How to cite this article:

Abduljabbar, M., Mizher, M. A., Ang, M. C., Abdullah, S. N. H. S., & Ng, K. W. (2019). An improved action key frames extraction algorithm for complex colour video shot summarization. *Journal of Information and Communication Technology*, *18*(2), 143-166.

AN IMPROVED ACTION KEY FRAMES EXTRACTION ALGORITHM FOR COMPLEX COLOUR VIDEO SHOT SUMMARIZATION

[1]Manar Abduljabbar Ahmad Mizher, [2]Ang Mei Choo, [3]Siti Norul Huda Sheikh Abdullah &[4]Kok Weng Ng

[1'2]Institute of Visual Informatics, Universiti Kebangsaan Malaysia, Malaysia

[3]Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Malaysia

[4]Faculty of Engineering, University of Nottingham Malaysia Campus, Malaysia

manar_mizher@yahoo.com; amc@ukm.edu.my; snhsabdullah@ukm.edu.my; kokweng.ng@nottingham.edu.my

ABSTRACT

Key frame extraction is one of the critical techniques in computer vision fields such as video search, video identification and video forgeries detection. The extracted key frames should be sufficient key frames that preserve main actions in a video with compact representation. The objective of this work is to improve our previous action key frames extraction algorithm (AKF) by adapting a threshold which selects action key frames as final key frames. The threshold adaptation was achieved by using the mean value, the standard deviation, and the L1-norm instead of the comparison of user summaries evaluation method to obtain a fully automatic video summarisation algorithm, and by eliminating the conditions in selecting the final key frames to reduce the complexity of the algorithm. We have validated our proposed Improved AKF on complex colour video shots instead of the simple grey level video shots. The Improved AKF algorithm was able to extract a compact number of action key frames by preventing redundant key

Received: 3 September 2018 **Accepted**: 11 December 2018 **Published:** 31 March 2019

frames, reduce processing complexity, and preserve sufficient information about the main actions in a video shot. We then evaluated the Improved AKF algorithm with the-state-of-the-art algorithms in terms of compression ratio using Paul videos and Shih-Tang dataset. The evaluation results showed that the Improved AKF algorithm achieved better compression ratio and retained sufficient information in the extracted action key frames under different testing video shots. Therefore, the improved AKF algorithm is a suitable technique for applications in computer vision fields such as passive object-based video authentication systems.

Keywords: Basic action, blocks differential, L1-norm, motion estimation, optical flow.

INTRODUCTION

The key frames (KFs) extraction algorithm should provide a compact video summarization with less processing time, and preserve the sufficient information in the video with simple implementation (Cao et al., 2012; Murtaza et al., 2018; Paul et al., 2017). From literature reviews, the key frame extraction algorithm starts by detecting the shot change to segment a video to several shots, then extracts the key frames from each shot (Truong & Venkatesh, 2007). The implicit computational techniques of the key frames extraction algorithms were divided into several categories namely sufficient content change, equal temporal variance, maximum frame coverage, minimum correlation between keyframes, sequence reconstruction error, clustering, curve simplification, and interesting events (Truong & Venkatesh, 2007).

Feature extraction is an essential step in several image processing systems (Sheena & Narayanan, 2015). Furthermore, the feature extraction has a significant role in recent video key frames extraction algorithms to select the key frames based on different features such as colour or motion. One of the key frame extraction algorithms is based on the colour feature which was developed recently to extract key frames inside a video shot by calculating the grey level histogram difference between two consecutive frames (Sheena & Narayanan, 2015). In this key frame selection algorithm (Sheena & Narayanan, 2015), a threshold was first computed based on the mean and the standard deviation of the absolute difference in the grey level histogram between every two consecutive frames from all video shots frames. The grey level histogram difference of the current frame was then compared with the computed threshold. If the difference was larger than a threshold, the current frame would be selected as a key frame. These steps were repeated until the end of the video shot to extract all key frames.

Another algorithm based on colour feature extracted the key frames by comparing the consecutive grey frame differences with a pre-determined threshold value (Thepade & Tonge, 2014). Their proposed algorithm reads each colour frame in a video shot and converts them into grey frame, and calculates their differences between the two consecutive frames. Then, the mean and standard deviation values are calculated simultaneously and its threshold value is set, equivalent to a multiplication of standard deviation and a constant number. If the grey difference between the current frame and the next frame is larger than the threshold, then the current frame will be saved as a key frame. Cao et al. (2012) proposed a key frame extraction algorithm based on the colour feature on frame blocks differential accumulation with two thresholds. In their algorithm, the first frame in a video shot was considered as the first reference frame. The remaining video shot frames were then partitioned into equal sized image blocks. The colour mean differences were computed in RGB colour space of the corresponding blocks in the reference frame and the current frame. Their proposed algorithm counted the blocks changing in the current frame in relation to the block changing in the reference frame. If the count number was greater than the global threshold, this means the current frame had more changes than the reference frame. Then, their proposed algorithm used the current frame as the key frame instead of the reference frame and similar steps were repeated until the last frame. Apart from its capability to identify the movements in high efficiency, it was also able to extract key frames with strong robustness in different types of video shots. Unfortunately, the weakness of their colour proposed algorithm is, it can only extract the key frames in video shots with different, vivid colour views or else they will extract a large number of the key frames. Regardless of the colour feature, motion able to extract the key frames keeps the important information about actions of the original video and provides a better compact video representation (Paul et al., 2017).

An algorithm used motion feature for key frame extraction was proposed, based on an accumulative optical flow with a self-adaptive threshold (AOF_ ST) as recommended in TRIZ inventive principles (Mizher et al., 2017b). The introduction of self-adaptive threshold namely AOF_ST can only support video shots duration for not more than 30 seconds. Later, Mizher et al. (2017a) overcame AOF_ST algorithm limitation by proposing an action key frames extraction algorithm based on *L1-norm* and accumulative optical flow for compact video shot summarisation.

The *Lp-norm* is a big family of norms. The least absolute errors (*L1-norm)* and the least squares (*L2-norm*) were commonly used to study the sparsity of motion by reflecting the complex spatial structure. Smooth nature of *L1-norm*

gives many solutions when compared to *L2-norm* which gives a single unique solution, and *L1-norm* is used when the difference between zero and nonzero elements is very important. *L1-norm* was proposed by Claerbout and Muir (1973) for data modelling. *L1-norm* was used by Schindler and Gool (2008) to read the sparsity of motion information to extract motion features. Xiao et al. (2015) selected the most correlated subset of motion from either several actions or from the same action based on *L1-norm* and *L2-norm* to convert the classic human motion denoising into *L1*-minimization framework(Xiao et al., 2015). The *L1-norm* and *L2-norm* in (Xiao et al., 2015) were robust against outliers and optimal with the Gaussian noise respectively. Xia et al. (2017) claimed that the *L1-norm* and *L2-norm* are not sparse as much as necessary to select one key frame from a sequence of very similar frames. The *L1-norm* was applied by (Mizher et al., 2017a) in calculating the sum of the absolute differences between each two optical flow frames to return motion sparsity information about these frames. Consequently, this action key frame extraction algorithm achieved a high compression ratio in the KTH dataset with action appearance accuracy between 91% and 100%.

A complex video is a video which has shots with a variety of challenging situations, such as multiple moving objects and moving camera which lead to variations in the background (Schuldt et al., 2004), or long uncut video containing noise (Murtaza et al., 2018). Several environments and camera conditions were experimented using key frames extraction algorithms. Grey colour with a static camera and static background video shots (Mizher et al., 2017a; Sheena & Narayanan, 2015), colour video shots with moving camera and dynamic background (Mizher et al., 2017b; Paul et al., 2017) Paul, Bhattacharya, and Gupta, 2017 extracted the key frames by formulating the dissimilarity between the video shot frames based on changes in object orientations and steerable filtering.

The process of extracting key frames from a video can affects the cost of storage and time. Key frames extracted should be compact and contain sufficient information about objects or events (Xia et al., 2017). Therefore, key frames extraction is a difficult task because the information should be selected to preserve important objects or events (Truong & Venkatesh, 2007). This task becomes more challenging when the key frames extraction needs to guarantee information integrity with fewer key frames (Xia et al., 2017). The selection of suitable key frames are essential and crucial for various fields such as a compact representation in video summarization, searching and retrieval (Mizher et al., 2017b), preserving sufficient information about objects or events effectively in object-based video forgery detection system (Yao et al., 2017; Mizher et al., 2017b). Therefore, a new key frames extraction algorithm

that is able to detect object motion sparsity and to extract the sufficient event key frames from complex video shots, is desirable and essential.

In this paper, an improved action key frames extraction algorithm is presented and experimented using complex colour video shots. The experiment results will be compared with the results of the previous algorithms (Cao et al., 2012; Mizher et al., 2017b; Paul et al., 2017). We have organized this paper into four sections beginning with an introduction, followed by Section 2, an improve action key frames extraction algorithm is presented and discussed which is based on *L1-norm* and optical flow estimation within the complex colour video shots; then, section 3 presents the experimental results and the analysis of the results, and finally, Section 4 concludes the work and suggestions for future research.

IMPROVED ACTION KEY FRAMES EXTRACTION ALGORITHM

The key frames extraction within a scene is an easier task than extracting them inside a shot. A scene has a transition between two sequential shots and different views from one shot to another shot, while the shot is a sequence of successive frames captured without interruption from the similar camera. Deriving an algorithm that can perform effective key frame extraction inside shots is a challenging task because video shot frames are attributed to many visual features such as motion and colour. Usually, the most important key frames which contain critical action of objects and extracted with a compact limited number will make them easier to use in different systems such as video retrieval and video fingerprint. Schindler and Gool (2008) defined basic action as a unit in which human behaviour shall be classified. Therefore, we attempt to improve the action key frames extraction algorithm (AKF) (Mizher et al., 2017a) which mainly focuses on key frames extraction under KTH dataset. Since KTH dataset was captured using a static camera and static background, and it has video shots in grey level with one basic action in each video shot, we focused on making improvement on AKF algorithm that is suitable for colour video shots with more actions such as one or more moving objects in video shot consisting of dynamic background or moving camera. Eventually, our Improved AKF algorithm was able to fit video shots with several environmental conditions by extracting key frames in long duration video shots in comparison to AKF algorithm.

The improvement in AKF algorithm was achieved by improving step-4 and eliminating step-5 until step-7 which is shown in Figure 1. Figure 1 illustrates

step-1 until step-7 of the action key frames extraction algorithm (Mizher et al., 2017a) to select the initial key frames based on *L1-norm* and optical flow with window size W. Hence, we improve step-4 in AKF algorithm (Mizher et al., 2017a) by picking up the action frames (*AFs*) as final key frames with adaptation of the threshold based on Equations (1), (2) and (3). The AKF algorithm (Mizher et al., 2017a) has used the threshold value in step-4 with *L1-norm* equals to a fixed value of 4600 based on the comparison of user summaries (CUS) evaluation method in selecting *AFs*. In order to evaluate the performance of the Improved AKF algorithm, compactness measure (CR) and image fidelity measure (IFM) is used. The compactness measure was calculated based on compression ratio (CR) and the image fidelity measure (IFM) was calculated based on the mean square error (MSE) between each two consecutive key frames to evaluate the performance of our key frame extraction algorithm. Comparisons between the AOF_ST algorithm, the AKF algorithm and the Improved AKF algorithm are shown in Table 1. The flow chart of the AOF_ST algorithm is shown in Figure 2.

Let us assume that *Threshold* in Equation (1) is a numeric value calculated by finding the absolute subtraction between A and B from a variable *Var*. In Equation (2), A is the mean value of all *Absdiff* images added to the standard deviation of all *Absdiff* images, B is the *L1-norm* for current *Absdiff* multiplied by a constant value $\propto$, the *Absdiff* is an absolute image difference deriving from subtracting the last optical flow frame in the previous window *W*, and the first optical flow frame from the current window *W*.

$$Threshold = |A - B| - Var, \quad (1)$$

$$\text{where } A = \sum_{j}^{n} \mu_i + \sum_{j}^{n} \sigma_i , \quad (2)$$

$$B = \propto \times L1\text{-}norm_i \quad (3)$$

Where μ and σ are the mean and standard deviation values from the range of *i* and *n*. The *i* is the current *Absdiff* image and the *n* is the last *Absdiff* image in the video shot respectively, and $\propto$=0.9 and *Var* is a variable in the range [0, 13]. *Var* range was used to balance the *Threshold* value and was chosen based on several experiments. When *Var* equals 0 or greater than 12 for each video shot in our dataset, the proposed algorithm will lose the effect of the *Threshold*. If the *Threshold* value is greater than 0, the current frame will be saved as an action key frame (*AKFs*) as shown in Figure 3. As shown in Figure 2 and Figure 3, the complexity of the Improved AKF was reduced when

compared to the AOF_ST and the AKF due to the elimination of conditions in selecting the final key frames and improving the *Threshold* selection through the Equations (1), (2) and (3).

Input: Video *V*, with a number of frames *n*, Window *W*, i current index, i-1 previous index, α and β user-defined numbers
Output: Set of keyframes *K* for input video *V*

***K* = AKF algorithm (*V*)**

{ Step 1: Read *W* consecutive frames from *V*
Estimate objects' velocities *Of* from each frame
Create *W* frames if frame velocities Not (Zero)
Step 2: Generate absolute difference image *AbsDiff* from *Ofi* and Of_{i-1},
where Of_i is the first frame in the current window,
Of_{i-1} is the last frame in the previous window
Step 3: Calculate *L1-norm* = Max(Sum (*AbsDiff* images))
Step 4: Action Frames *Af* = Threshold (*L1-norm* selected frames)
Step 5: Calculate *Af differences* = $Af_i - Af_{i-1}$
where Af_i is the current action frame,
Af_{i-1} is the previous action frame
Step 6: Candidate keyframes *CKF* = Sum (accumulate β of (*Af*)),
discard *Af* with none or tiny motion information
Step 7: For each *CKFs*
{ if *CKFs*> = α and < = β
K = *CKFs*
}
{ if CKFs has Extra keys < = β
K = Max β of (Calculate difference frames ($CKF_i - CKF_{i-1}$))
}
{ if *CKFs* has Extra keys > β
Binary *CKFs* = Convert to binary image (*CKFs*)
K = Max β of (Calculate the difference words (Binary CKF_i - Binary CKF_{i-1}))
}
}

Figure 1. The pseudo code of the AKF algorithm steps (Mizher et al., 2017a).

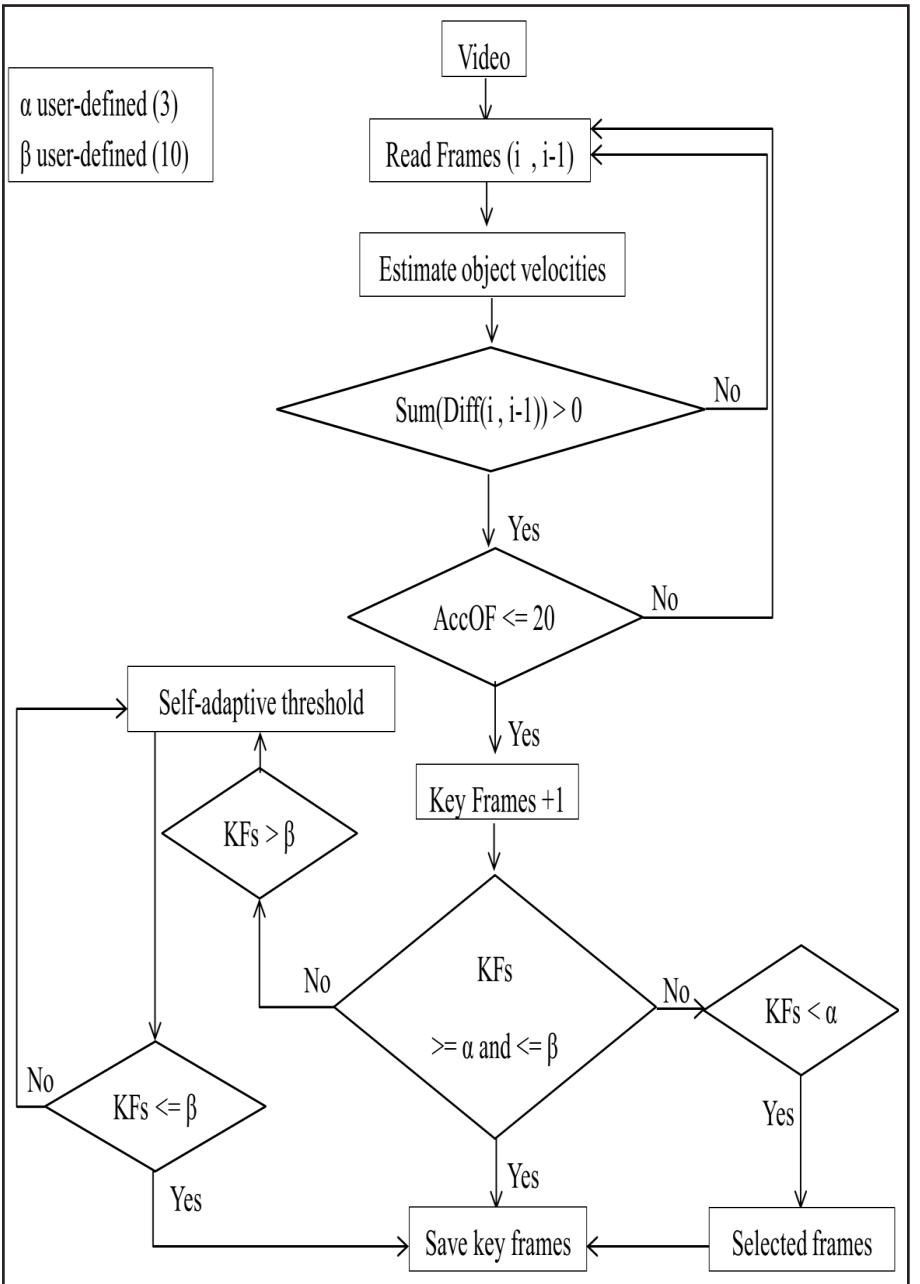

Figure 2. The flow chart of the AOF_ST algorithm (Mizher et al., 2017b).

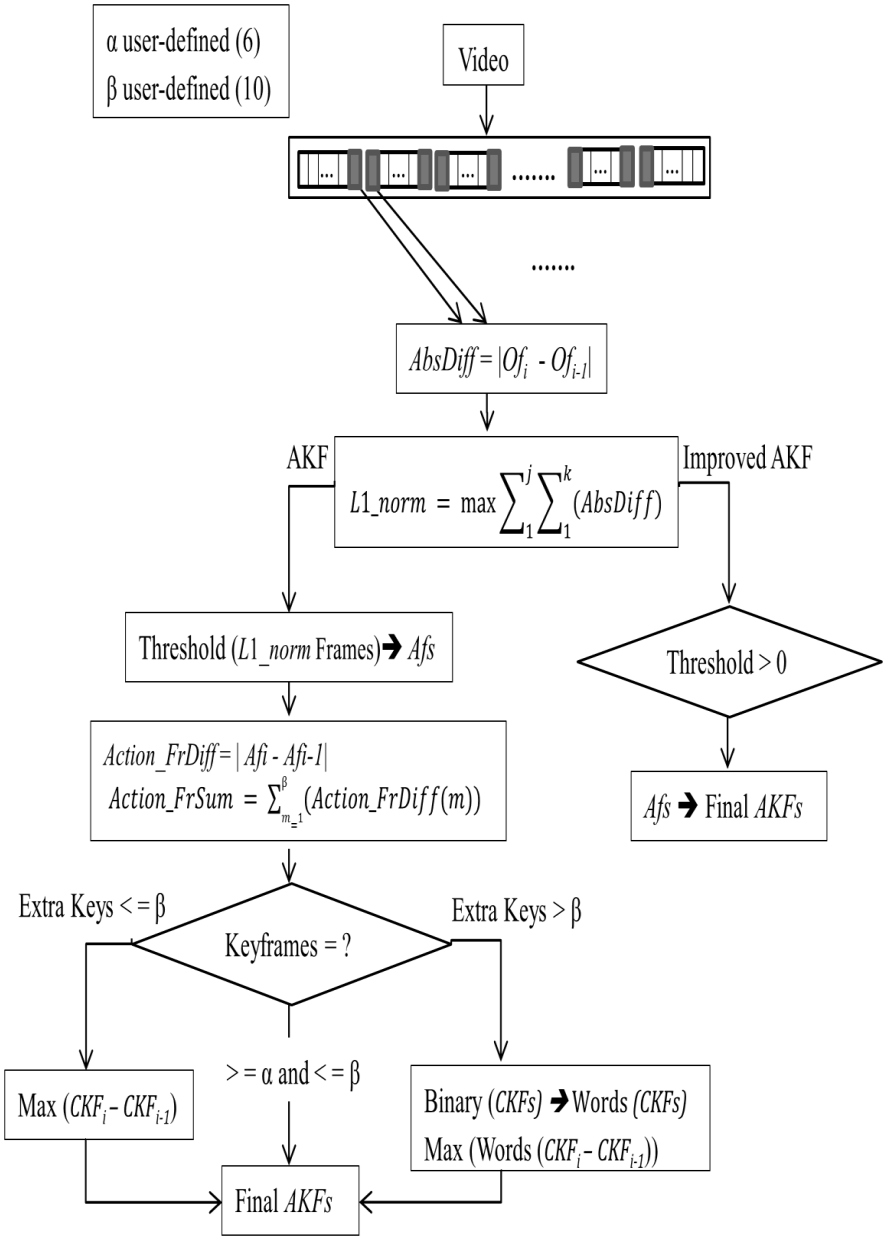

Figure 3. Comparison between the flow charts of AKF (Mizher et al., 2017a) and Improved AKF.

Table 1

Comparisons Between AOF_ST, AKF and Improved AKF Algorithms Using Figure 1 Steps.

Category of Comparisons	**AOF_ST (Mizher et al., 2017c)**	**AKF (Mizher et al., 2017a)**	**Improved AKF**
Video shot type	Complex true colour RGB	Simple grey scale	Complex true colour RGB
Application on forgery	Video shots with object-based forgery	None	Video shots with object-based forgery
Background condition	Static / Dynamic	Static	Static / Dynamic
Camera condition	Static /Moving	Static	Static /Moving
Dataset	KTH dataset (Schuldt et al., 2004) as 5 testing shots (Sheena & Narayanan, 2015). 24 validation shots from Shih-Tang dataset.	KTH dataset (Schuldt et al., 2004) as 5 testing shots (Sheena & Narayanan, 2015), 300 validation shots (jogging, running, and walking)	6 shots from Paul et al. (2017) testing video shots. 48 video shots from Shih-Tang dataset
Function	Manually	*L1-norm* (Matlab built-in)	*L1-norm* (Matlab built-in)
Threshold calculation	First diamond in Figure 2 (Summation of absolute difference function)	Step-4 in Figure 1 (using CUS method)	Equation (1), (2), (3)
Threshold condition	Greater than zero	Equal 4600	Greater than zero
Step-1 until Step-4	Used	Used	Used
Step-5 until Step-7	Not used	Used	Not used

EXPERIMENTS AND RESULT ANALYSIS

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. Schindler and Gool (2008) claimed that from one to seven frames are sufficient to recognise a basic action from a very short video shot. Unlike above, (Murtaza et al., 2018) claimed that the video content could adaptively determine the suitable number of frames, and in a typical video, requires between ten to twenty five frames. We chose ten for window size W similar to Paul et al. (2017) algorithm because it reflects the texture and motion features within video shots effectively. And selected the first frame as the first key frames following Paul et al. (2017) algorithm. Our proposed algorithm was implemented under two datasets which were the Paul et al. (2017) testing video shots, and Shih-Tang dataset.

Xia et al. (2017) claimed that the goal of extracting the key frames is to represent the overall motion by finding the most representative frames from a given motion sequence. It is unfortunate that the definition of what should be detected and extracted for video summarization is not a clear task. This occurs in view of the fact that every key frame extraction algorithm or video skims algorithm has its own evaluation methodology due to the absence in finding a stable evaluation framework. Therefore, Truong and Venkatesh (2007) have grouped the existing key frames extraction algorithms evaluation methods into three groups. These groups are based on result description, the objective metrics, or the user studies which require independent users to evaluate the quality of video summary or extracted key frames. Ideally, we should use the same datasets and the same metrics to compare our proposed algorithm with the state-of-the-art algorithms. However, most of the previous algorithms that used Shih-Tang dataset have extracted key frames manually and this made direct comparisons almost impossible. Descriptive tracking of the changes in the visual contents of the extracted key frames was used in (Paul et al., 2017) to evaluate the performance of the proposed algorithm. For this reason, we have used the user studies evaluation method based on descriptive tracking of the changes in the target objects of the extracted key frames to verify our key frames extraction algorithm.

Paul et al. (2017) mainly designed an algorithm to extract key frames within video shots for video colourization. Meanwhile, the less attention in literature researches focused primarily on extracting key frames within video shots. Therefore, we implemented our Improved AKF algorithm and the AOF_ST algorithm (Mizher et al., 2017b) under Paul et al. (2017) testing video shots.

Then we compared the experimented results with Paul et al. (2017) results. The tested video shots are Akiyo, Coastguard, Cartoon, Flower, Skiing, and Soccer. The testing video shots have a variety of different aspects as the colour space varied from grey to colour shots, camera conditions as static or moving, environmental conditions as static or dynamic background, and the object motion has either a slow or a rapid change. These variations make the decision of choosing the best threshold can extract the compact sufficient key frames, a challenging task.

The effect of changing *Var* value in the *Threshold* in Equation (1) was experimented with the range [0, 13] as shown in Figure 4. On the other hand, as the *Var* equals to a specific value for each video shot, the algorithm will lose the effect of *Threshold* (called this value the reflection value). Figure 4 illustrates the reflection value for a video shot with a static camera and object with small motion such as Akiyo equals five, in a video shot with low camera movement and low object motion such as Skiing and Cartoon is between six and seven, in video shots with moving camera with static background such as Soccer and Flower equals to 9.5, in video shot with moving camera and dynamic background such as Coastguard equals to 12.5.

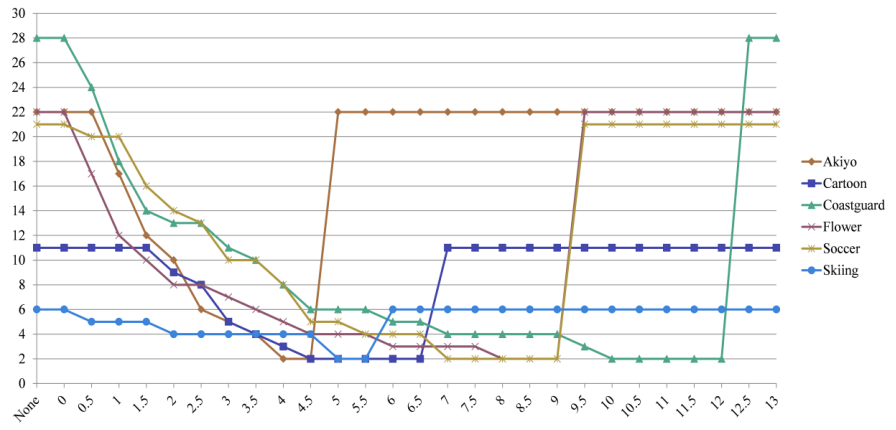

Figure 4. The effects of changing Var value of the Threshold.

As a result, we concluded that when *Var* values equal zero or equal the reflection value for each video shot the *Threshold* has no effect in the algorithm and choosing the best *Var* value is mainly depending on the video shot conditions. For the category of Improved AKF, the best *Var* value (Equation 1) for all video shots are chosen based on the two criteria; the compact number of the

extracted key frames and achievement of sufficient key frames by protecting the salient actions in the video shots. The compactness measure of video 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 performs better (Sheena & Narayanan, 2015). The CR was calculated using the Equation (4) from (Sheena & Narayanan, 2015).

$$CR = \frac{\#\,Video\ frames}{\#\,Extracted\ key\ frames} \tag{4}$$

There is often a trade-off between CR values and extracting sufficient key frames which is able to preserve the global content of a video (Sheena & Narayanan, 2015); therefore these two criteria will be taken into consideration in our experiments. Table 2 shows the effects on the number of the extracted key frames when chosing a range between [0, 2] for the *Var* values in the *Threshold* in Equation (1). The trend shows that as *Var* value equals to one or less, the *Threshold* will extract more similar key frames. These key frames will reduce the compactness ratio. On the other hand, if *Var* equals to two or more, the *Threshold* would miss some important information in the last video shot due to the low number of the key frames extraction as shown in Figure 5. On top of that, missing essential information would also disturb sufficient key frame achievement. As a result, *Var* value equals to 1.5 is the best value as it could extract a compact number of the key frames by protecting sufficient information within the video shots.

Table 2

Total extracted key frames with the chosen Var values on the Threshold.

Video shot	Var values				
	0	0.5	1	1.5	2
Akiyo	22	22	17	12	10
Cartoon	11	11	11	11	9
Coastguard	28	24	18	14	13
Flower	22	17	12	10	8
Soccer	21	20	20	16	14
Skiing	6	5	5	5	4

Figure 5. Key frames extracted from the Skiing when a) Var =0, b) Var =1.5, c) Var =2.

A comparison between Improved AKF algorithm and the state-of-the-art-algorithm is shown in Table 3. The RGB colour space frame blocks differential accumulation with two thresholds (Cao et al., 2012), changes in the object orientations algorithm (Paul et al., 2017), and accumulative optical flow with a self-adaptive threshold algorithm (AOF_ST) (Mizher et al., 2017b) respectively. Zheng et al. (2015) claimed using motion features can efficiently extract accurate visual differences of the sequential frames comparing with using colour features. Therefore, Cao et al. (2012) algorithm has the lowest CR results under the testing video shots because it is based on extracting the colour feature rather than motion feature. The Improved AKF algorithm has a moderate CR compared to changes in the object orientations algorithm (Paul et al., 2017), and accumulative optical flow with a self-adaptive threshold algorithm (AOF_ST) (Mizher et al., 2017b) as shown in Table 3. Although the Improved AKF algorithm did not achieve higher CR (Equation (4)) for all the videos when compared to AOF_ST, and it had less CR for three videos (Akiyo, Cartoon, and Coastguard videos) when compared to Paul algorithm, the Improved AKF achieved the best in terms of sufficient key frames that protects video shots information as shown in Figure 6 (when compared to AOF_ST and Paul algorithms).

Table 3

CR for the Improved AKF Algorithm and the-state-of-the-art Algorithms Comparing under Paul et al. (2017) testing video shots.

Video shot	All Frames	Cao	Improved AKF	Paul	AOF_ST
Akiyo	242	1.06	20.17	121.00	80.67
Cartoon	92	1.12	8.36	23.00	23.00
Coastguard	300	1.01	21.43	50.00	75.00
Flower	250	1.07	25.00	5.68	83.33
Soccer	240	1.05	15.00	1.61	24.00
Skiing	59	1.07	11.80	4.54	19.67
Total	1183				

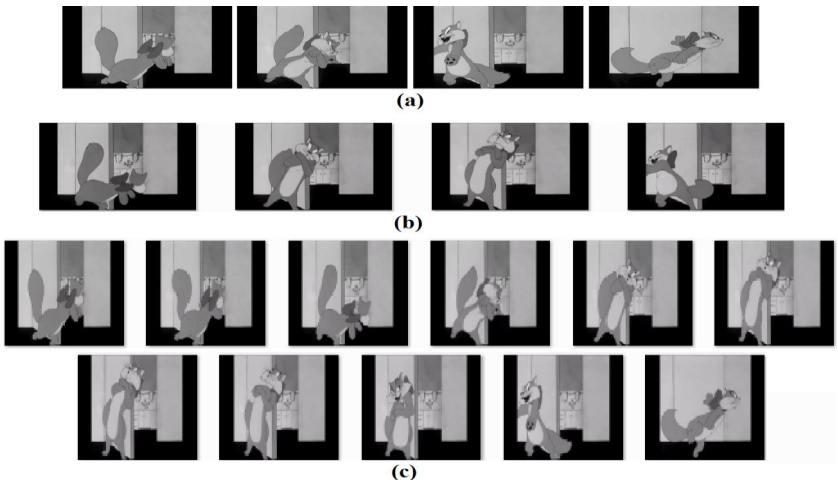

Figure 6. Key frame extracted from the Cartoon a) Paul algorithm (Paul et al., 2017) b) AOF_ST (Mizher et al., 2017b) c) Improved AKF.

Shih-Tang Dataset

The Improved AKF algorithm was further implemented under Shih-Tang dataset (Shih et al., 2008; Shih et al., 2011; Shih et al., 2010; Tang & Shih, 2008). It contained 50 video shots with 10570 frames. According to the author's information, Shih-Tang dataset is the largest dataset to handle various types of the object-based video forgery as shown in Table 4. The

variety occurred due to different techniques to forge the objects in motion interpolation mechanism based on reference stick figures (Shih et al., 2011), patch referencing and frame blending (Shih et al., 2010), textural synthesis and pixel interpolation (Tang & Shih, 2008), and modified mean shift mechanism (Shih et al., 2008). The video shots are in true RGB colour divided into three different categories: static camera with a static background, moving camera with a static background, and static camera with a moving background. All the video shots were in .avi and .wmv format video types, have moving objects in original video shots and in forged video shots, frame rate equal to 30 frames per second, with resolution equal to 320 x 320 pixels per frame, and a duration length of not more than one minute.

Table 4

Description of the Total Frames in Shih-Tang Dataset

Video shots	All Frames	Max Frames	Min Frames
50	10570	1151	84

In the experiments, we used only 48 video shots and eliminated the video shots with a duration of more than 30 seconds. This is because AOF_ST is designed for video shots not longer than 30 seconds. Table 5 gives the CR value of the experimented algorithms under Shih-Tang dataset. As shown in Table 5, the Improved AKF algorithm has higher CR (Equation (4)) of extracting key frames when compared with RGB colour space frame blocks differential accumulation with two thresholds (Cao et al., 2012).

Table 5

CR for the Improved AKF and the-state-of-the-art Algorithms under Shih-Tang Dataset

Algorithm	Extracted Key frames				CR	Time/Sec.
	Sum	Avg.	Max.	Min.		
RGB block diff.	6147	128.06	347	3	3.45	7.26
AOF_ST	319	6.65	10	3	27.35	6.48
Improved AKF	548	11.42	29	3	18.18	7.52

Although the Improved AKF algorithm CR is less than the accumulative optical flow with a self-adaptive threshold algorithm (AOF_ST) (Mizher et al., 2017b). As shown in Figure 7, Figure 8, and Figure 9, the AOF_ST algorithm

extracted a compact number of the key frames compaed to the Improved AKF algorithm. Nevertheless, the Improved AKF algorithm extracted sufficient information (jumping man) from the video shot while the AOF_ST algorithm missed this information in the 44th frame. In object-based video forgery systems, it is essential to protect all details related to objects.

Figure 7. Key frames extracted by AOF_ST from video19.avi under Shih-Tang dataset.

Figure 8. Key frames extracted by Improved AKF from video19.avi under Shih-Tang dataset.

Figure 9. The third key frame extracted by Improved AKF has sufficient information was missed by AOF_ST from video19.avi under Shih-Tang dataset.

As shown in Figure 10, AOF_ST gives an incomplete story of the 35.avi video shot, while the Improved AKF algorithm presents the visual understanding of the same video shot as shown in Figure 11. Image fidelity measure (IFM) was implemented by calculating the mean square error (MSE) between each two consecutive key frames as shown in Equation (5). MSE is the *L2-norm* of the difference between the source and the target and gives the degree of similarity (Wang & Bovik, 2009).

$$MSE = \frac{1}{N} \sum_{i=}^{N} (X_i - X_{i+1})^2 \, , \tag{5}$$

where X is a key frame, i is the current key frame, and N is the total number of the key frames in a video shot. If the mean square error equals to a zero, that means the two consecutive key frames are highly correlated and they are similar. As shown in Table 6, the maximum and minimum IFM between the key fames in video shot 34.avi are 0.8815 and 0.8237 respectively after using the Improved AKF algorithm.

Figure 10. The key frames extracted by AOF_ST from video 35.avi under Shih-Tang dataset.

Figure 11. The key frames extracted by Improved AKF from video 35.avi under Shih-Tang dataset.

(continued)


KeyFrame7_Frame77.png
KeyFrame8_Frame88.png
KeyFrame9_Frame99.png
KeyFrame10_Frame110.png
KeyFrame11_Frame121.png
KeyFrame12_Frame132.png
KeyFrame13_Frame143.png
KeyFrame14_Frame154.png
KeyFrame15_Frame165.png
KeyFrame16_Frame176.png
KeyFrame17_Frame198.png
KeyFrame18_Frame209.png
KeyFrame19_Frame220.png
KeyFrame20_Frame231.png
KeyFrame21_Frame242.png

(continued)

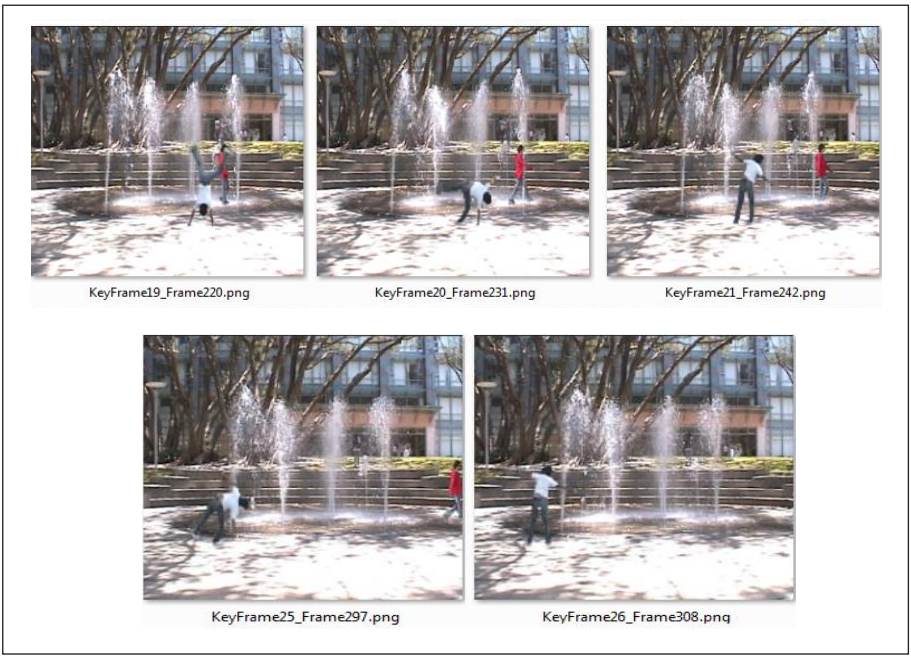

Table 6

Key Frames Fidelity measure using MSE in Video 35.avi using Improved AKF Algorithm

Total frames	Total key frames	Max. MSE	Min. MSE
349	26	0.88	0.82

Table 7 shows the comparisons between the AOF_ST, AKF and Improved AKF algorithms advantages, disadvantages and limitations.

As shown in the results, the Improved AKF algorithm, has a higher CR compared to Cao et al. (2012) and a good CR compared to Paul et al. (2017). Although the Improved AKF algorithm has a slightly higher running time with an average of 0.26 seconds when compared to Cao et al. (2012) and 1.04 seconds with AOF_ST, it is a better technique for the passive object-based video forgery detection system. This is because the Improved AKF algorithm prevents redundant key frames and preserve more sufficient information when compared to the AOF_ST algorithm (Mizher et al., 2017b) and changes in the object orientations algorithm (Paul et al., 2017).

Table 7

Comparisons between AOF_ST, AKF and Improved AKF Characteristics

Category of Comparisons	AOF_ST (Mizher et al., 2017c)	AKF (Mizher et al., 2017a)	Improved AKF
Advantages	High CR, suitable for complex videos with moving camera and multiple moving objects.	High CR, high action appearance accuracy of 91% to 100%, suitable for simple videos.	Good CR, suitable for complex videos with moving camera and multiple moving objects, prevents redundant key frames and preserve more sufficient information, and simple implementation.
Disadvantages	Loss some sufficient information.	Affected by objects shadow and shaky camera.	Need more execution time.
Limitations	Support video shot duration not more than 30 seconds.	Suitable for static camera with one moving objects.	Experimented under video shots not more than one minute.

CONCLUSION AND FUTURE WORK

The extracted key frames can be used to represent the video as a whole and summarized the important objects or actions of the video. The optimal number of extracted key frames is mainly depending on video complexity, such as camera motion, shot visual contents, or the dynamicity of foreground or background. The Improved AKF outperformed the-state-of-the-art algorithms whereby it detects the important actions in the video shots with a relatively good compression ratio of key frames. For all video shots, the Improved AKF algorithm can extract action key frames automatically which makes it suitable for full automatic application for forgery video detection, video retrieval and searching system. As further research,, we are now working on feature extraction from the extracted key frames as a resultant of the Improved AKF algorithm. It yields to build a full automatic passive object-based video forgery detection system.

ACKNOWLEDGEMENT

The authors would like to thank the Ministry of Higher Education, Malaysia and Universiti Kebangsaan Malaysia for supporting this work through the Fundamental Research Grant Scheme (FRGS/1/2018/TK03/UKM/02/6), the Arus Perdana Grant (AP 2017-005/2) and the Prototype Research Grant Scheme (PRGS/1/2016/ICT02/UKM/02/1). The authors also would like to thank Prof. T. K. Shih for sharing us his videos, and Prof. Sergio A Velastin for his useful comments and suggestions.

REFERENCES

Cao, C., Chen, Z., Xie, G., & Lei, S. (2012). *Key frame extraction based on frame blocks differential accumulation.* 24th Chinese Control and Decision Conference, 3621-3625. doi: 10.1109/CCDC.2012.6243092.

Claerbout, J. F., & Muir, F. (1973). Robust modeling with erratic data. *Geophysics, 38*(5), 826-844. doi: https://doi.org/10.1190/1.1440378.

Mizher, M. A., Ang, M. C., Abdullah, S. N. H. S., & Ng, K. W. (2017a). Action key frames extraction using l1-norm and accumulative optical flow for compact video shot summarisation. In Badioze Zaman H. et al. (Eds.), *Advances in Visual Informatics*. Lecture Notes in Computer Science, Bangi, Malaysia (pp. 364-375). doi: https://doi.org/10.1007/978-3-319-70010-6_34.

Mizher, M. A., Ang, M. C., & Mazhar, A. A. J. (2017b). A meaningful compact key frames extraction in complex video shots. *Indonesian Journal of Electrical Engineering and Computer Science, 7*(3), 818-829. doi: 10.11591/ijeecs.v7.i3.pp%25p.

Murtaza, F., Yousaf, M. H., & Velastin, S. A. (2018). PMHI: Proposals from motion history images for temporal segmentation of long uncut videos. *IEEE Signal Processing Letters, 25*(2), 179-183. doi: 10.1109/LSP.2017.2778190.

Paul, S., Bhattacharya, S., & Gupta, S. (2017). *Selection of keyframes for video colourization using steerable filtering*. In S. Sādhanā, India *42*(10), (pp. 1685-1692): Springer. doi.org/10.1007/s12046-017-0720-y.

Schindler, K., & Gool, L. V. (2008). *Action snippets: How many frames does human action recognition require?* Conference on Computer Vision and Pattern Recognition, 1-8. doi: 10.1109/CVPR.2008.4587730.

Schuldt, C., Laptev, I., & Caputo, B. (2004). *Recognizing human actions: A local SVM approach.* 17th International Conference on Pattern Recognition, 32-36. doi: 10.1109/ICPR.2004.1334462.

Sheena, C. V., & Narayanan, N. K. (2015). Key-frame extraction by analysis of histograms of video frames using statistical methods. *Procedia Computer Science, 70*, 36-40. doi: 10.1016/j.procs.2015.10.021.

Shih, S. K., Tan, N. C., Tsai, J. C., & Zhong, H.-Y. (2008). *Video falsifying by motion interpolation and inpainting.* IEEE Conference on Computer Vision and Pattern Recognition, 1-8. doi: 10.1109/CVPR.2008.4587701.

Shih, T. K., Tang, N. C., Tsai, J. C., & Hwang, J.-N. (2011). Video motion interpolation for special effect applications. *IEEE Transactions on System Man and Cybernatics Part C* (*Applications and Reviews*), *41*(5), 720-732. doi: 10.1109/TSMCC.2010.2077674.

Shih, T. K., Tsai, J. C., & Li, K.-C. (2010). *Video narrative authoring with motion inpainting.* 1st ACM international workshop on Multimodal pervasive video analysis, 11-16. doi: 10.1145/1878039.1878043.

Tang, N. C., & Shih, T. K. (2008). *Digital inpainting and video falsifying.* International Symposium on Information Technology, 1-8. doi: 10.1109/ITSIM.2008.4631531.

Thepade, S. D., & Tonge, A. A. (2014). *An optimized key frame extraction for detection of near duplicates in content based video retrieval.* International Conference on Communications and Signal Processing, 1087-1091. doi: 10.1109/ICCSP.2014.6950015.

Truong, B. T., & Venkatesh, S. (2007). Video abstraction: A systematic review and classification. *ACM Transactions on Multimedia Computing, Communications, and Applications, 3*(1), Article (3). doi: 10.1145/1198302.1198305.

Wang, Z., & Bovik, A. C. (2009). Mean squared error: Love it or leave it? A new look at signal fidelity measures. *IEEE Signal Processing Magazine, 26*(1), 98-117. doi: 10.1109/MSP.2008.930649.

Xia, G., Sun, H., Niu, X., Zhang, G., & Feng, L. (2017). Keyframe extraction for human motion capture data based on joint kernel sparse representation. *IEEE Transactions on Industrial Electronics, 64*(2), 1589-1599. doi: 10.1109/TIE.2016.2610946.

Xiao, J., Feng, Y., Ji, M., Yang, X., Zhang, J. J., & Zhuang, Y. (2015). Sparse motion bases selection for human motion denoising. *Signal Processing, 110*, 108-122. https://doi.org/10.1016/j.sigpro.2014.08.017.

Yao, Y., Shi, Y., Weng, S., & Guan, B. (2017). Deep learning for detection of object-based forgery in advanced video. *Symmetry 2018, Special Issue* (*Emerging Data Hiding Systems in Image Communications*), *10(1)*(3). doi: 10.3390/sym10010003.

Zheng, R., Yao, C., Jin, H., Zhu, L., Zhang, Q., & Deng, W. (2015). Parallel key frame extraction for surveillance video service in a smart city. *PLOS ONE, 10*(8), e0135694. doi: 10.1371/journal.pone.0135694.