International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

IJIEMR Transactions, online available on 12th July 2021.
Link:https://ijiemr.org/downloads/Volume-10/Issue-07

DOI: 10.48047/IJIEMR/V10/I07/11

Title: **ALGORITHM FOR COMPARING AUDIO FRAMES**

Volume 10, Issue 07, Pages: 48-54

Paper Authors: **Saida Safibullayevna Beknazarova, Fozilov Firdavs Dilshodo'gli, Talipova Ozoda Xabirovna**

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

ALGORITHM FOR COMPARING AUDIO FRAMES

Saida Safibullayevna Beknazarova
Tashkent University of Information Technologies named after Muhammad Al- Khwarizmi, 105, A. Temur, Tashkent, 100142, Uzbekistan
Fozilov Firdavs Dilshodo'gli
Student of Tashkent University of Information Technologies named after Muhammad Khwarizmi
Talipova Ozoda Xabirovna
Teacher of Tashkent University of Information Technologies named after Muhammad Khwarizmi

Abstract: In the modern world, television, as a mass media, plays a very important role in the life of every person, because it allows you to convey information to him from almost any part of the world. In this regard, one of the most urgent tasks in the field of audio-video data processing is the development and improvement of methods for compressing audio-video data, taking into account the elimination of temporary redundancy of TV images and audio accompaniment. This problem is very relevant in the conditions of the global financial crisis, in conditions of limited frequency resources. In addition, it becomes possible to significantly reduce the time of preparing television reports for broadcast directly from the event sites by transmitting signals from TV cameras directly to the installation hardware of television centers over cellular networks, and the need to use expensive and not always available broadband communication channels disappears.

Keywords: algorithms, splitting, audio file, frames, video images.

Introduction

As a result of the conducted research, it was found that when compressing audio files with MP3, AAC, OGG codecs, the best quality/volume ratio (data volume) is provided by the MP3 standard [1-3]. At the same time, the main compression in these codecs is performed on the basis of psychoacoustic signal processing, which removes from the stream those components that the human ear does not perceive. However, with high compression coefficients, signal losses significantly increase, which creates distortions of the restored sound in the form of clicks, noises, loss of high frequencies, rustles. A good sound quality of the above formats is provided at an output stream speed of 128-256 kbit / s. At lower speeds, it drops noticeably.

Methodology

The results of the analysis showed that these methods of audio signal compression do not take into account the temporal repeatability of individual fragments of the audio signal

Algorithm for comparing audio frames.The block diagram of the algorithm for comparing audio frames for identity is shown in Figure 1 This procedure is

designed to find similar audio frames, and if such audio frames are found, they are assigned a pointer to the coordinates of their location in the data array. This algorithm works as follows.

Results

Using the obtained values of the segmentation algorithm, the obtained data is considered as audio frames. For example,

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Figure 2 shows the counts of the first and second audio frames, where 1.1, 1.2, ... are the counts of the first, and 2.1, 2.2, ... are the counts of the second audio frames.

In block #3, a processing cycle is created for an array of audio frame samples, in which the first count (1.1) of the current I frame is checked by block #4, according to the condition: whether this count was processed or not. If the count was not processed, a new cycle is created in block 5, where the count is checked for the early processing of the samples of the next audio frame (frame II) in block 6. If it was not used, then a check is made in block 7 using the isEqual function, which takes the difference between the samples of the first and second audio frame and compares it with a given error, which can be set from 0 to 256 (1.1-2.1. < 256). If the difference is less than this error, then these samples are considered equal, and the remaining samples of these audio frames are checked until the end of the cycle.

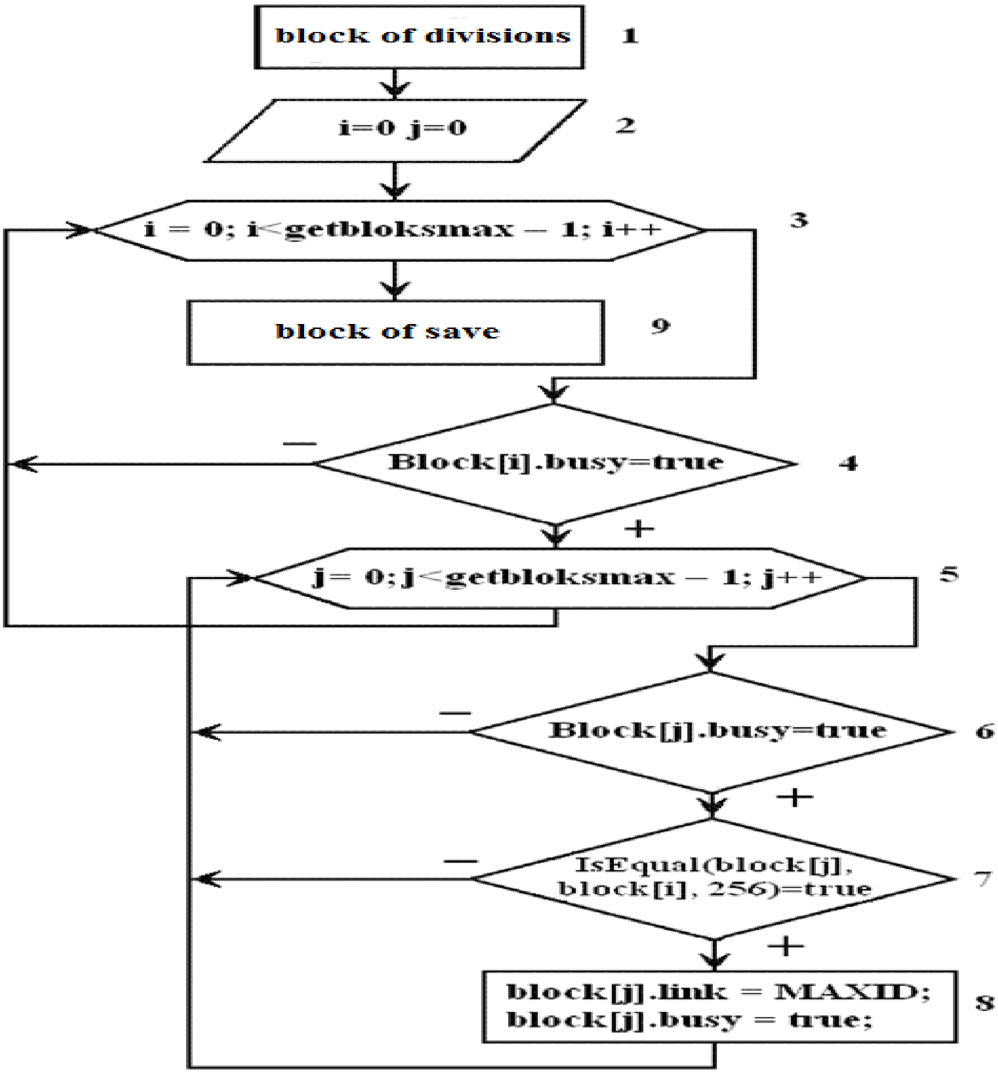

Fig. 1. The algorithm for comparing audio frames..

If all the samples are equal, then these frames are considered as equivalent frames, and the first frame will be considered a reference frame, the second-similar to it. In this case, the loss coefficient for each difference is calculated as follows:

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Error = Difference/Max. Amp.*100% ≈ 0...10% where Max. Amp.is the maximum amplitude of the current audio file.

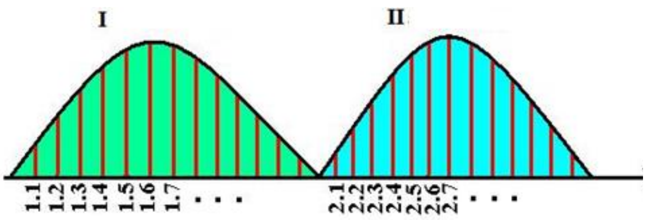

Fig. 2 Time representation of samples of adjacent audio frames.

Significant losses can occur when the samples themselves are small, this can lead to the appearance of noise of small amplitudes.

If the audio frames are equal after checking in block #7, then a reference to the coordinates of the second frame in the stream is saved, where the reference audio frame will be written during recovery, and a similar (repeated) audio frame is marked as already used in block # 8.

After the loop compares all the frames with each other, control is transferred to the block for saving and generating the output file (block #9).

Discussion

Algorithm for saving a compressed file.Figure 6 shows the algorithm for generating the output stream with the previously obtained results. This procedure is designed to save the results obtained from previous algorithms and generate an output stream with compression of the received files using the long series algorithm (RLE). This algorithm works as follows.

In block #2, new variables are declared and new data streams are created, designed to read and write data from and to a file, respectively.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

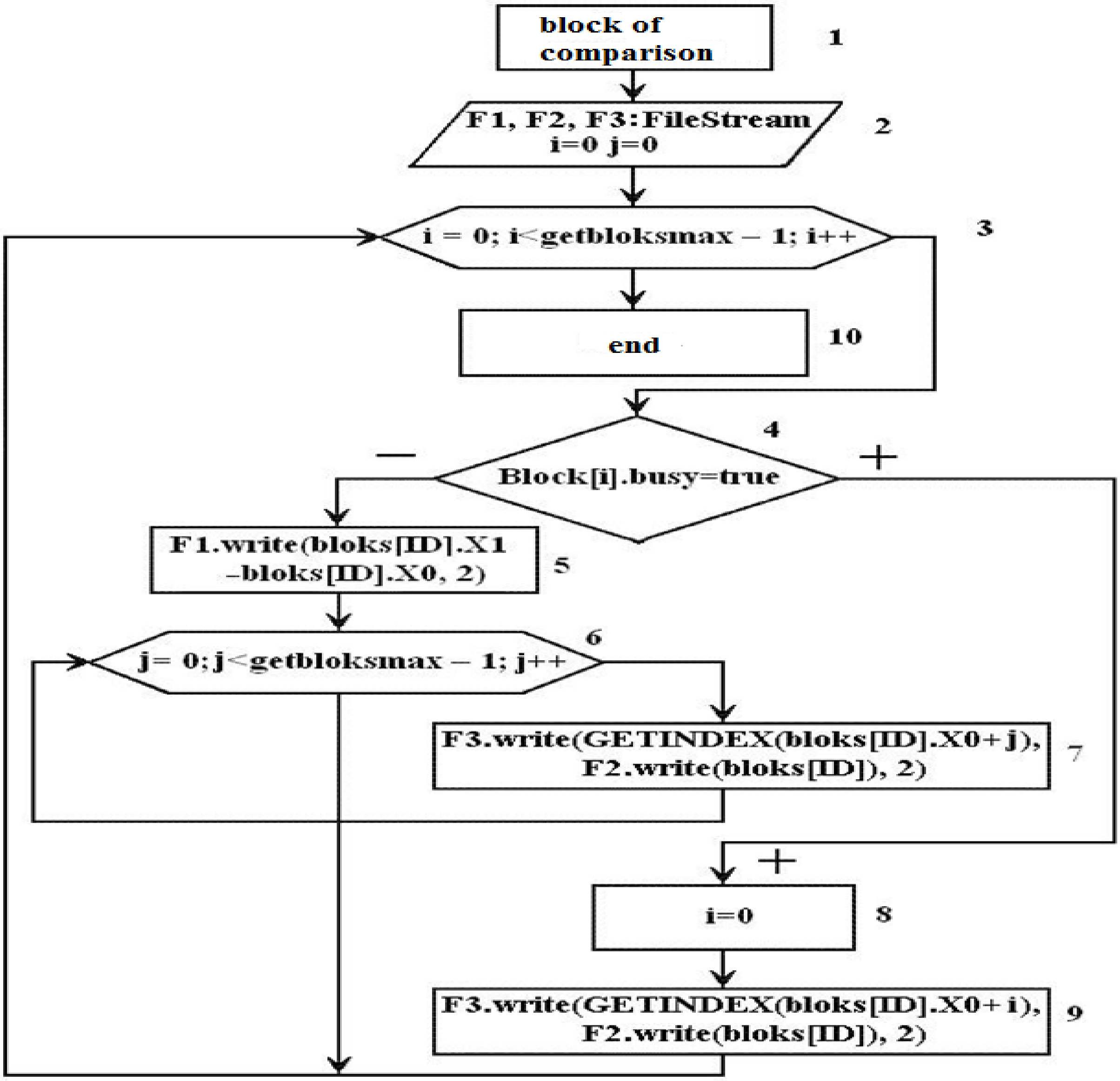

Fig.3. The algorithm for saving the obtained results.

Next, a loop is created in block No. 3, which reads the audio frames in a count-by-count manner. In block No. 4, each sample read using the cycle coefficient is checked whether it has a reference to repeatability or not, i.e. audio frames that have repetitions are selected in this block. Depending on whether this audiocar repetition or not, the process proceeds to either save audiocode with its replication factor (unit 5, 6, 7), or audiocar remain without recurrence (unit 8, 9).

After all audiometry to test the repeatability, the output to be four files:

the 1st file is a wawheader containing 44 bytes of the waw file header information.

the 2nd file is block1, in which the samples of non-repeating and reference audio frames are sequentially recorded.

the 3rd file is a headscript, where the sizes of audio frames (borders) stored in the block1 file are stored

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

the 4th file-filemap – the frequency of audio frames. It has the following structure:

the total number of audio frames is recorded in the first 6 bytes;

next, 4 bytes are allocated in memory to store the audio frame index, i.e. where the audio frame will be recorded during recovery (for example, if the audio frame coefficient is 6, then it will be recorded as the sixth in the restored file).

After receiving all four files, additional compression will be performed using the long series algorithm (RLE), according to Figure 3and to increase the compression ratio, you can use the Huffman algorithm, which redistributes the required number of bits in the output stream. I.e., code combinations that occur frequently in the stream are encoded with short codes, and those that are rare are long. As a result of using the Huffman algorithm, the compression efficiency usually increases by 20-25%.

Table 1.

Results of compression of audio files with a fractal codec

Number of the song	Sourcesize (wav), bytes	Compression ratio, once for an error of 1%	Compression ratio, once for an error of 5%	Compression ratio, once for an error of 10%
1	43 186 206	2,3	4,5	6,3
2	34 993 248	1,8	3,4	4,5
3	36 965 458	1,9	2,8	4,2
4	37 647 466	1,7	2,7	5,4
5	30 873 714	1,8	3	6
6	23 040 030	2	4,5	8,2
7	44 550 236	1,6	3,2	5,1
8	76 763 180	6,5	9,2	16,8
9	24 957 468	12,7	25,1	37,8

Conclusion

Experimental evaluation of the efficiency of compression of audio files by fractal and fractal-spectral codec. To evaluate the effectiveness of the proposed method of audio signal compression based on the elimination of temporary redundancy of audio frames, an experimental study of the compression of audio files of various genres with various errors in the identification of audio frames was conducted. The results of the study are summarized in Table 1 and presented in the form of histograms in Fig. 7. [14].

As can be seen from the above data, the highest compression ratio was obtained when processing on the rhythmic melody 9, in which there are many repeating musical fragments that provide good compression of the stream.

However, on other compositions, the compression ratio is relatively low and at 10% the identification error is on average 4-6 times.

Currently, Haar wavelets are widely used for information compression, which is characterized by simplicity of implementation, since it has only 2 coefficients. However, the Haar wavelet is not very suitable for compressing audio signals, because it does not provide a high degree of compression of the ZS, since when a large number of conversion coefficients are discarded, distortions occur in the form of extraneous noise, crackling and rumbling. To eliminate this disadvantage, higher-order wavelets can be used, for example, Daubeshi-4th order, having 4 coefficients and Daubeshi-10, having 10 coefficients [4,34]. Moreover, the functions of higher-order wavelets have a more "smooth" shape, due to which the compression ratio can be increased while maintaining the sound quality. Therefore, for the implementation of the compression algorithm, it is most advisable to use the Dobshy wavelet of the 10th order, since on the one hand, it provides greater conversion accuracy, and on the other, it does not significantly reduce the processing speed.

References

1. Artyushenko V. M., Shelukhin O. I., Afonin M. Yu. "Digital compression of video information and sound" I.: Moscow 2003, p. 430
2. D. Vatolin, A. Ratushnyak, M. Smirnov, V. Yukin "Data compression methods" http://compression.graphicon.ru/;
3. D. Salomon. "Data, image and sound compression". // Moscow: Technosphere, 2004 (translated from the English by V. V. Chepyzhov).
4. V. I. Vorobyov, V. G. Gribunin. "Theory and practice of the wavelet transform". / / St. Petersburg: VUS, 1999
5. Boriskevich A. A. Electronic educational and methodological complex on the discipline "Digital speech and image processing", Minsk 2007. 124 p.
6. K. Onthriar, K.K. Loo, Z. Xue. Performance Comparison of Emerging Dirac Video Codec with H.264/AVC, International Conference on Digital Telecommunication (ICDT'06), 2006
7. Beknazarova S., MukhamadiyevA.Sh. JaumitbayevaM.K.Processing color images, brightness and color conversion//International Conference on Information Science and Communications Technologies ICISCT 2019 Applications, Trends and Opportunities. Tashkent 2019N. Sedova, V. Sedov, R. Bazhenov, A. Karavka, S.Beknazarova. Automated Stationary Obstacle Avoidance When Navigating a Marine Craft //2019 International Multi-Conference on Engineering, Computer and Information Sciences, SIBIRCON 2019; Novosibirsk; Russian Federation; 21 October 2019
8. Beknazarova S., MukhamadiyevA.Sh. JaumitbayevaM.K.Processing color images, brightness and color conversion// International Conference on Information Science and Communications Technologies ICISCT 2019 Applications, Trends and Opportunities. Tashkent 2019.
9. ZitovaB.,FlusserJ. Image registration methods: a survey // Image and Vision Computing. - 2003. - V.21. - P. 977-1000.
10. Gonzalez R., Woods R. Digital image processing / Transl. from English-M.: Technosphere, 2006 — - 1070 p.
11. Guzman I. S. et al. Digital image processing in information systems-Novosibirsk: NSTU, 2000. - 156 p.
12. Durand F. and Dorsey J. Fast Bilateral Filtering for the Display of High-Dynamic-Range Images. In John Hughes, editor,

SIGGRAPH 2002 Conference Graphics Proceedings, Annual Conference Series // ACM Press/ACM SIGGRAPH 2002 — P. 257—265.

13. Hirschmiiller //., Gehrig S. Stereo Matching in the Presence of Sub-Pixel Calibration Errors // IEEE Conf. on Computer Vision and Pattern Recognition. Miami, Florida, USA. - 2009. - P. 437-444.
14. Hirschmiiller H. Accurate and efficient stereo processing by semi-global matching and mutual information // IEEE Conf. on Computer Vision and Pattern Recognition. San Diego, California, USA. - 2005. - V. 2. - P. 807-814