A Fast Recognition Algorithm for Detecting Common Broadcasting Faults

GOU Chao, WANG KunFeng
State Key Laboratory of Management and Control for Complex Systems
Institute of Automation, Chinese Academy of Science
Beijing, China

HU Bin, XIONG Gang, ZHU Fenghua
Beijing Engineering Research Center of Intelligent Systems and Technology
Institute of Automation, Chinese Academy of Science
Beijing, China

***Abstract*—The video before it is been broadcast should be detected to find out whether it has broadcasting faults. Broadcasting fault mainly refers to black field, static frame, color bar, mute, volume excess. In this paper, we proposed a fast recognition algorithm based on MATLAB to detect common broadcasting faults. We used M language to complete a software platform which can identify the static frame, black fields, color bar in the video sequence, and the mute, volume excess in the audio sequence. Experimental evaluation shows that our approach significantly performs well in the automatic broadcasting faults detection.**

Keywords—MATLAB, Broadcasting fault, Audio/Video detection

I. INTRODUCTION

With the development of modern communication technology and computer technology, automatically broadcasting equipment has a increasing degree of reliability and stability. While audio and video programs may go wrong at any time due to some operational errors or equipment failures. Therefore, it is necessary to develop a system to detect the broadcasting faults. This system can detect audio and video programming problems before been broadcast.

Correct detection for broadcasting faults plays a very important role in the broadcasting of digital video. It determines the ability to achieve television programs' smooth broadcast and it occupies a very important position in the entire technology.

For example, researches are carried out on the common methods of video quality monitoring and a solution is proposed for "*MegaEye*" system video quality monitoring. It analyzes image quality on the video end and catches the occurrences of black field, static frame and other faults [1]. Nowadays, the main trends for broadcasting fault monitoring are digitized, networked, intelligent and flexible.

II. SYSTEM FUNCTIONS

This system is developed to detect audio and video signal broadcasting faults based on digital image and audio signal processing under MAGLAB GUI environment.

Video with format of *avi* can be read. And each frame of the video is extracted and stored as format of *jpg* in the memory of the PCs. This system can detect black field, static frame and color bar. Furthermore, it can identify the fault location.

In addition, audio with format of *wav* can be read. Meanwhile it can draw the digital audio signal waveform and detect mute and volume excess fault. Furthermore, it can identify the fault location.

A. Video Detection Modules of This System

The video detection modules are mainly based on a MATLAB digital image processing. The difficulty is the implementation of algorithm. In order to identify the black field, static frame, color bar, we firstly extract the video frame images. At the same time, in order to reduce the amount of computation, true-color images are converted to grayscale and named by a serial of numbers stored in the memory of PCs. The basic detection algorithm is that the elements in the gray-scale image matrix of the black field are less than a certain threshold. It is the static frame when three consecutive frames are identical. It is the color bars when the differences of any two rows in the grayscale matrix of each frame are less than a certain threshold. Video Sequence Detection flowchart is shown in Figure 1-1.

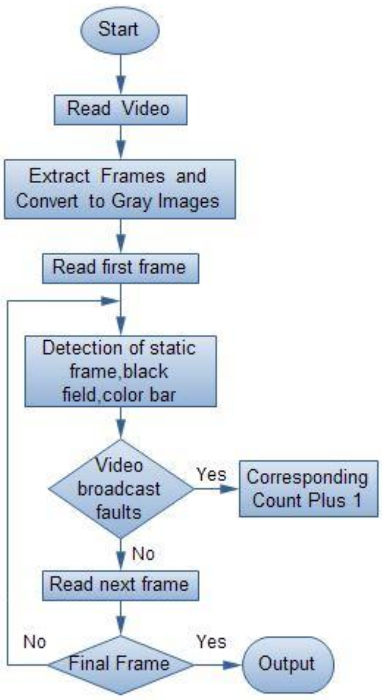

Figure 1-1 Video detection flowchart

978-1-4799-0530-0/13/$31.00 ©2013 IEEE

B. *Audio Detection Module of This System*

The audio detection modules are based on a MATLAB digital signal processing. The audio signal is time-varying, but in short period (usually within 10-20 ms), its basic characteristics are relatively stable. Thus the short-term spectral characteristics are important methods for analyzing audio signal. The approach we proposed is able to identify the sequence of an audio signal in mute and volume excess. At first, we should artificially make sub-frame processing to the audio signals. The frame length is 20ms and the frame shift is 10ms. With the threshold set before, we detect the speech state by calculating the short-term energy and zero-crossing rate of audio signal. The audio signal detection flowchart is shown in Figure 1-2.

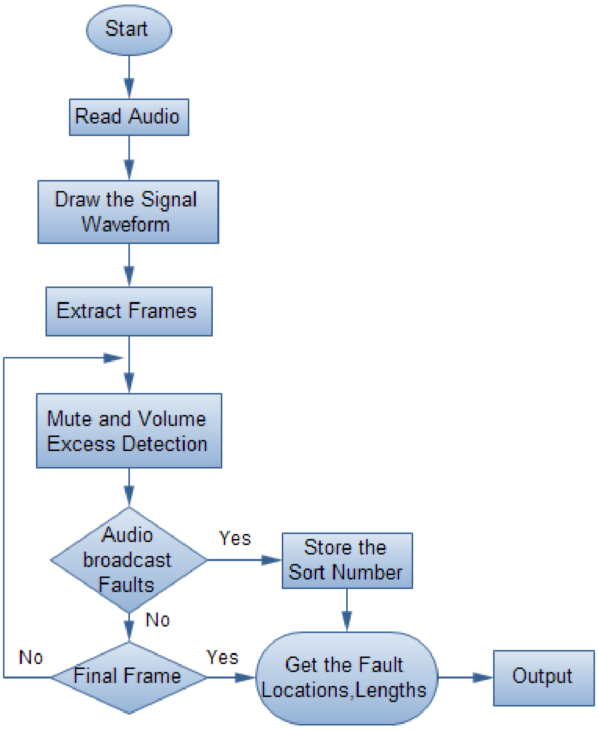

Figure 1-2 Audio detection flowchart

III. REALIZATION OF THE SYSTEM

A. *Video Detection*

In image processing, we often study two-dimensional image as a two-dimensional matrix. The MATLAB based on matrix operations can be naturally extended to the field of image processing applications. In the video broadcasting fault detection, the key research will be on the detection of image frames. Aimed at fault detection for the video broadcasting, we proposed this approach. Firstly, the video is read into MATLAB. Then each frame of the *avi* video sequence is extracted. The true color image is converted to two dimensional gray-scale images to reduce the amount of calculation. In order to facilitate the storage and read each image frame easily, we save the *jpg* images by the numbering sequence of 1,2,3,4….Thirdly, we read the gray images into the memory of MATLAB do some correlation detection.

1) *Detection of Black Field*

The black filed is plain black and the value of image matrix read in MATLAB will be less than a certain threshold. In order to ensure the accuracy of detection, the existing detection algorithms always set a threshold value and compare it to all the elements of image matrix[2]. Each element of the image matrix is less than a threshold and then this image will be considered to black field. These algorithms waste a lot of time on comparison because most of the frames are not black fields.

We propose a new approach which can significantly improve the detection speed. The basic idea is to detect in blocks. The image is divided into 8x8 pixels of blocks out of the frame. It comes to the next frame as long as the average value of the block is greater than the threshold value set in the detection. But if the value is less than the threshold value, it comes to the next block and the flag plus one. The comparison lasts until the block of image is the last piece which is less than the threshold value .Then it can be determined that the frame is a black field. This reduces the complexity of computation and significantly increases the detection efficiency and accuracy. Figure 2-1a and Figure 2-1b can be detected correctly when we do the evaluation.

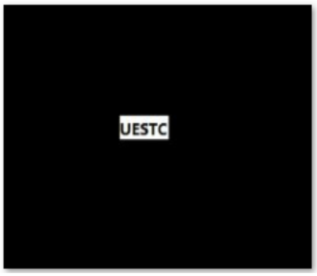

Figure 2-1a Not black field

Figure 2 -1b Black field

2) *Detection of Color Bar*

The color bar’s colors from left to right are white, yellow, cyan, green, purple, red, blue and black. It includes the three primary colors of light (red, green, blue). In addition, it has complementary color (cyan, purple, yellow) and non-color (black and white). Color bar signal waveform is simple because the values of each column in the gray image matrix are almost the same. It jumps out of the loop calculation and comes to the next frame as long as the differences of two rows are greater than the threshold set before. This will reduce the computing and improve the detection speed due to most of the video sequence image frame are not color bars.

3) *Detection of Static Frame*

Static frames are inter-frame detection. So it needs to compare several consecutive frames to decide whether the current frame is static frame. We should firstly do black field detection due to continuous black frame is not static frame. Then we can do static frame detection after the judgment of black frame. The basic idea of the static frame detection is that do difference between every two successive frames. Only when the differences are smaller than the threshold the third frame is considered as a static frame. The detection algorithm is shown in Figure 2-2.

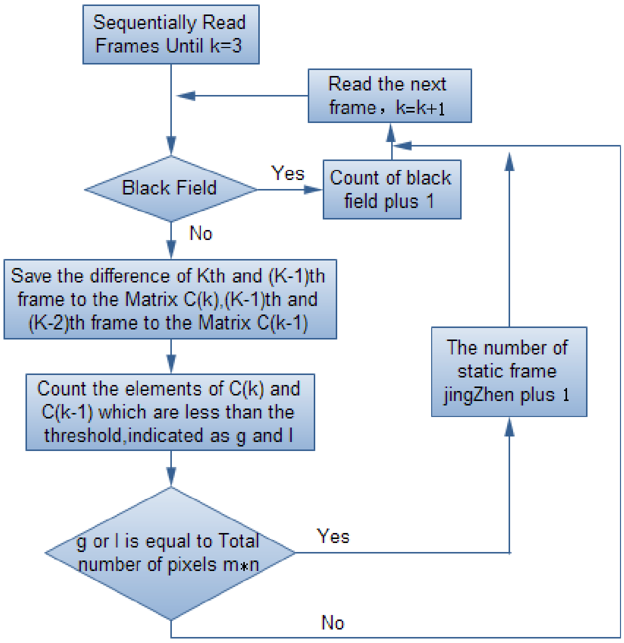

Figure 2-2 The Algorithm for Static Frame Detection

B. Audio Detection

The audio broadcast faults are mute and volume excess. Within a short period of time (usually in a short time of 10-30ms), the basic characteristics of audio remain unchanged. So this can be seen as a quasi-steady-state process. The voice processing in MATLAB toolbox *Voicebox* can achieve the fast processing of the speech signal [3]. This article uses the zero-crossing rate and the short-term energy to do the detection [4][5].

The zero-crossing rate indicates the number of a voice signal waveform passes through the zero level. For discrete signals, it is called zero-crossing if the adjacent sampling values change signs. Zero-crossing rate is the times of changing signs. For the audio signal x (n), the short zero-crossing rate of the formulation [6]:

$$Zn = \frac{1}{2}\sum_{m=-\infty}^{\infty} |\operatorname{sgn}[x(m)] - \operatorname{sgn}[x(m-1)]|\, w(n-m)$$

$$= \frac{1}{2}\sum_{m=n}^{n+N-1} |\operatorname{sgn}[x_w(m)] - \operatorname{sgn}[x_w(m-1)]| \qquad (1)$$

$$\text{subject to} \quad \operatorname{sgn}[x(n)] = \begin{cases} 1, & x(n) > 0 \\ -1, & x(n) \le 0 \end{cases}$$

Short-term energy En is the energy of the n-th point. Short-term energy can be seen as the square of the signal through the output of a linear filter. It is calculated as follows:

$$En = \sum_{m=-\infty}^{\infty} [x(m)w(n-m)]^2 \qquad (2)$$

1) Detection of mute

For the short-term energy of some syllables is not so high, using a single short-term energy to detect the mute may bring up misjudgment. In order to improve the detection accuracy, the mute detection algorithm uses short-term energy and zero-crossing rate to detect the status of the voice signal. The mute detection system flowchart is shown in Figure 2-3.

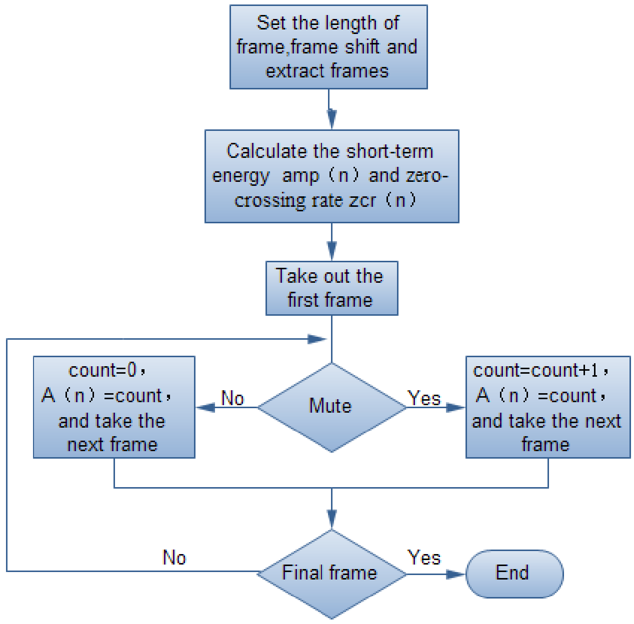

Figure 2-3 Mute Detection Flowchart

2) Detection of volume excess

The exceeded volume's voltage is significantly higher than general audio signal. Our system uses short-term energy to detect volume excess. We can predicate that it is exceeded when the energy is greater than the threshold set beforehand. The specific implementation method is that we firstly find the serial number of frame which is higher than the threshold. It is defined as the beginning of excess when the number of span is less than 300, otherwise it is the beginning of another excess.

C. Location of Audio and Video Faults

It needs to find the where the video/audio broadcasting faults emerge and how long the video/audio broadcasting faults last after they are detected.

The basic idea of this system to find the location of video faults is to read the video by using the *mmreader* function which can get the rate of frames. This rate, named as *Fs*, means that *Fs* frames are played per second. The number of *K* is the sort number of the extracted video frames. When the black field, static frame or the color bar firstly appear, the sort number *k* is stored. As a result, the quotient of k/Fs is the location which is measured by seconds.

A matrix is used to store the mute frame sequence. The number of the mute is equal to the number which is up to more than 100. When it adds to 100, the sort number subtracts 100 and multiply the length of a frame shift named as *inc*. As a result, we can get the locations after it is divided by the rate of sampling named as *fs*. For the volume excess locations and excessive number of times, it is considered to be a volume excess when the interval is less than 300. Otherwise another excess starts, and then the number plus 1.Meanwhile we should find the frame of the short-term energy which exceeds the threshold value to get the locations and the number of excess.

IV. Tests on This System

A. *Tests on Video Detection*

Now we can do some experimental evaluations. Click *Read Video* button to choose the test source named as *test-Iron*, and then click the *Play Video* button as shown in Figure 3-1. The results are shown in Figure 3-1 after clicking *Start Video* Detection. In order to verify its correctness, we return to the storage of PC and find out the video frame sequence of images are stored by 24 black field frames, 26 static frames and 45 color bars. Furthermore we can get the *avi* video properties to find the frame rate is 18. At last we can calculate the location of the black field is at about 5.3 and lasts for about 1.3 seconds, static frames at 11.9 seconds and lasts for about 1.4 seconds, color bars at 23.3 seconds and lasts for about 2.5 seconds. It is the same as the result shown in figure 3-1.

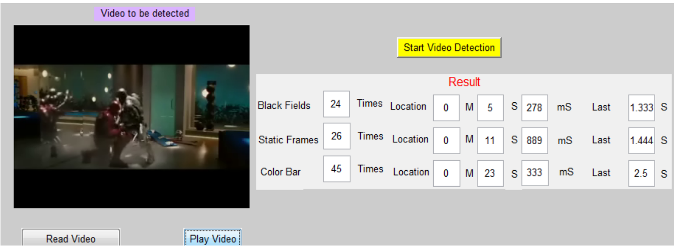

Figure 3-1 Video Detection

B. *Tests on Audio Detection*

Click the *Read Audio* button and choose the test audio named as *11.wav*.Then we can see the audio signal waveform shown in Figure 3-2. Click the *Play Audio* button and we can hear the *wav* audio played. Click *Start Audio Detection* and we can get the results as shown in Figure 3-3.We verify its accuracy by comparing the waveform with the output.

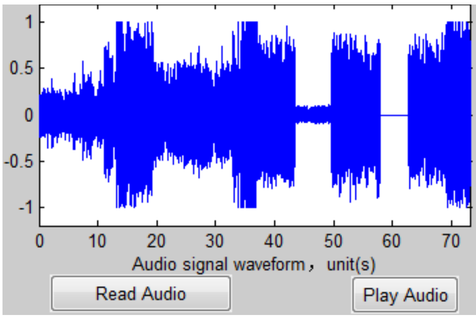

Figure 3-2 Audio signal waveform

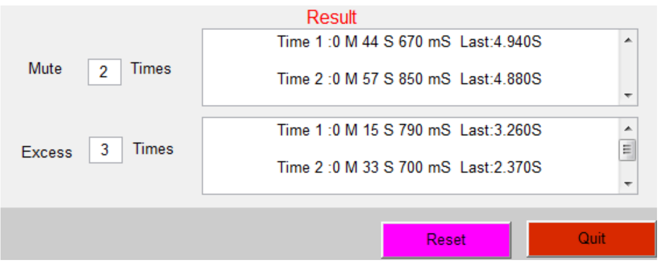

Figure 3-3 Result of audio detection

V. Conclusion

With the rapid development of digital image and digital signal processing technologies, there are more and more audio and video broadcasting faults detection systems or products. But there is little research on MATLAB-based broadcasting fault identification algorithm. I believe there will be more and more research in this area due to MATLAB powerful matrix calculation function.

We use MATLAB GUI to design the user interface shown in Figure 4-1. In this paper, we proposed a fast recognition algorithm to detect common broadcasting faults. Thorough quantitative evaluation has shown that the proposed algorithm performs well. In future work, the speed can be further improved by making use of GPU. Highly parallel structure of GPUs makes them more effective than general-purpose CPUs for algorithms where processing of large blocks of data is done in parallel.

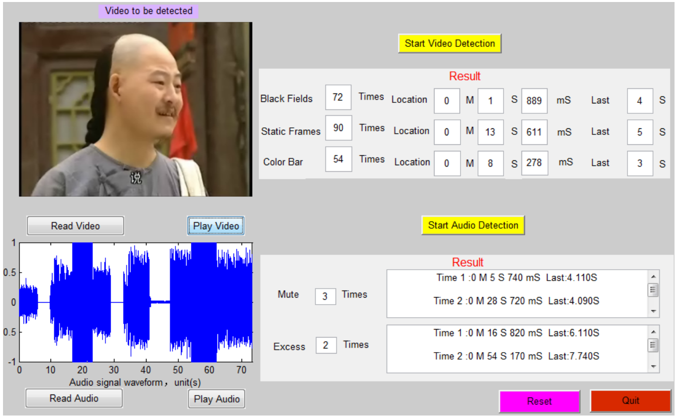

Figure 4-1 User interface

Acknowledgment

This work was supported in part by National Natural Science Foundation of China projects 71232006, 61233001, 90920305, 61174172, 61104054 and 61203079, 61203166; CAS projects 2F11N01, 2F11N07 and 2F12E05. We would like to express our sincere gratitude to Prof. Fei-Yue Wang for his guidance on this project.

References

[1] Zhong J, Hu Y, Zheng M. MegaEye Video Monitoring and Control System Service Quality Monitoring Study [J]. Computer Applications and Software, 2011, 28(9).

[2] Zuyu Sun . Design and Implementation of Detecting the Black Field and Static Frame [J]. Advanced Television Engineering, 2010 (012): 122-123.

[3] M. Bosi and RE Goldberg, Introduction to Digital Audio Coding and Standards, Kluwer Academic Publishers, 2003:8-12

[4] Dawei Xu,Bian Wu.A Real-time Voice Endpoint Detection Algorithm in A Noisy Environment [J].Computer Engineering and Applications，2007，(1)： 115 – 117.

[5] R. Cai, L. Lu, and H.-J. Zhang, “Highlight sound effects detection in audio stream,”Proc. Of ICME,2003:37-40

[6] Xiongwei Zhang， Liang Chen. Modern Technology of Voice Processing and Applications [M]. Machinery Industry Press.2003：30－31.