AIMS Mathematical Biosciences and Engineering

MBE, 18(6): 9294–9311.
DOI: 10.3934/mbe.2021457
Received: 12 August 2021
Accepted: 12 October 2021
Published: 27 October 2021

http://www.aimspress.com/journal/MBE

Research article

Feature fusion and clustering for key frame extraction

Yunyun Sun [1], Peng Li [2,3,*], Zhaohui Jiang [4] and Sujun Hu [2]

[1] School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing, 210023, China
[2] School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing, 210023, China
[3] Institute of Network Security and Trusted Computing, Nanjing, 210023, China
[4] School of Information and Computer Science, Anhui Agricultural University, Hefei, 230036, China

* **Correspondence:** Email: lipeng@njupt.edu.cn.

Abstract: Numerous limitations of Shot-based and Content-based key-frame extraction approaches have encouraged the development of Cluster-based algorithms. This paper proposes an Optimal Threshold and Maximum Weight (OTMW) clustering approach that allows accurate and automatic extraction of video summarization. Firstly, the video content is analyzed using the image color, texture and information complexity, and video feature dataset is constructed. Then a Golden Section method is proposed to determine the threshold function optimal solution. The initial cluster center and the cluster number *k* are automatically obtained by employing the improved clustering algorithm. k-clusters video frames are produced with the help of K-MEANS algorithm. The representative frame of each cluster is extracted using the Maximum Weight method and an accurate video summarization is obtained. The proposed approach is tested on 16 multi-type videos, and the obtained key-frame quality evaluation index, and the average of Fidelity and Ratio are 96.11925 and 97.128, respectively. Fortunately, the key-frames extracted by the proposed approach are consistent with artificial visual judgement. The performance of the proposed approach is compared with several state-of-the-art cluster-based algorithms, and the Fidelity are increased by 12.49721, 10.86455, 10.62984 and 10.4984375, respectively. In addition, the Ratio is increased by 1.958 on average with small fluctuations. The obtained experimental results demonstrate the advantage of the proposed solution over several related baselines on sixteen diverse datasets and validated that proposed approach can accurately extract video summarization from multi-type videos.

Keywords: Cluster; feature data; threshold; optimization; video summarization

1. Introduction

Video summarization, the task of concussing the original video to a summary, can fully catch the eye-catching video information. Since the 1990s, video summarization technology has gained considerable domestic and international attention. It has made significant contribution in quickly understanding the video information, which is an efficient tool for fast browsing and retrieval of videos [1].

Key-frame extraction is an indispensable assistant for static video summarization technology that characterizes the principal video contents with representative frames extraction in order to provide a convenient method to quickly and comprehensively grasp video information. The prevailing assumption is that the goal is to extract video summary accurately and automatically. Key frame extraction methods can be divided into three categories: shot based, content based and cluster based [2]. Currently, some shot based techniques are developed in the area of computer vision and image processing [3]. Huang C. extracts representative frames from each shot by computing the frame image difference in saliency and edge map features [4]. Mehmood I. analyzes the difference between frame images in a shot by modeling an auditory and perceptual attention feature [5]. Song G. H. computes the color difference in one shot by employing the average histogram method [6]. It is common for shot based methods to segment the original video into several shots at first. However, the shot segmentation process is computationally expensive. Content based methods can avoid this problem. Rachida H. proposes MSKVS, a content-based method, to measure the inter-frame distance by time and visual features. MSKVS guarantees superior performance over other content-based methods [7]. Gianluigi C. conducts experiment on six new and sport competition videos by employing his content-based method. Experimental results demonstrate that his method can effectively extract key frames [8]. Generally, these content-based methods analyze the video content by extracting color, texture or motion feature. A limiting factor of content-based methods is that the computational cost is incurred in the process of frame image features [9]. This limitation encourages the development of cluster-based methods. Cluster based techniques work by clustering together the similar frames and extracting one representative frame of each class. They avoid shot segmentation error of shot based methods, decrease inter-frame difference analysis frequency of content-based methods, and can be well-suited to key frame extraction in related fields [10–12].

Numerous limitations have been posed to encourage the development of cluster-based key-frame extraction algorithms [13]. The prevailing steps are cluster data, aggregate video frames into multiple clusters, and extract representative frame to compose a video summarization [14]. The cluster-based key-frame extraction not only avoids shot segmentation error and complexity, but also decreases the inter-frame difference analysis frequency. There are also many other problems of the cluster-based methods, such as extracting single image feature cannot fully represent frame image, manually determining the number of clusters and the initial cluster center caused a low degree of automation, the extracted key frame cannot represent the original videos. Therefore, this paper aims to improve the accuracy and the automation of key-frame extraction. In this paper, a novel cluster based key frame extraction approach is proposed. The benefits of this approach are as follows:

1) A video content analysis method is proposed to improve the representative of video feature data by using three visual features: color, texture and information complexity.
2) We develop a threshold optimization method to avoid manual selection of clustering threshold. This method can improve the automation of cluster-based key frame extraction.
3) We utilize the fusion of the frame density, inter-cluster distance and intra-cluster distance to

filter the key frame candidates and employ the max weight factor parameter to further refine key frame candidates. This method is favorable to improve the frame representation and the overall fidelity.

The rest of this paper is organized as follows. Section 2 describes the details of cluster-based methods. In section 3, we present the implementation of feature extraction method. Section 4 provides a detailed description of the proposed cluster based key frame extraction method. In section 5, we explore the performance of the proposed approach. Finally, the major work is discussed and wrapped up in Section 6.

2. Related work

It is quite common for cluster-based methods to transform the frame image into data points in feature space and cluster these data points to extract key frames. It is similar to the clustering algorithm, which gathers similar elements together, and takes the cluster center as the representative of clusters. Recently, some cluster-based key frame extraction methods have been proposed in literature [15,16].

The basic cluster-based methods can be categorized into two: automatic and semi-automatic cluster-based methods [17]. In general, semi-automatic cluster-based method requires manual determination the initial cluster centers and the number of clusters. Setting the number of clusters in advance may affect key frames extraction results [14]. It is more reasonable to determine the number of clusters according to different video contents in clustering process. Therefore, the automatic key frame extraction technology is more practical. Among the automatic cluster schemes, Kuanar extracts the color and texture features and propose an automated method of video key frame extraction using dynamic Delaunay graph clustering [10]. In [11], a fused key frame extraction framework is proposed. This method generates video summaries by combining sparse selection and agglomerative hierarchical clustering method based on mutual information. It extracts candidate key frames by an improved MIAHC algorithm caused the fidelity and operation efficiency are improved. In [12], the authors propose the coarse clustering and fine clustering key frame extraction. The traditional spectral clustering method with simple histogram features is used to remove most of the redundant frames. Then, the image classification method based on SIFT feature sparse coding is used to perform fine clustering for each time period. In [19], Liu and his colleges propose a key frame extraction method combining k-means algorithm and hierarchical clustering algorithm. They obtain initial clusters by employing the improved hierarchical clustering algorithm. Then they use the k-means algorithm to optimize the initial clusters to obtain the optimal clusters. In [20], the authors proposed a novel cluster-based algorithm, which is inspired by the idea of high-density peak search clustering algorithm. This method gathers similar frames into classes by integrating the important attributes of the video. In [21], the proposed cluster-based method combines image information entropy and uses a density clustering algorithm to extract key frames in gesture videos. In cluster-based key frame extraction, calculation method of clustering threshold has a great impact on the fidelity and compression rate of key frames. The core idea behind such automatic cluster-based methods is to set a favorable threshold. In literature, researchers usually receive the threshold by defining formula or by setting fixed value. Kuanar S. K. computes the cluster threshold by employing formula $2(1\text{-}\varepsilon)$ [10]. Jeong D. J. selects 0.0001 as the cluster threshold [12]. The other researchers compute cluster threshold by a self-defined formula [11,19]. In representative frame extraction, some cluster-based techniques take the cluster centers or centroids as the representative frames of each class. In [10], the author selects the frames which are closest to the cluster centroids as the representative frames. In [19,20], the authors directly extract cluster centers as the representative frames.

In general, those cluster-based methods may remain redundant because the clustering threshold setting influences optimal key frame extraction. Also, these methods evaluate the representative of the frames in one cluster by a single image feature. However, a single image feature cannot be able to fully characterize the frame content and complexity. In a more reasonable way, optimal threshold and feature fusion should be computed in cluster-based key frame extraction.

3. Materials and method

In this section, we provide a detailed description of the proposed OTMW method which includes feature extraction and key frame extraction. At first, the color, texture, and information complexity features are computed to express video content. Then, an optimization function is developed to compute the optimal clustering threshold. Next, the frame density, inter-distance and intra-distance are computed and fused as the clustering weight factor. Finally, a Max Weigh method is proposed to extract the cluster representative frame. The proposed approach is summarized in figure 1.

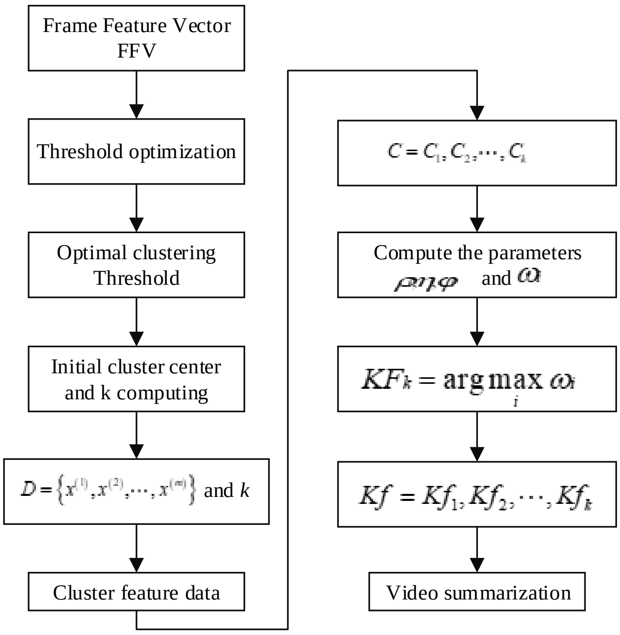

Frame Feature Vector FFV
Threshold optimization
Optimal clustering Threshold
Initial cluster center and k computing
$D=\{x^{(1)},x^{(2)},\cdots,x^{(m)}\}$ and k
Cluster feature data
$C = C_1, C_2, \cdots, C_k$
Compute the parameters $\rho\eta\varphi$ and ω
$KF_k = \arg\max_i \omega_i$
$Kf = Kf_1, Kf_2, \cdots, Kf_k$
Video summarization

Figure1. Framework of the proposed cluster-based method.

3.1. Feature extraction

In this section, we describe the proposed video content analysis method which distinguishes frames by computing feature data [22]. We extract the color, texture and information complexity features to discriminate different frame images. The video frame feature data is construct by

$$\mathrm{FFV}=[\mathrm{C\ T\ E}] \tag{1}$$

where C, T and E represent the color, texture and information complexity, respectively.

3.1.1. Color feature

We take the color feature as one feature to characterize the difference of frame images. We compute the first color moment, second color moment and third color moment in H, S, and V channels to construct the color feature data vectors of frame images. The first color moment reflects the brightness difference, which is calculated by

$$C_m = \frac{1}{w \times h} \sum_{p=1}^{w} \sum_{q=1}^{h} f_i(x_p, y_q) \qquad [1, 2] \tag{2}$$

where the parameters w and h are the pixel width and height, $f_i(x_p, y_q)$ is the pixel value in position (x_p, y_q), and $1 \le p \le w, 1 \le q \le h$. The second color moment reflects the color distribution range, which is compute by

$$C_v = (\frac{1}{w \times h} \sum_{p=1}^{w} \sum_{q=1}^{h} (f_i(x_p, y_q) - C_m)^2)^{\frac{1}{2}} \tag{3}$$

The third color moment represents the color distribution symmetry, which is computed by

$$C_s = (\frac{1}{w \times h} \sum_{p=1}^{w} \sum_{q=1}^{h} (f_i(x_p, y_q) - C_v)^3)^{\frac{1}{3}} \tag{4}$$

where C_m, C_v and C_s include first moment mean, second moment variance and third moment slope three parameters, respectively.

3.1.2. Texture feature

We take the texture feature as another feature to characterize the difference of frame images in image surface structural organization information\cite{r23}. We compute the mean of angular second moment, contrast, correlation and homogeneity texture features in 0, 45, 90 and 135 directions to construct video frame texture feature data vectors. The angular second moment characterizes the thickness and gray distribution uniformity of images, and it can be calculated by

$$T_A = \frac{1}{w \times h} \sum_{p=1}^{w} \sum_{q=1}^{h} f_i(x_p, y_q)^2 \tag{5}$$

The contrast characterizes the groove depth and clarity of images, it can be calculated by

$$T_{Con} = \frac{1}{w \times h} \sum_{p=1}^{w} \sum_{q=1}^{h} (p-q)^2 f_i(x_p, y_q) \tag{6}$$

The correlation characterizes the local gray similarity in row or column direction, it can be calculated by

$$T_{Cor} = \frac{\sum_{p=1}^{w} \sum_{q=1}^{h} (x_p - \sum_{p=1}^{w} \sum_{q=1}^{h} f_i(x_p, y_q) \times p) \times (y_p - \sum_{p=1}^{w} \sum_{q=1}^{h} f_i(x_p, y_q) \times q) \times f_i(x_p, y_q)^2}{\sum_{p=1}^{w} \sum_{q=1}^{h} f_i(x_p, y_q) \times (x_p - \sum_{p=1}^{w} \sum_{q=1}^{h} f_i(x_p, y_q) \times p)^2} \tag{7}$$

The homogenization characterizes the local gray level uniformity of images, it can be calculated by

$$T_H = \sum_{p=1}^{w} \sum_{q=1}^{h} f_i(x_p, y_q) \times \frac{1}{1 + (p-q)^2} \tag{8}$$

3.1.3. Information complexity feature

We take the information complexity as the last feature to characterize the difference of frame images in aggregated and spatial feature. Information entropy proposed by Shannon is a holistic perspective information complexity measuring method [24]. It can characterize the image aggregated and spatial features. Larger image information entropy and greater internal non-uniformity degree commonly occur together in higher diversity level. The two-dimensional information entropy Ef_i can be calculated by

$$E = -\frac{1}{w \times h} Cf \times \log_2 Cf \tag{9}$$

where Cf is the occurrence probability of each gray level in i-th frame image.

3.2. Clustering for key frame extraction

In this section, we describe the proposed cluster based key frame extraction method, which develops a new optimization function to compute the optimal threshold.

3.2.1. Threshold optimization

We narrow the search interval of optimal threshold by computing the function values of trial points. In cluster-based method, the fidelity [8] and ratio [7] are negatively and positively correlated with the threshold, respectively. Therefore, we infer that the quality of key frames is optimal when fidelity and ratio are infinitely close. We introduce a new parameter FR to characterize this relationship and to obtain the optimal key frames. The distance between frame $x^{(i)}$ and frame $x^{(j)}$ is compute by

$$d_{ij} = \| x^{(i)} - x^{(j)} \|_2 = \sqrt{\sum_{u=1}^{n} |x_u^{(i)} - x_u^{(j)}|^2} \tag{10}$$

where $(i, j) \in [1, 2, ..., m]$.

The average distance is computed by

$$d_c = \frac{2}{m(m-2)} \sum_{i=1}^{m} \sum_{j=1}^{m} d_{ij} \tag{11}$$

The threshold is defined as

$$t = d_c \pm \varepsilon \times std_{ij} \tag{12}$$

where ε is a variable factor, std_{ij} is the standard deviation of d_{ij}. We define the new parameter FR as

$$FR(t) = \frac{fidelity(t)}{ratio(t)} \tag{13}$$

The threshold optimization function is defined as

$$f(t) = | \ FR(t) - 1| \tag{14}$$

Giving *a*, *b*(a<b, $a = d_c$-3×std_{ij} , $b = d_c$+3×std_{ij}) and *c*. We compute threshold change factor *c* and compute $f(a)$, $f(b)$ and $f(c)$. We change *c* by comparing the $f(a)$, $f(b)$ and $f(c)$. When $f(c) < 0$, Fidelity is less than Ratio. Therefore，we change *c* to increases Fidelity and decrease Ratio. When $f(c) \geq 0$, Fidelity is more than Ratio. We change *c* to decrease Fidelity and increase Ratio. If $f(c)$=$f(a)$=$f(b)$ in consecutive three times and $\left(b - a\right) \leq 0.001$， we compute the optimal cluster threshold by ∂=$\frac{a+b}{2}$. The calculation process is Algorithm1.

Algorithm 1 Compute the optimal cluster threshold

```
Input: parameters $a$ and $b$
Output: optimal threshold $\rho$
compute $c$=$a$+0.618*($b$-$a$);
compute $f(a)$, f($b$), f($c$);
    if not $f(a)$ = $f(b)$ = $f(c)$ and ($b-a$) > 0.001 then
        while ($b-a$) > 0.001 do
        compute $c$=$a$+0.382*($b$-$a$);
        compute $f(a)$, f($b$), f($c$);
        while not $f(a) = f(b) = f(c)$ do
            if $f(c) < 0$ then
                change $c$ to increase fidelity and decrease ratio, $b = c$;
            else
                change $c$ to decrease fidelity and increase ratio, $a = c$;
            end if
        end while
    end while
    $\rho=(a+b)/2$;
else
    $\rho=(a+b)/2$;
end if
```

3.2.2. Initial cluster centers and the number of clusters

In this section, we compute the initial cluster centers and the number of clusters by clustering the data of *FFV*. Frame Feature Vector data sets $FFV = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(m)}\right\}$ is a 14-dimensional video frame feature data, which includes color, texture and information entropy features.

The density of the sample frame i in video stream is defined as:

$$\rho_i = \sum_j e^{-\left(\frac{d_{ij}}{d_c}\right)^2} \tag{15}$$

The distance of the j-th frame image from the i-th frame image in the same cluster is defined as:

$$\eta_i = dist(x^{(i)}, x^{(j)}), i \in D, j \neq i \tag{16}$$

where i and j are in the same cluster.

The distance between the i-th frame element and the element j of another cluster is defined as:

$$\varphi_i = dist(x^{(i)}, x^{(j)}), i \in D, j \in \mu^{(k)} \tag{17}$$

where i is a frame in one cluster, j is the elements of the cluster center that has completed the clustering.

According Eq. (17), φ_i may contain multiple elements. The product of ρ_i, η_i^{-1} and φ_i weight factor is defined as the weighted product:

$$\omega(x) = \Pi\left(\rho_i \times \eta_i^{-1} \times \varphi_i\right) \tag{18}$$

Initial cluster center and cluster number k are directly related to $\omega(x)$, the solution is shown in Algorithm 2. Firstly, the density of samples is calculated using Eq. (15), and then the maximum density frame is selected as the first initial clustering center $\mu^{(1)}$. The distance between frames is computed and set as the first initial cluster center. Then the frames are classified with a less than t

distance into the first cluster, and these frames are removed from *D*. The ω_i of *D* is also calculated using Eq. (18), and the second initial cluster center $\mu^{(2)}$ is obtained by calculating the maximum of $\omega(x)$. Similarity, the samples that satisfies the same condition are classified as the second, three and *k* cluster, and are removed from *D*. Finally, all samples are assigned to cluster, and the initial cluster centers $\mu^{(1)}$, $\mu^{(2)},\cdots,$ $\mu^{(k)}$ and the number of *k* are obtained. Clustering process is shown in Figure 3.

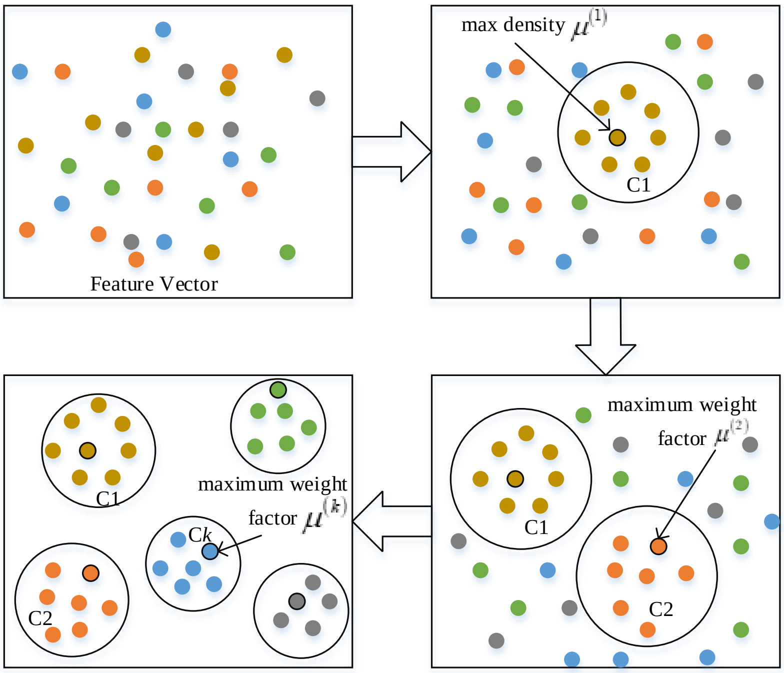


Figure 2. The computation of initial cluster centers and the number of clusters.

Algorithm 2 Compute the initial cluster centers and the number of clusters

```
Input: $FFV = \{x^{(1)}, x^{(2)}, ..., x^{(m)}\}$
Output: the initial cluster centers $\{\mu^{(1)}, \mu^{(2)}, ..., \mu^{(k)}\}$ and the number of clusters $k$
for each sample $i \in D$
    compute the density $\rho_i$;
end for
while $D \neq \varnothing$ do
    set the frame with maximum $\rho_i$ as the initial cluster center $\mu^{(1)}$;
    if $d(\mu^{(1)}, i) \leq t$ then
        $i \in C_1$, remove $i$ from $D$;
    end if
    for each sample $i \in D \cup (C_i \notin D)$ do
        compute $\eta_i, \varphi_i, \omega_i$;
        cluster center $\mu^{(i)} = \arg\min_i \omega_i$
       $i \in C_i$, remove $i$ from $D$;
    end for
end while
```

3.2.3. Key frame extraction

In this section, we classify frame images into k clusters $C=\{C_1,C_2,...,C_k\}$ and extract representative frames from this clusters. The error square sum criterion function is used as the criterion function. The frame images are classified into different clusters by employing Algorithm 3. Giving the cluster $C=C_1,C_2,\cdots,C_k$, the specific steps are as follows:

Step1: Calculate frame ρ_i, η_i and φ_i in cluster C_1 by Eq. (13), (14) and (15).

Step2: Compute maximum weight factor by Eq. (17) and select key-frame f_1 from cluster C_1.

Step3: Similarly, the key-frame f_k can be obtained by the computing of maximum weight factor in Eq. (17).

Step4: Repeat Step3 until all cluster representative frames are selected.

Step5: Key-frames $f=f_1,f_2,\cdots,f_k$ are extracted. Video summarization is generated.

Algorithm 3 Cluster the frame feature value

```
Input: frame feature value $FFV=\{x^{(1)},x^{(2)},...,x^{(m)}\}$, the initial cluster centers $\{\mu^{(1)},\mu^{(2)},...,\mu^{(k)}\}$ and the number of clusters k.
Output: k clusters $C=\{C_1,C_2,...,C_k\}$
    Let $C_i=\varnothing(1\le i\le k)$;
repeat
    for $j=1,2,...,m$ do
        compute the distance between the sample $x^{(j)}$ and the cluster center $\mu^{(i)}(1\le i\le k)$;
        compute the $\lambda_j=argmind_{ji}$;
        divide the sample $x^{(j)}$ into the nearest cluster $C_{\lambda j}=C_{\lambda j}\cup x^{(j)}$;
    end for
    for $i=1,2,...,k$ do
        calculate the new cluster center $(\mu^{(i)})'=\frac{1}{|C_i|}\sum_{x\in C_i}x$;
        if $(\mu^{(i)})'=\mu^{(i)}$ then
            update the current cluster center $\mu^{(i)}$ to $(\mu^{(i)})'$;
        else
            keep the current mean vector unchanged;
        endif
    end for
until the current cluster center vectors are not updated.
```

4. Results and discussion

In this section, we turn to an empirical study on the proposed OTMW method in key frame extraction tasks and compare it with state-of-the-art cluster-based key frame extraction methods such as HVM [25], ACSC [26], RGPH [27] and FCME [28]. We report improved performance across open video dataset. We conduct a set of experiments by using surveillance, documentary, lecture on TV and phone recording four different video datasets. These videos are publicly shared on https://open-video.org/. In this section, we take Hcil2000_01 video as an example to report the performance of OTMW method. Here Hcil2000_01 is a random video of open video dataset.

4.1. Optimization of threshold

The parameters of Hcil2000_01 video in threshold optimization is shown in table 1. In Hcil2000_01 video, the average distance d_c and standard deviation std_{ij} of d_{ij} are 2.3107 and 0.2259, respectively. Therefore, the parameters *a* = 1.6442, *b* = 2.9993. The variable interval of parameter *c* is [1.6442,2.9993]. The parameter *c* is computed by *c* = *a* + 0.618*(*b*-*a*) = 2.481652. In sixth iteration, the *f*(*a*) = *f*(*b*) = *f*(*c*) = 0.0063. The calculation of parameter *c* is changed to *c* = *a* + 0.382*(*b*-*a*) in subsequent iterations. As shown in table1, the value of *f*(*a*) = *f*(b) = *f*(*c*) = 0.0063 are not change in the next two iterations. However, *b*-*a* = 0.122156 > 0.001. Finally, *b*-*a* = 0.000509 < 0.001 in 13th iteration. Therefore, the optimal threshold $\tilde{n} = \frac{a+b}{2} = 1.644709$.

Table 1. The parameters of Hcil2000_01 video in threshold optimization.

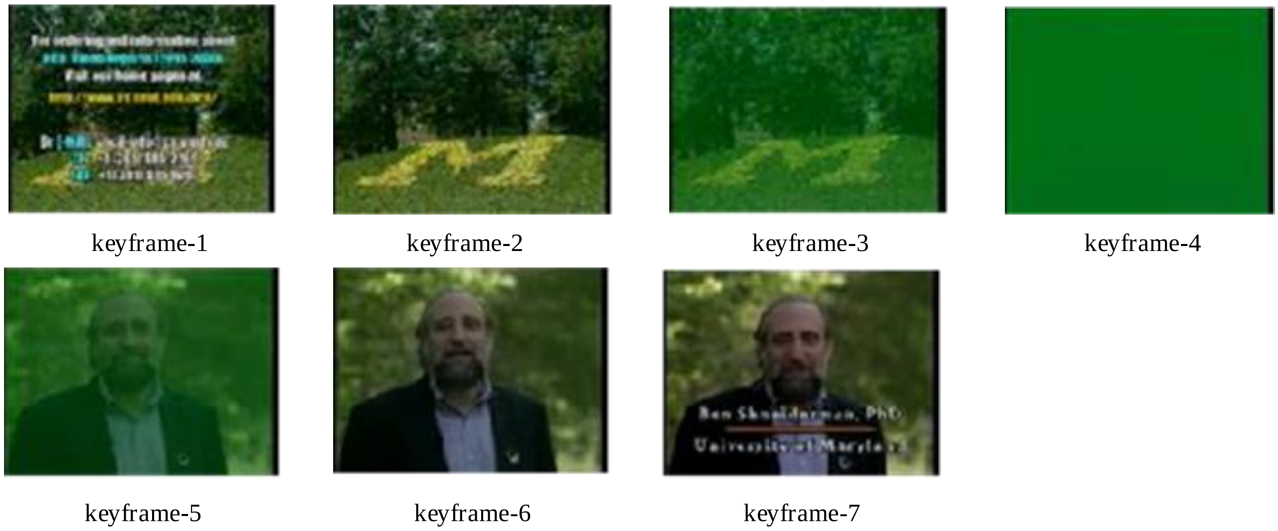
iterations	a	b	c	f'(a)	f'(b)	f'(c)
1	1.6442	2.9993	2.4817	-0.63	-7.6	-6.95
2	1.6442	2.4817	2.1617	-0.63	-6.95	-3.33
3	1.6442	2.1617	1.9640	-0.63	-3.33	-2.11
4	1.6442	1.9640	1.8419	-0.63	-2.11	-2.11
5	1.6442	1.8419	1.7664	-0.63	-2.11	-0.63
6	1.6442	1.7663	1.7197	-0.63	-0.63	-0.63
7	1.6442	1.7197	1.6720	-0.63	-0.63	-0.63
8	1.6442	1.6720	1.6544	-0.63	-0.63	-0.63
9	1.6442	1.6544	1.6479	-0.63	-0.63	-0.63
10	1.6442	1.6479	1.6455	-0.63	-0.63	-0.63
11	1.6442	1.6455	1.6447	-0.63	-0.63	-0.63
12	1.6442	1.6447	1.6443	-0.63	-0.63	-0.63
13	1.6442	1.6447	1.6445	-0.63	-0.63	-0.63

Note: $f'(a) = 100 * f(a)$, $f'(b) = 100 * f(b)$ and $f'(c) = 100 * f(c)$

4.2. Extraction of key frames

The key frames are extracted by employing OTMW method across open video dataset. The key frames of Hcil2000_01 video is shown in figure 3. The fidelity and ratio results are shown in table 2. *Nf* represents the number of frames, *Nrf* represents the number of key frames. The fidelity measures of different videos are changed from 93 to 98 with an average of 96.12. The ratio measures are changed from 95 to 98 with an average of 97.13. The key frames are consistent with artificial judgment.

Figure 3. Results of the video from Open video dataset.



keyframe-1 keyframe-2 keyframe-3 keyframe-4

keyframe-5 keyframe-6 keyframe-7

Table 2. The fidelity and ratio performance of videos.

video name	Nf	Nrf	fidelity	ratio
Traffic monitoring	120	5	95.94	95.83
Office video_01	161	5	96.84	96.89
Entertainment_01	151	5	97.14	96.69
Entertainment_02	191	7	98.01	96.34
Entertainment_03	101	4	97.79	96.04
Office video_02	77	3	96.34	96.10
Indi012	201	6	94.53	97.00
UGS01_008	301	5	93.58	98.34
UGS07_005	400	8	95.07	98.00
UGS01_006	360	8	95.81	97.78
UGS01_001	331	7	94.26	97.89
Hcil2000_01	210	7	95.58	96.67
Marchionini	100	4	96.96	96.00
RayDiessenIBM	230	6	96.73	97.39
Entertainment_04	201	3	97.63	98.51
Daily life	283	5	97.45	98.23

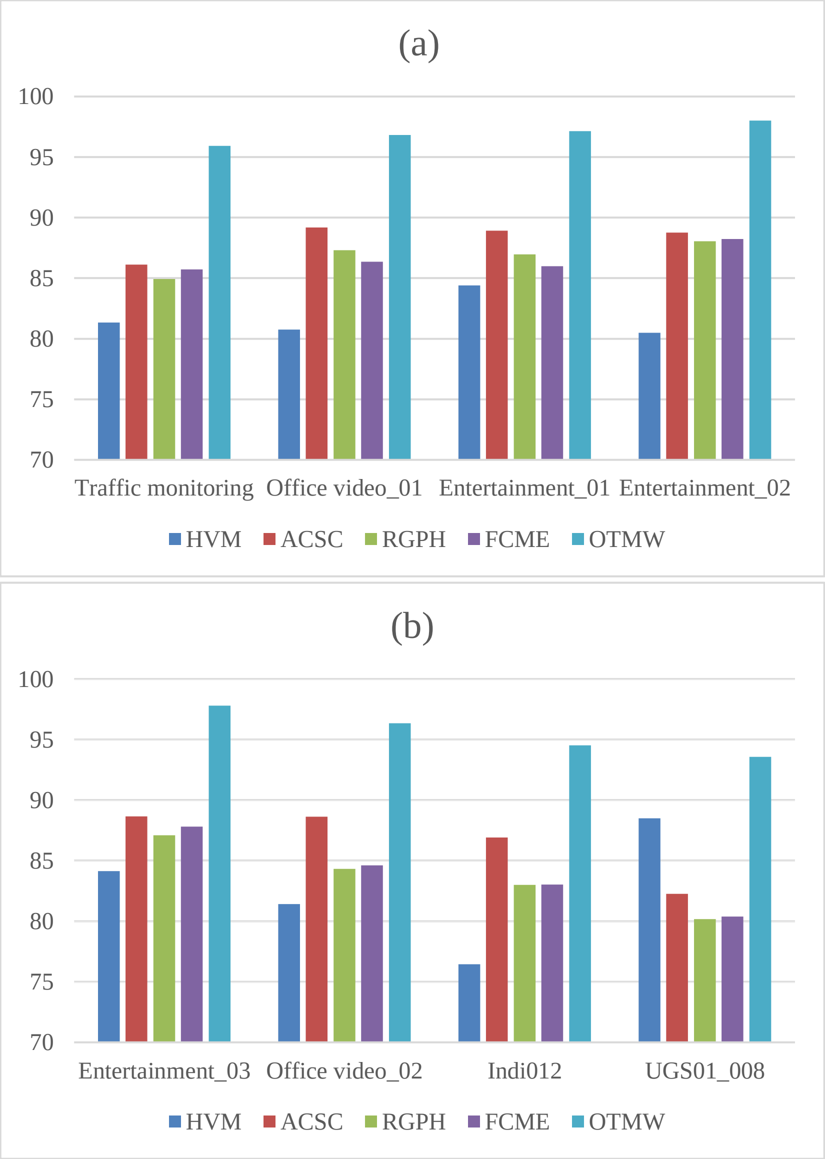
4.3. Comparison with several state-of-the-art methods

We compare OTMW with state-of-the-art cluster-based algorithm in term of the fidelity and ratio measure performance. In experimental, the number of clusters of semi-automatic cluster-based methods are same as OTMW method. The results of fidelity and ratio are shown in figure 4 and table 3. HVM method with an average fidelity of 83.75 and an average ratio of 95.59. ACSC with an average fidelity of 85.75 and an average ratio of 94.79. The average fidelity of RGPH and FCME are 85.61 and 85.38. The proposed OTMW method with an average fidelity of 96.24 and with an average ratio of 97.15. OTMW method achieves a 10.63–12.49 fidelity improvement over other cluster-based methods. The fluctuations of ratio measure of different videos are shown in figure 5.

OTMW method with a ratio variance of 0.73. The ratio variance of HVM and ACSC cluster-based methods are 22.11 and 11.12. They are 15 and 30 times larger than OTMW, respectively. OTMW method achieves a 1.56–2.24 ratio improvement over other cluster-based methods and has a small fluctuation.

4.3.1. Extraction of key frames on various datasets

To assess the performance of OTMW method, we consider key frame extraction tasks on surveillance, documentary, lecture on TV and phone recording datasets. The HVM method achieves an average fidelity of 82.42, 86.90,84.36 and 81.22, respectively. It achieves an average ratio of 95.76,92.73, 94.767 and 99.05. ACSC method achieves an average fidelity of 88.40, 84.07, 81.02 and 89.57. It achieves an average ratio of 92.19, 97.37, 97.42 and 92.13. The average fidelities of FCME method are 86.39, 83.77, 85.51 and 87.62. The average ratios of RGPH method are 86.39, 83.18, 85.94 and 86.45. OTMW method achieves an average fidelity of 97.07, 94.40, 95.87 and 97.54 and an average ratio of 96.43, 97.83, 97.01 and 98.37. The fidelity measures on various datasets are shown in figure 6. OTMW method achieves a 9.91-11.66 fidelity and 0.91-2.77 ratio improvement over HVM, ACSC, FCME, RGPH.



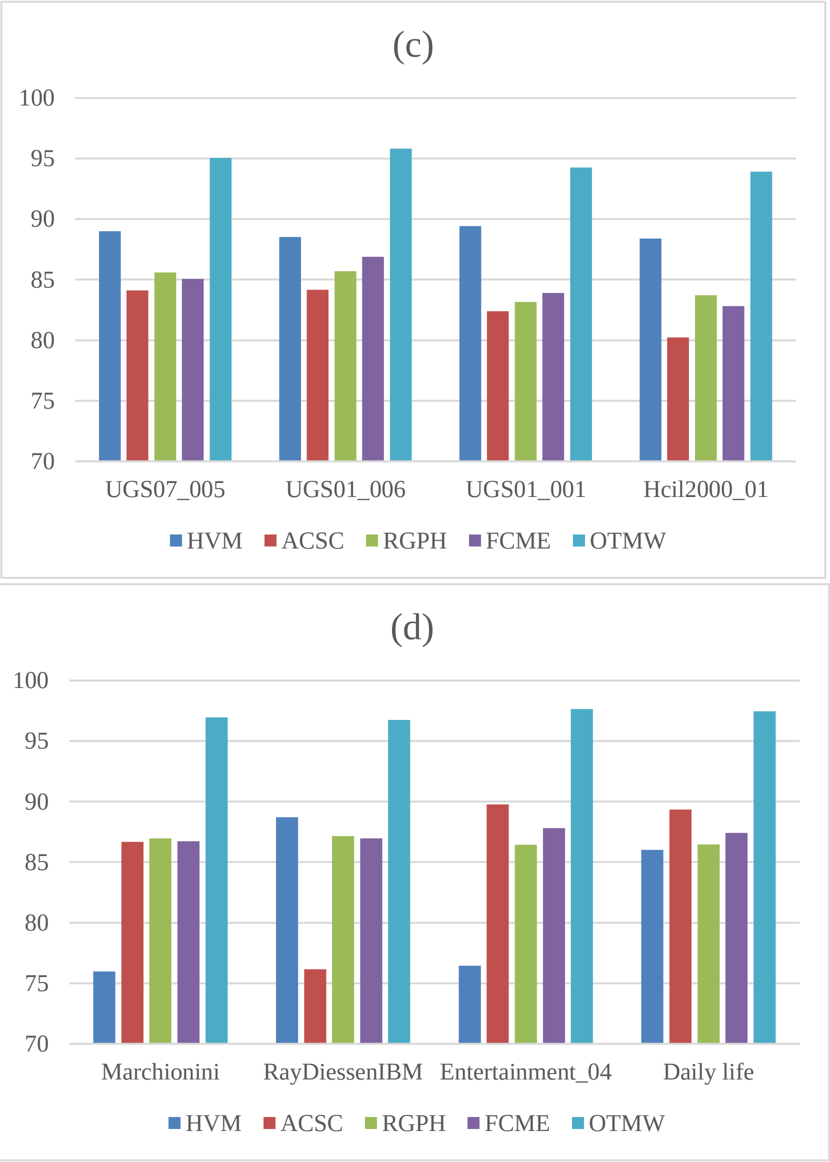


Figure 4. The fidelity measure performance of cluster-based methods.

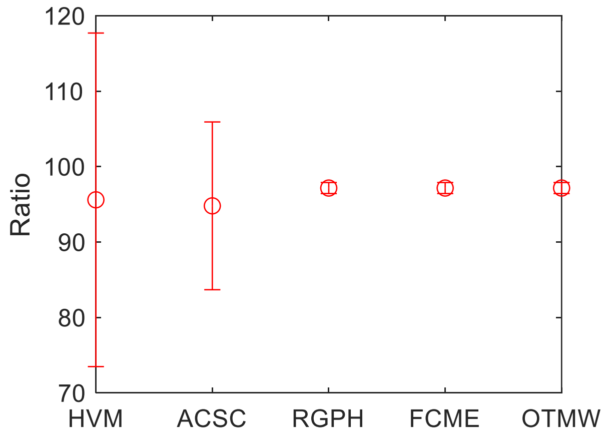


Figure 5. The ratio error-bar of cluster-based methods.

Table 3. The ratio measure performance of cluster-based methods.

video name	HVM	ACSC	OTMW
Traffic monitoring	97.5	95.0	95.83
Officevideo_01	98.76	88.20	96.8
Entertainment_01	81.46	92.05	96.67
Entertainment_02	98.43	93.72	96.34
Entertainment_03	98.14	94.06	96.04
Officevideo_02	98.70	89.61	96.10
Indi012	99.50	94.50	97.02
UGS01_008	95.26	98.50	98.34
UGS07_005	93.75	98.50	98.00
UGS01_006	96.11	97.78	97.78
UGS01_001	89.43	98.19	97.89
Hcil2000_01	90.95	97.14	96.67
Marchionini	99.00	96.00	96.96
RayDiessenIBM	94.35	99.13	97.39
Entertainment_04	99.50	89.55	98.51
Daily life	98.58	94.70	98.23

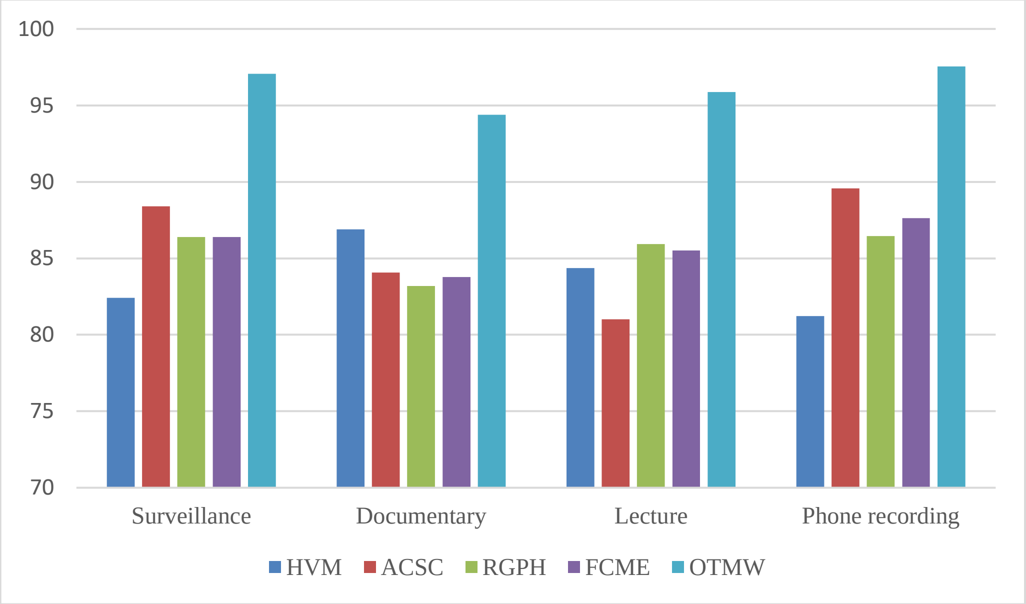


Figure 6. The fidelity results of cluster-based methods on different datasets.

5. Conclusion

In this paper, an innovative cluster based key frame extraction method is presented for multi-type videos. This method analyzes the video content by extracting the color, texture and information complexity features. The threshold optimization function is constrained by fidelity and ratio measures. It avoids the dependence on a fixed threshold problem in traditional cluster-based method by computing the optimal threshold. The parameters density ρ_i , inter-distance η_i , intra-distance φ_i and weight factor ω_i are used to compute the initial cluster centers and the number of clusters, extract representative frames from *k* clusters. The method shows promising result on different video datasets. Meanwhile, OTMW achieves competitive and even better fidelity and ratio measure performance when compared with several state-of-the-art cluster-based methods. Overall, we found that OTMW well suited to process key frame extraction problem in the field of static video summarization. However, whether the proposed method also applies in the real-life production and life environment is subject to be verified. In the future, we will explore to apply our proposed method for real-time video surveillance. In addition, we will investigate how to integrate the proposed method into the camera client and how to apply it to daily production and life.

Acknowledgments

The subject is sponsored by the National Natural Science Foundation of P. R. China (No. 61872196, No. 61872194, No. 61902196, No. 62102194 and No. 62102196), Scientific and Technological Support Project of Jiangsu Province (No. BE2019740, No. BK20200753 and No. 20KJB520001), Major Natural Science Research Projects in Colleges and Universities of Jiangsu Province (No. 18KJA520008), Major Natural Science Research Projects in Colleges and Universities of Anhui Province (No.KJ2019ZD20), Six Talent Peaks Project of Jiangsu Province (No. RJFW-111), Postgraduate Research and Practice Innovation Program of Jiangsu Province (No. KYCX19_0909, No. KYCX19_0911, No. KYCX20_0759, No. KYCX21_0787, No. KYCX21_0788 and No. KYCX21_0799).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. A. G. Money, H. Agius, Video summarization: A conceptual framework and survey of the state of the art, *J. Visual Commun. Image Represent.*, **19** (2008), 121–143.
2. W. Li, D. Qi, C. Zhang, J. Guo, J. Yao, Video summarization based on mutual information and entropy sliding window method, *Entropy*, (2020), 22. doi:10.3390/e22111285.
3. X. Yan, S. Z. Gilani, M. Feng, L. Zhang, H. Qin, A. Mian, Self-supervised learning to detect key frames in videos, *Sensors*, (2020), 20. doi:10.3390/s20236941.
4. C. Huang, H. Wang, A novel key-frames selection framework for comprehensive video summarization, *IEEE Trans. Circuits Syst. Video Technol.*, **30** (2020), 577–589.
5. I. Mehmood, S. Rho, S. W. Baik, Divide-and-conquer based summarization framework for extracting affective video content, *Neurocomputing*, **174** (2016), 393–403.
6. G. H. Song, Q. G. Ji, Z. M. Lu, Z. D. Fang, Z. H. Xie, A novel video abstraction method based on fast clustering of the regions of interest in key frames, *AEU Int. J. Electron. Commun*. **68** (2014), 783–794.
7. H. Rachida, E. Abdessamad, A. Karim, MSKVS: Adaptive mean shift-based keyframe extraction for video summarization and a new objective verification approach, *J. Visual Commun. Image Represent.*, **55** (2018), 179–200.

8. G. Ciocca, R. Schettini, Erratum to: An innovative algorithm for key frame extraction in video summarization, *J. Real Time Image Process*., **8** (2013), 225.
9. H. S. Chang, S. Sull, S.U. Lee, Efficient video indexing scheme for content-based retrieval, *IEEE Transact. Circ. Syst. Video Technol*., **9** (1999), 1269–1279.
10. S. K. Kuanar, R. Panda, A. S. Chowdhury, Video key frame extraction through dynamic Delaunay clustering with a structural constraint, *J. Visual Commun. Image Represent*., **24** (2013), 1212–1227.
11. W. Jiang, M. Fei, Z. Song, W. Mao, New fusional framework combining sparse selection and clustering for key frame extraction, *Iet Computer Vision*, **10** (2016), 280–288.
12. D. J. Jeong, H. J. Yoo, N. I. Cho, Consumer video summarization based on image quality and representativeness measure, *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, (2015), pp. 572–576.
13. Y. Yin, R. Thapliya, R. Zimmermann, Encoded semantic tree for automatic user profiling applied to personalized video summarization, *IEEE Transact. Circ. Syst. Video Technol*., **28** (2018), 181–192.
14. N. Ejaz, I. Mehmood, S. W. Baik, Efficient visual attention-based framework for extracting key frames from videos, *Signal Process. Image Commun*., **28** (2013), 34–44.
15. P. Zheng, L. Shuai, S. A. Kumar, M. Khan, Visual attention feature (VAF): A novel strategy for visual tracking based on cloud platform in intelligent surveillance systems, *J. Parallel Distrib. Comput*., **120** (2018), 182–194.
16. Y. Zhang, X. Liang, D. Zhang, M. Tan, E. P. Xing, Unsupervised object-level video summarization with online motion auto-encoder, *Pattern Recognit. Lett*., **130** (2020), 376–385.
17. Y. Z. Zhang, Key frame extraction of surveillance video based on frequency domain analysis, *Intell. Autom. Soft Comput*., **258** (2021), 259–272.
18. Y. Lu, Key frame extraction based on global motion statistics for team-sport videos, *Mult. Syst*., (2021), 1–15.
19. H. Liu, H. Hao, Key frame extraction based on improved hierarchical clustering algorithm, *International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, (2014).
20. J. Wu, S. H. Zhong, J. Jiang, Y. Yang, A novel clustering method for static video summarization, *Mult. Tools Appl*., **76** (2017), 9625–9641.
21. H. Tang, H. Liu, W. Xiao, Fast and robust dynamic hand gesture recognition via key frames extraction and feature fusion, *Neurocomputing*, **331** (2019), 424–433.
22. H. Zhou, A. H. Sadka, M. R. Swash, J. Azizi, U. A. Sadiq, Feature extraction and clustering for dynamic video summarization, *Neurocomputing*, **73** (2010), 1718–1729.
23. K. Xiao, S. Lingfeng, G. Wenzhong, C. Dewang, Multi-dimensional traffic congestion detection based on fusion of visual features and convolutional neural network, *IEEE Transact. Intell. Transp. Syst*., **20** (2019), 2157–2170.
24. T. Koch, G. Vazquez-Vilar, A rigorous approach to high-resolution entropy-constrained vector quantization, *IEEE Trans. Inf. Theory*, **64** (2018), 2609–2625.
25. D. Liu, G. Hua, T. Chen, A hierarchical visual model for video object summarization, *IEEE Transact. Pattern Anal. Mach. Intell.*, **32** (2010), 2178–2190.
26. C. Fahy, S. Yang, M. Gongora, Ant colony stream clustering: A fast density clustering algorithm for dynamic data streams, *IEEE Transact. Cybernet.*, **49** (2018), 2215–2228.

27. J. Niu, D. Huo, K. Wang, C. Tong, Real-time generation of personalized home video summaries on mobile devices, *Neurocomputing*, **120** (2013), 404–414.
28. C. Yang, L. Chuang, Y. Lin, Epistasis analysis using an improved Fuzzy C-means-based entropy approach, *IEEE Trans. Fuzzy Syst.*, **28** (2020), 718–730.