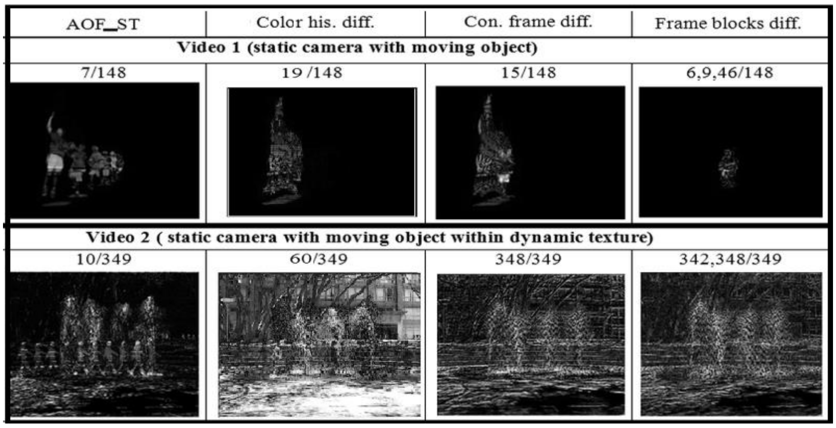

Figure 4. Meaningful key frames extraction between AOF_ST and the-state-of-the-art algorithms

In addition, it is hard to determine the threshold value for the RGB color space frame blocks differential accumulation [11] and several experiments need to be attempted to find the appropriate threshold value. For all video shots, our proposed algorithm in this paper was able to extract key frames automatically. The threshold in our algorithm was self-adaptive and that makes our algorithm suitable for full automatic application in future.

5. Conclusion and Future Work

We analyzed the influence of using several techniques based on different visual features (color, motion) on key frame extraction accuracy using TRIZ. Also, we devised an algorithm to increase the accuracy by generating meaningful compact key frames using accumulative optical flow with self-adaptive threshold inspired by the TRIZ inventive principles. In our experiment, the extracted key frames were shown to be able to represent the whole video and summarizes the important objects and salient events of the video. In addition, our proposed algorithm was found to be able to extract key frames in video with slow or fast moving objects and regions without prior knowledge of their shapes and sizes. On top of that, our proposed algorithm achieved better compression rate in KTH data set in comparison with the-state-of-the-art algorithms. In future work, we intend to utilise our proposed algorithm as a basis of the development of advanced video processing systems to perform video summarization and retrieval, and to increase the level of detection in the process of video forgery systems based on fingerprint.

References

[1] A Nasreen. Keyframe Extraction from Videos - A survey. *International Journal of Computer Science and Communication Networks*. 2013; 3(3): 194-8.

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

[3] MA Mizher, MC Ang. *Hybrid Video Moving Objects Detection System*. The Second Visual Informatics International Seminar 2014(VIIS'14), in conjunction with the 7th MYREN Conference; 25-27 Nov.; Kuala Lumpur, Malaysia. 2014: 46-52.

[4] C Sujatha, U Mudenagudi. *A study on Keyframe Extraction Methods for Video Summary*. International Conference on Computational Intelligence and Communication Networks (CICN); 2011 7th Oct.: IEEE: 73-7.

[5] D Patel, S Upadhyay. Optical Flow Measurement Using Lucas Kanade Method. *International Journal of Computer Applications*. 2013; 61(10).

[6] N Lu, J Wang, L Yang, QH Wu. Motion Detection Based On Accumulative Optical Flow and Double Background Filtering. *World Congress on Engineering (WCE)*; 2007 2-4 Jul.; London, U.K.: Citeseer: 602-7.

[7] JC Sosa, R Rodríguez, VHG Ortega, R Hernández. Real-time Optical-flow Computation for Motion Estimation under Varying Illumination Conditions. *International Journal of Reconfigurable and Embeded System (IJRES)*. 2012; 1(1): 25-36.

[8] I Chugh, R Gupta, R Kumar, P Sahay. *Techniques for key frame extraction: Shot segmentation and feature trajectory computation*. 6th International Conference - Cloud System and Big Data Engineering (Confluence); 2016 14-15 Jan: IEEE: 463-6.

[9] SCV, NK Narayanan. Key-frame Extraction by Analysis of Histograms of Video Frames Using Statistical Methods. *Procedia Computer Science*. 2015; 70: 36-40.

[10] SD Thepade, A Tonge. *An optimized Key Frame Extraction for Detection of near Duplicates in Content Based Video Retrieval*. International Conference onCommunications and Signal Processing (ICCSP); 2014: IEEE: 1087-91.

[11] C Cao, Z Chen, G Xie, S Lei. *Key Frame Extraction Based on Frame Blocks Differential Accumulation*. 24th Chinese Control and Decision Conference (CCDC); 2012: IEEE: 3621-5.

[12] MA Mizher, MC Ang, AA Mazhar, MA Mizher. A review of video falsifying techniques and video forgery detection techniques. *Int J Electronic security and digital forensics*. 2017; 9(3): 191-209.

[13] R Ahuja, SS Bedi. Video Watermarking Scheme Based on Candidates I-frames for Copyright Protection. *Indonesian Journal of Electrical Engineering and Computer Science*. 2017; 5(2): 391-400.

[14] Y Shi, H Yang, M Gong, X Liu, Y Xia. A Fast and Robust Key Frame Extraction Method for Video Copyright Protection. *Journal of Electrical and Computer Engineering*. 2017; 2017, Article ID 1231794: 1-7.

[15] R Hamza, K Muhammad, Z Lv, F Titouna. Secure video summarization framework for personalized wireless capsule endoscopy. *Pervasive and Mobile Computing, publisher: Elsevier* BV 2017.

[16] K Muhammad, M Sajjad, MY Lee, SW. Baik, Efficient visual attention driven framework for key frames extraction from hysteroscopy videos. *Biomedical Signal Processing and Control*. 2017; 33: 161-8.

[17] MC Ang, DT Pham, AJ Soroka, KW Ng. *PCB assembly optimisation using the Bees Algorithm enhanced with TRIZ operators*. 36th Annual Conference of the IEEE Industrial Electronics Society (IECON-2010); 2010 7-10 Nov, 2010; Phoenix, Arizona, USA.

[18] SA Ahmad, DT Pham, KW Ng, MC Ang. *TRIZ-inspired Asymmetrical Search Neighborhood in the Bees Algorithm.* The Asian Modelling Symposium (AMS2012): the 6th Asia International Conference on Mathematical Modelling and Computer Simulation; 2012 28 May 2012 & 31 May 2012; Kuala Lumpur, Malaysia & Bali, Indonesia: IEEE: 29-33.

[19] A Aghamohammadi, MC Ang, AS Prabuwono, M Mogharrebi, KW Ng. *Enhancing an Automated Inspection System on Printed Circuit Boards Using Affine-SIFT and TRIZ techniques*. MCAIT 2013 conference; 2013; Shah Alam, Selangor, Malaysia: Springer Communications in Computer and Information Science series (SCOPUS indexed).

[20] MC Ang, KW Ng, DT Pham, A Soroka. *Simulations of PCB Assembly Optimisation Based on the Bees Algorithm with TRIZ-inspired Operators*. 3rd International Visual Informatics Conference (IVIC 2013); 2013 13-15 November 2013; Equatorial Hotel Bangi, Selangor, Malaysia: Springer International Publishing, Switzerland, LNCS 8237, pp. 335-346. (SCOPUS indexed).

[21] MC Ang, KW Ng, SA Ahmad, ANA Wahab. Using TRIZ to generate ideas to solve the problem of the shortage of ICT workers. *Applied Mechanics and Materials* 2014; 564: 733-9.

[22] PS Bajwa, D Mahto, Concepts. Tools and Techniques of Problem Solving through TRIZ: A review. *International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)*. 2013; 2(7): 3061-73.

[23] K Sahu, S Verma. Key Frame Extraction from Video Sequence: A Survey. *International Research Journal of Engineering and Technology (IRJET)*. 2017; 4(5): 1346-50.

[24] H Liu, L Pan, W Meng. *Key frame Extraction From Online Video Based on Improved Frame Difference Optimization*. International Conference on Communication Technology (ICCT); 2012 9-11 Nov.; Chengdu, China: IEEE 940-4.

[25] J Majumdar, DKM, A Vijayendra. Design and Implementation of Video Shot Detection on Field Programmable Gate Arrays. *International Journal of Robotics and Automation (IJRA)*. 2013; 2(1): 17-25.

[26] J Luo, C Papin, K Costello. Towards extracting semantically meaningful key frames from personal video clips: from humans to computers. *IEEE Transactions on Circuits and Systems for Video Technology*. 2009; 19(2): 289-301.