ELECTRONICS AND ELECTRICAL ENGINEERING
ISSN 1392 – 1215 2011. No. 10(116)
ELEKTRONIKA IR ELEKTROTECHNIKA

T 120 SYSTEM ENGINEERING, COMPUTER TECHNOLOGY
SISTEMŲ INŽINERIJA, KOMPIUTERINĖS TECHNOLOGIJOS

A New Method for Video Parsing of Uncompressed Digital Video Streams

H. I. Eskikurt, B. Boru
Electronics & Computer Education Department, Technical Education Faculty, Sakarya University, Sakarya, Turkey, phones: +90 264 2956448, +90 264 2956464, e-mails: eskikurt@sakarya.edu.tr, barisb@sakarya.edu.tr

crossref http://dx.doi.org/10.5755/j01.eee.116.10.891

Introduction

With wide spread use of computers and digital equipments, and digital communication getting easier and more common to apply than the analogue communication through the platforms like the Internet, the need for transferring the video information to the digital medium has arisen, as it happened in all types of information.

Digital video is applied by benefiting from the weakness of the human eye. A human eye can only sense consecutive figures changing more than 15-20 frames in a second. A digital video is obtained by photographing a video scene as much as 15~30 frames in a second. The information about how many frames in a digital video are taken and recorded in a second is called "fps (frame per second)".

As mentioned above, digital video information is digital data pieces coming one after another. If there is a sound record in a video, it consists of both picture frames and sound information.

As a result of the technological advances today, almost all videos are used by recording in a digital form. Since digital video records are increasing more and more, the need for fast access appears when a search is desired. For example, to get the information about the video records of a person or a vehicle in an archive of security videos recorded, the old records in the archive should be searched from beginning to end.

As the standard text search methods cannot be used in digital video, the picture comparison methods are not suitable as well, because of the need for computational overhead and the demands for the meaningful search [1]. Therefore, special methods should be used for video access. Researchers dealing with video access have made a suggestion that instead of searching in the whole video, it would be better to search in a video index got by processing the video before, and then access to video segments referenced by the suitable results [1, 2].

In the next section, studies about video parsing and indexing algorithms were given in details through the examples. In the third section, the proposed algorithm for the filtered video histogram comparison was explained and the steps of this algorithm were defined. The evaluation of this algorithm and the comparisons were given in the fourth section.

Video parsing and indexing

If a subject title in a book is wanted to be accessed, the word can be found from the index or contents of the book and then the page referenced is directed. To make a search in a video and be able to access to a wanted scene, it is better to generate an index for the video and store it to the special database (Fig. 1) instead of scanning all the information. Each video added to the video database could be saved after having an index with a number of procedures. When a browsing is made, the required object is scanned in the index. If a matching is found the video segment referenced is showed to the user.

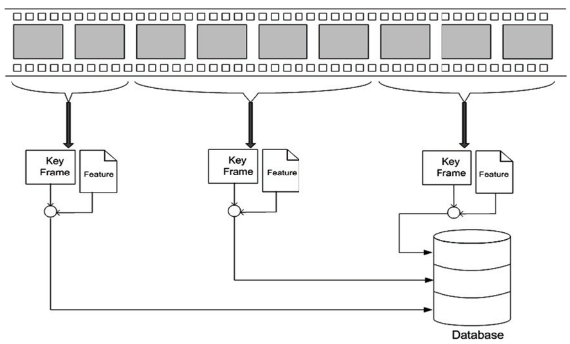

Fig. 1. Parsing a video and storing the summarized information to the database by referring to the segments

Video segments consist of consecutive frames which have a meaningful integrity. Features and key frames defined for each segment constitute the video index. Video parsing, the most important section of video indexing, is the grouping of the frames having a meaningful integrity (Fig. 2). Thus, this group is defined with information in the index. Also, it provides the preparation of defining the segments showed after browsing.

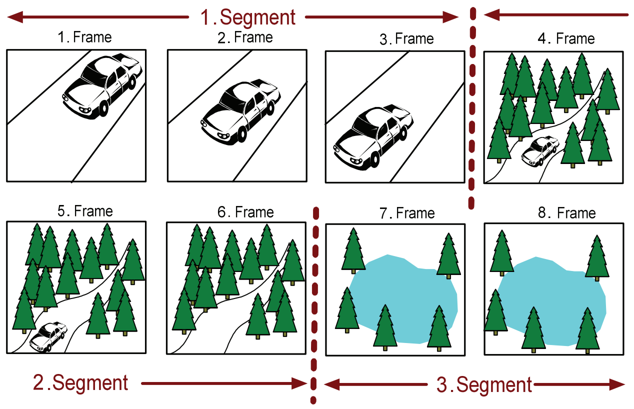

Fig. 2. Examples of consecutive video frames and segments

In the literature, a lot of researches about video databases and access can be found dealing with video parsing [1, 2]. The developed methods in these researches are focused on the right comprehending of video segments as defining the transitions in which there are meaningful changes on consecutive video frames.

There are two type segment transitions, such as gradual and sudden. Sudden segment transitions are usually formed by stopping the record of the camcorder like in filming and restarting for a different scene. Gradual segment transitions are the switching the scene contents gradually as happening by using to combine two segments in the film effects. Good examples of gradual transitions are fade-effect and dissolve effect. Because of the soft change instead of sharp change between consecutive frames, it is difficult to define gradual segment transition. These types of sensing should be separated from camcorder and object movements. Especially, it is too difficult to sense the segment transitions consisting of brightness level change compared to the other segment transitions.

Video parsing algorithms usually performs segment sensing by getting visual changing rates between consecutive frames based on the idea that there is a passing through the segments when the changing rate is high. Parsing algorithms start to browse from the first frame of video and evaluate the consecutive video frames. In some cases, even if there is no meaningful difference between two video frames, some big differences could appear when the numerical values were examined. The best examples of these types of changes are camcorder movement and brightness level change.

During the filming, even though the contents of the scene are the same, the objects in the scene move a few pixels between two consecutive frames (Fig. 3.a). Similarly again, changing the light source and the camcorder viewing angle result in change in the darkness level of all pixels in the video frame (Fig. 3.b). This causes big differences in the pixel values of two consecutive frames, which is sensed as a wrong segment transition. Therefore, parsing algorithms should have low sensibility to camcorder movement and brightness level change.

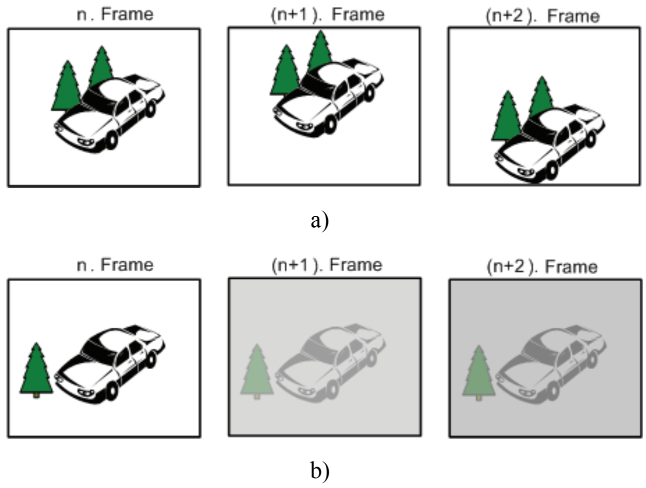

Fig. 3. An example of camcorder movements(a) and an example of brightness level changes (b)

The parsing algorithms for uncompressed videos usually obtain a similarity rate between the consecutive frames due to the technique used. If the similarity rate of two consecutive frames is too low, it is accepted that there is segment distinction and the video parsing can be done. In these techniques, over which values for consecutive frame similarities can be assumed as a segment transition, are defined with a threshold value. In literature, these types of algorithms are called threshold based algorithms.

The algorithm having the simplest approach among the video parsing algorithms and being the fundamental of the other algorithms is the pixel comparison algorithm. This algorithm defines a change value for each video frame as calculating the total differences of the overlapped pixels between two consecutive video frames. Defined change value is compared with a predefined threshold value and then it is decided whether it is segment transition or not. Due to the higher computational overhead and the sensitivities of the camcorder movement and brightness level change, this simple method is not preferred today. Block based pixel comparison algorithm, developed as an alternative to the pixel comparison algorithm was designed by Kasturi R., Jain R. in a way that comparing the overlapped blocks with a similarity rate obtained from the pixel values and average darkness level values as dividing the video frame into blocks [3]. This algorithm increased the computational overhead together with decreasing the sensibility of the brightness level change.

To decrease the computational overhead and the sensibility to the camcorder objects movement, Histogram based comparison methods instead of pixel based approaches were developed. The main idea of the histogram comparison methods is that there is not a big difference with the histograms of two consecutive frames which have unchanging background and objects (moving or not) [2]. In addition to that, histogram is not sensible to the picture rotation and the changing of the shooting angle. According to these principles, there are a number of studies related to histogram comparisons of the consecutive frames.

In the first histogram based parsing algorithm, the histograms of two consecutive frames were compared and a segment transition approach was applied by [2]. In literature, some amendments to this algorithm can be seen [4]. In these methods, although the sensibility to the camcorder and object movements is lower, the sensibility to the brightness level change is considerably high, because the brightness level change is completely replacing the histogram of the picture [5].

Block based comparison techniques were developed by [6], based upon the histogram comparison techniques as mentioned above. In this technique, a histogram comparison for the overlapped blocks in consecutive frames was done by dividing the video frames into blocks. Due to the failure of these methods in gradual segment transitions, twin comparison technique was developed by [7]. In this technique, a second sub-threshold value was used for sensing the differences between the frames occurred in gradual segment transition and then the values above this sub-threshold value and the differences between the consecutive frames were added. If the result is higher than the real threshold value, gradual segment transition could be defined. Boreczky and Rowe have decided that twin comparison technique was simple and reliable [4].

The proposed algorithm for the filtered video histogram comparison

The success of parsing algorithms on uncompressed videos has been defined as low performance. These algorithms have better sensibilities to the brightness level change and camcorder movements. This work introduces a new algorithm for filtered video histogram comparison to eliminate the disadvantages in the histogram based methods. The video parsing process have benefited from the picture filtering techniques to have higher performance and functionality [8, 9]. After the whole video stream was filtered with the picture filtering techniques, it is subjected to histogram comparison.

The main idea of designing the algorithm is to consider pixel movements, occurred from camcorder movements, and to interest with the objects in the scene. Consequently, motion blur and sobel filters are applied for picture processing. Sobel picture filter has put forward the section, having sharp colour change in the filtered picture. In other words, sobel filter makes the object edges in the picture clear. When a frame is filtered with sobel filter, object edges and other areas are readily differentiated with different brightness levels [10]. If the frame differences are computed by using sobel filtered frames, shot detection is done according to the objects in the scene. The coefficients of the sobel filter used in this work are following: [2 3 2; 1 0 -1; -2 -3 -2]. With these coefficients the flat areas are converted to dark and the object edges are converted to bright. If an object leaves from a scene or any object changes in the segment transitions, consecutive frame histograms differ. If the camera moves/ the brightness changes/ the light behind the objects changes, then the darkness levels around the object edges change slightly. If the sobel filter is used alone, these changes result in high consecutive frame differences and incorrect shot detection.

To lower the sensibility to the camcorder movements, it is foreseen that the neighbour pixels are copied into themselves with the specified weights. Hence, sharp edge values obtained by sobel filter are being softened and the effects of sections (background and object edge sections) near to the edges are investigated. To be able to meet these requirements, a motion blur filter was used before the sobel filter. The motion blur filter has the following coefficients: [2 3 2; 1 0 -1; -2 -3 -2]. In this method, due to the tracing of the movement of an object edge, videos were processed after converting the video to monochrome. The steps of filtering process are shown in Fig. 4.

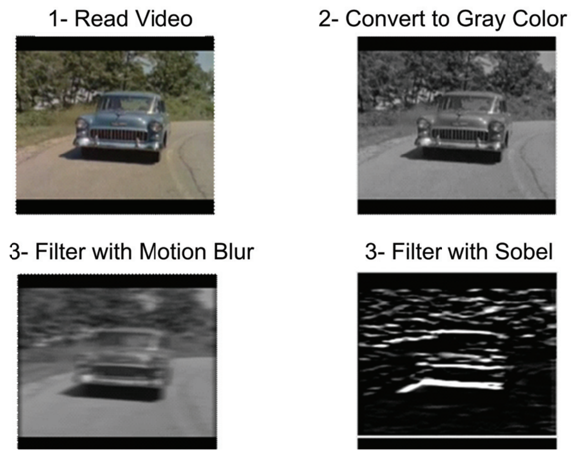

Fig. 4. Picture filtering steps

If the standard formulas of histogram comparison (Equation 1) are used to obtain the difference values of consecutive frames, the flat areas, not interested in this method, will be causing false results to increase [11].

$$D(i,i+1) = \sum_{j=0}^{255} \left| H_i(j) - H_{i+1}(j) \right|. \quad (1)$$

To remove this disadvantage, the formula (Equation 2) used in the filtered video histogram comparison method, was obtained with neglecting "0" value representing black colour in the standard histogram comparison formula. The formula in Equation 2, can be used in digital videos having an 8-bit colour darkness value (darkness values: 0-255, n=256). D represents the difference value between i and $i+1$ frames, H value represents the histograms of the corresponding frames and j value shows the darkness values. As mentioned above, to neglect the darkness values representing the flat areas in the filtered histograms, the darkness values between 1 and 255 were used

$$D(i,i+1) = \sum_{j=1}^{255} \left| H_i(j) - H_{i+1}(j) \right|. \quad (2)$$

The working steps for the Filtered Video Histogram Comparison method can be summarized as follows:

1. Making the video gray scale;
2. Filtering the video with Motion Blur Filter;
3. Filtering the video with Sobel Filter;

4. Getting the difference values of consecutive frames with histogram comparison, while neglecting the black components of the filtered video ("0" darkness value);
5. Defining the frames over the threshold value as a segment transition;
6. Making the segments generated using the defined segment transitions.

These operations are realised as shown in the following flowchart (Fig. 5).

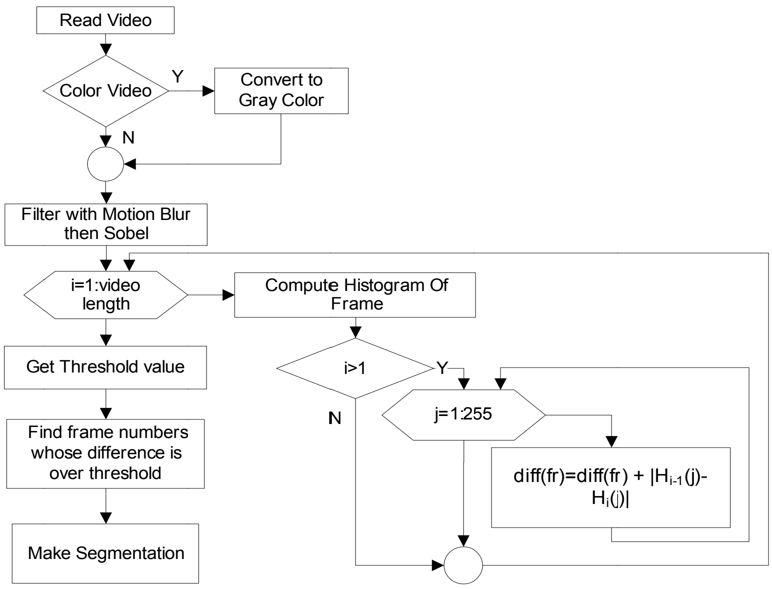

Fig. 5. Overall flowchart of proposed method

Evaluation and comparisons

When the results, obtained from the application of proposed algorithm evaluated, it is easy to say that the sensibility to camcorder movements and brightness level changes is much less compared to other uncompressed video segmentation algorithms. Test with several videos showed that especially sensibility to brightness level changes reduced much and videos including camcorder lighting changes were segmented with very high accuracy.

In Fig. 6, sequential frame differences, resulted by HSV Histogram Comparison and Filtered Video Histogram Comparison for a video including lots of brightness level changes and camcorder movements, are shown with graphics. HSV Histogram Comparison was chosen for comparing proposed algorithm because it has less sensibility in standard algorithms. Real segment transitions are shown in graphics with an arrow.

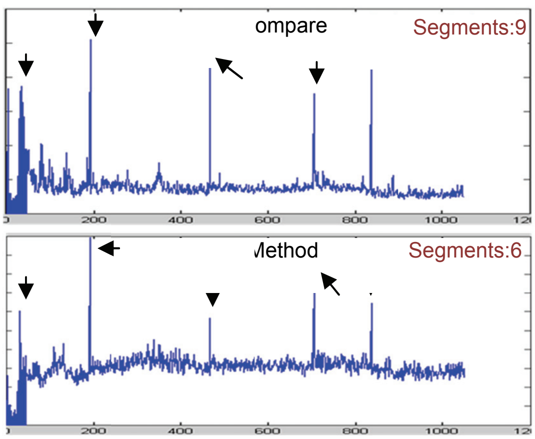

Fig. 6. Computed consecutive frame differences for As.avi video

As.avi video, used for comparison, was recorded specially with a poor camera and it has camcorder motions, brightness level changes also it consists of 6 segments.

In Fig. 7, the fourth segment of Chevrolet.avi video (between 300^{th} and 515^{th} frames) has lots of brightness level changes and camcorder motions. In the graphics, it is shown that, the differences computed with HSV Histogram Comparison algorithm for fourth segment's brightness level changes is higher than real segment transitions. In addition, HSV Histogram Comparison computed difference level for camcorder motion just after the starting of 5^{th} segment as near to real transition difference level.

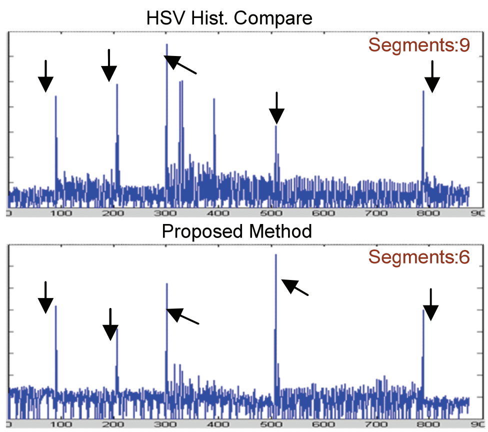

Fig. 7. Computed consecutive frame differences for Chevrolet.avi video

For measuring of the proposed algorithm, all algorithms mentioned above were applied to several videos from [12]. After getting results, it is observed that proposed algorithm has better performance than the other algorithms. In Table 1 and Table 2, comparison results for As.avi (a sample video including camera movements and lighting changes, five segments transitions) video and Chevrolet.avi [12] (including five segments transitions) video are given. In results, computing times, number of segments detected by algorithm, accuracy rate for real segment transitions, and extra (unnecessary) number of segments detected are shown. For the best performance it is expected to have minimum computing time, best accuracy rate (%100), and not to detect any other extra segments.

Table 1. Comparison results for As.avi video

ALGORITHM	Time (s)	Number of Segment	Accuracy (%)	Extra Segment
Pixel Comparison	123.73	6	80	2
Block Based Pixel Comparison	201.18	6	80	2
RGB Histogram Comparison	1.61	6	60	3
Block Based Histogram Comparison	9.32	7	60	4
HSV Histogram Comparison	48.71	8	100	3
Proposed Filtered Video Histogram Comparison	11.65	5	100	0

Table 2. Comparison results for Chevrolet.avi video

ALGORITHM	Time (s)	Number of Segment	Accuracy (%)	Extra Segment
Pixel Comparison	153.41	7	100	2
Block Based Pixel Comparison	212.23	9	80	5
RGB Histogram Comparison	2.74	8	40	6
Block Based Histogram Comparison	27,09	7	60	4
HSV Histogram Comparison	34.98	8	100	3
Proposed Filtered Video Histogram Comparison	17.67	5	100	0

For comparison of computing overhead, it is clear that filtering steps used in Filtered Video Histogram Comparison algorithm are making computation time longer. But in the computation part of the frame differences, the proposed method computes faster in rate 1/3 because of using only gray scale darkness levels. Other methods use Red, Green, Blue levels for computing consecutive frame differences. For this reason their computing times are much longer than the proposed method.

The interface shown in Fig. 8 was designed to apply the algorithms, include the proposed and existing algorithms. With this interface, a loaded video can be divided into sections according to the parsing method and then these segments can be stored for access. In the interface, the number of segments represents the segments transitions.

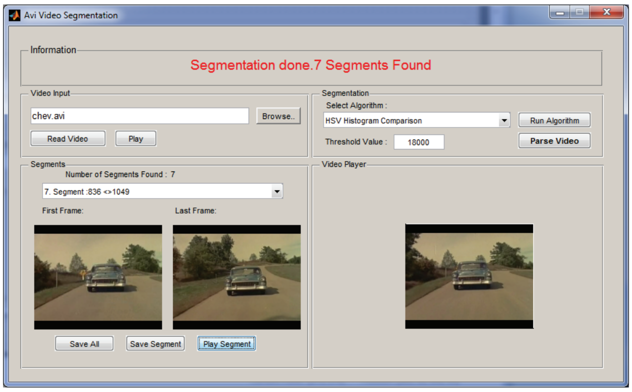

Fig. 8. Parsing interface

Conclusions

In this work, a new parsing method for uncompressed videos was proposed which has superiority over the existing parsing algorithms. Through the investigation of existing algorithms, it is found that, these algorithms have some deficiencies. The new method was applied to a number of example video streams and compared with the other algorithms. As a result of the comparison, it is seen that the proposed algorithm has a better segment transition, and an acceptable level of computational overhead.

References

1. **Cotsaces C., Nikolaidis N., Pitas I.** Video shot detection and condensed representation A review. // IEEE Signal Processing Magazine. – IEEE, 2006. – Vol. 23. – Iss. 2. – P. 28–37.
2. **Koprinska I., Carrato S.** Temporal Video Segmentation: A Survey // Signal Processing: Image Communication. – Elsevier Science, 2001. – Vol. 16. – Iss. 5. – P. 477–500.
3. **Kasturi R., Jain R.** Dynamic vision. In R. Kasturi and R. Jain, editors, Computer Vision: Principles, – IEEE Computer Society Press, Washington, 1991. – P. 469–480.
4. **Boreczky J. S., Rowe L. A.** Comparison Of Video Shot Boundary Detection Techniques // In Proceedings of Storage and Retrieval for Image and Video Databases (SPIE), 1996. – P. 170–179.
5. **Gargi U., Oswald S., Kosiba S., Devadiga S., Kasturi R.** Evaluation Of Video Sequence Indexing And Hierarchical Video Indexing // In Proceedings of Storage and Retrieval for Image and Video Databases (SPIE), 1995. – P. 144–151.
6. **Swanberg D., Shu C. F., Jain R.** Knowledge guided parsing in video databases // In Proceedings of Storage and Retrieval for Image and Video Databases (SPIE), 1993. – P. 13–24.
7. **Zhang H. J., Low C. Y., Smoliar S. W.** Video Parsing and Browsing Using Compressed Data // SPIE Conf. Image and Video Processing II, Multimedia Tools and Applications, 1994. – Vol. 1. – No. 1. – P. 89–111.
8. **Masalskis G., Navickas R.** Reverse Engineering of CMOS Integrated Circuits // Electronics and Electrical Engineering. – Kaunas: Technologija, 2008. – No. 8(88). – P. 25–28.
9. **Vyšniauskas V.** Subpixel Edge Reconstruction using Aliased Pixel Brightness // Electronics and Electrical Engineering. – Kaunas: Technologija, 2008. – No. 8(88). – P. 43–46.
10. **Adwan S., Arof H.** A New Approach for an Efficient DTW in Face Detection through Eyes Localization // Electronics and Electrical Engineering. – Kaunas: Technologija, 2011. –

No. 2(108). – P. 103–108.
11. **Zhang H. J., Kankanhalli A., Smoliar S. W.** Automatic partitioning of full–motion video, Readings in Multimedia Computing and Networking. – Morgan Kaufmann Publishers, 1993. – P. 10–28.
12. **The Open Video Project.** 1955 Chevrolet Screen Ads. Online: http://www.open-video.org/details.php?videoid=3410&surrogate=storyboard.

Received 2011 07 11
Accepted after revision 2011 10 11

H. I. Eskikurt, B. Boru. A New Method for Video Parsing of Uncompressed Digital Video Streams // Electronics and Electrical Engineering. – Kaunas: Technologija, 2011. – No. 10(116). – P. 89–94.

Today, the use of digital videos instead of analog ones has become much popular due to their easy recordability and low-cost storage. Hence, huge amount of digital video archives have come about. The meaningful search in a video has been an important demand in order to fast access the video database. While the standard picture and text searching methods cannot be used in a digital video, the video indexing has become a popular interest, and lots of researches have been done in this field. In this study, an implementation of a new video parsing algorithm design for uncompressed videos is presented. Through the investigation of the parsing algorithms in the existing literature, lacks of these algorithms were defined and a new method for achieving high performance was proposed. It is determined that the proposed method is much better than the other algorithms considering the computational overhead and the performance. Ill. 8, bibl.12, tabl. 2 (in English; abstracts in English and Lithuanian).

H. I. Eskikurt, B. Boru. Videoanalizei skirto naujo metodo taikymas nesuspaustuose skaitmeniniuose videosrautuose // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2011. – Nr. 10(116). – P. 89–94.

Skaitmeninis videomedžiaga tampa vis populiaresnė, palyginti su analogine, dėl saugojimo ir įrašymo paprastumo. Besiplečiančiose videoinformacijos saugyklose reikia užtikrinti tinkamą ir sparčią paiešką. Šiuo atveju negalima pritaikyti standartinių paveikslų ir teksto paieškos metodų. Todėl populiarėja video- informacijos indeksavimas. Pasiūlytas naujas metodas nesuspaustiems skaitmeniniams videosrautams tirti. Atsižvelgiant į gautus rezultatus nustatyta, kad metodas yra našesnis nei kiti esami. Il. 8, bibl. 12, lent. 2 (anglų kalba; santraukos anglų ir lietuvių k.).