Some algorithms of videosequences compression and their investigation

Tagir Waleev, Feodor Murzin
Institute of Informatics Systems, Russian Academy of Sciences, Siberian Branch, Novosibirsk, Russia
lany@ngs.ru, murzin@iis.nsk.su

Abstract

In the article, the experiments on perfecting and creation of several variants of videocodecs and also their comparative testing are described.

Keywords: *Image Processing, Video Compression, Videocodec, Bitrate.*

1. INTRODUCTION

For a compression of videosequences [1], different algorithms are applied: basing on wavelet transforms; MPEG2-like, using discrete cosine transform; interpolation algorithms, in which fragments of the luminosity function are approximated by some surfaces, fractal algorithms etc. Note that a title of algorithm reflects, as a rule, a central idea of the algorithm. Actually, in each technology of video coding, for example, in MPEG4 using wavelets, additionally the large complex of algorithms of compression can be used: RLE - method, different variants of LZW-algorithms, Huffman method, arithmetical coding, special methods of approximation of curves and surfaces, methods taken from cryptography.

The areas of an application of video compression are given below.

- Cable TV;
- Videotelephony;
- Videoplayers.

Within the framework of the given work, using experience of contacts with the foreign customers, mostly Korean and Japanese ones, the experiments on perfecting and creation of several variants of videocodecs and also their comparative testing are carried out.

In the current time, two purposes are considered.

- Creation of videocodecs for high bitrates (1-3 Mbit/sec) having a property of a minimal possible difference of selected frames of a videosequence from their parent versions in the PSNR metric.
- Creation of videocodecs for very low bitrates (160-180 Kbit/sec) having a more or less moderate reduction of quality and sufficiently fast.

The input data represent a sequence of not compressed images, in the form of AVI-file, i.e. a sequence of images in the BMP format of a size up to 720×480.

Output data represent the video-file of our original format, which in particular can be converted back into AVI-file by our programs to obtain the possibility to compare them with source files. The designed algorithms allow to realize compatibility with the majority of existing standards: NTSC, PAL, SECAM, DVB, DBS, DSS and other.

Three videocodecs which are currently developing by our group are described in this paper.

2. QVC CODEC

Videocodec Qvc is using MPEG2 - like technology supplemented by our last original algorithms. The time of compression approximately twice is more than a decompression time. On the computer with the processor of a Pentium - 3/1.2GHz the compression is carried out in a real-time, and the decompression is fulfilled more, than in 2 times faster, that is already surplus. The algorithm is easy paralleled, that is convenient for a hardware support. The videocodec *Qvc* gives a rather good quality of video even up to 180 Kbps.

In the Figure 1 the graph is given demonstrating the frame-by-frame difference in the L_2 metric between a source file and the AVI-file restored after compression by *QVC* videocodec and by the popular MPEG4-like *XviD* codec.

At an identical bitrate 1 Mbit/sec, in a case when the moving of objects on the image is not too fast, we obtain that *Qvc* gives in the metric L_2 on the average twice smaller difference between a compressed and an original image. In case of fast moving, *Qvc* anticipates *XviD* not always, but in general they are comparable.

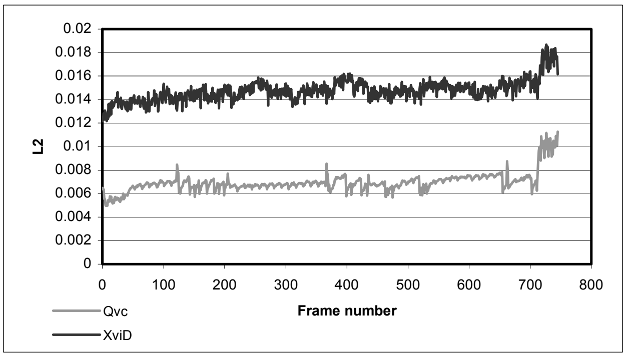

Figure 1: Difference between source frames and compressed frames in the L_2 metric for *Qvc* and *XviD* videocodecs

Below the analogous graph is given showing the frame-by-frame difference in PSNR metric.

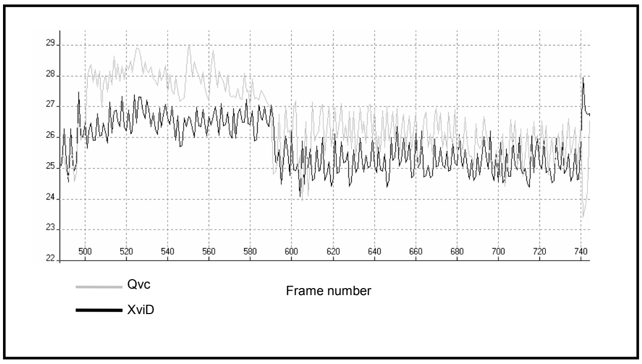

Figure 2: Difference between source frames and compressed frames in the PSNR for *Qvc* and *XviD* videocodecs

3. PTV CODEC

Videocodec Ptv is based on some original scaling transformation. Rarefying or using another way we can reduce the given initial frame I_0 twice in vertical and horizontal directions and get frame I_1 of smaller size (see Figure 3). After application of a scaling algorithm, we will obtain an image of the same size. In this image will be close (near) to initial one, but more smooth, dispossessed of a high-frequency component, which can be effectively compressed. The difference of the initial and smoothed images R_0 can be quantized and than a sequence of methods RLE, LZW, Huffman and so on can be applied, or it can be divided onto square blocks (for example, 8×8). Then each block can be compressed by any method.

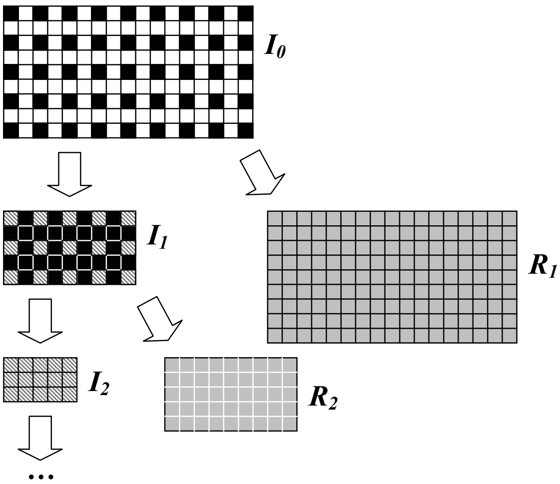

Figure 3: Scaling transformation

After this the procedure can be repeated recursively for I_1 to get a more small image and so on (see Fig. 3). Thus we will obtain the possibility to store a very small (in principle, 1×1 pixel) image I_n and a set of efficiently compressed residuals R_i. Thereby a decompression will get I_n, upscale it, decompress R_{n-1}, and combine them to get I^*_{n-1} image which approximates the previous image. Repeating this step several times we will get I^*_0, which approximates the initial image image I_0.

Different variants of this algorithm was realized by Ivanov M.A. [2]. The algorithms allow to obtain a reasonable quality at a compression of videosequences of a size 720x480 of 24-bit colour for a bitrate up to 600-650 kbps.

But in the current time, the purpose is considered of adapting the given method for a case of high bitrates. Therefore in the main different more thin variants of compression of originating differences are researched.

4. IVC CODEC

Videocodec Ivc is based on using the technology of interpolation of fragments of the luminosity function by surfaces of a special form supplemented by algorithms borrowed from a cryptography. It is intended for very low bitrates (160-180 Kbit/sec) in a real time on rather slow processors and has more - less moderate reduction of quality.

Shortly we will describe the scheme of the algorithm. First an initial image is divided into a set of large not intersected rectangular blocks. In each block, it is checked up, whether it is possible to approximate the function of luminosity by a surface of some form according to the current bitrate and quality settings.

If it is possible, this fact is stored in an appropriate mask, and then only the parameters describing the given surface are stored.

If it appears possible then the information about this is placed in the corresponding mask. Otherwise the appropriate block is divided into more small-sized blocks, usually 4 blocks, and the given procedure is applied to more small-sized blocks.

The data originating in this recursive process are disposed in a special tree-like structure which will be used for decompression. In addition the obtained tree-like structure is compressed with the help of algorithms borrowed from cryptography.

Figure 4 demonstrates an example of such block structure of source frame.

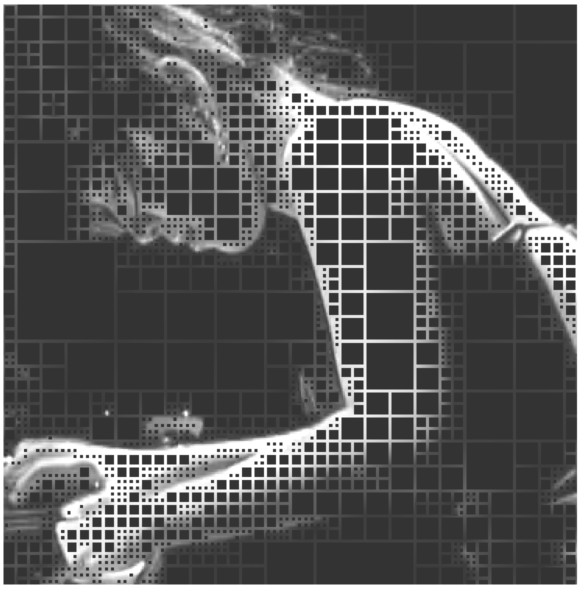

Figure 4: An example of block structure of the source image.

One of variants of approximation of a fragment of image by a surface of a polynomial form and a criterion that the given surface approximates the function of luminosity well are described below.

Suppose that on the image four points are settled in tops of a square of $K \times K$ size.

Accordingly the coordinates of angular points have the following form:

$$p_1 = (i_1, j_1) = (i, j),$$
$$p_2 = (i_2, j_2) = (i, j+K),$$
$$p_3 = (i_3, j_3) = (i+K, j),$$
$$p_4 = (i_4, j_4) = (i+K, j+K).$$

Our further purpose is to clarify a possibility of enough qualitative approximating the function $S(i, j)$ by a polynomial of a form $f(x) = a + bx + cy + dxy$.

Denote $e_k = S(p_k) = S(i_k, j_k), (1 \le k \le 4)$, i.e. e_k are equal to values of the function S in angular points of the given square.

Further we suppose that the considered square is transferred into an origin of coordinates, i.e. $i = j = 0$. Let us require that in angular points values of a desired polynomial have coincided with values of the function of luminosity.

From here the system of four equations is appearing:

$$e_1 = a + b \cdot 0 + c \cdot 0 + d \cdot 0 \cdot 0,$$
$$e_2 = a + b \cdot K + c \cdot 0 + d \cdot K \cdot 0,$$
$$e_3 = a + b \cdot 0 + c \cdot K + d \cdot 0 \cdot K,$$
$$e_4 = a + b \cdot K + c \cdot K + d \cdot K \cdot K.$$

Its solution has the form:

$$a = e_1,$$
$$b = (e_2 - e_1)/K,$$
$$c = (e_3 - e_1)/K,$$
$$d = (e_4 - e_3 - e_2 + e_1)/K^2.$$

Now suppose that two thresholds $\delta_1 < \delta_2$ are given. For a component Y it is possible to put $\delta_1 = 5, \delta_2 = 10$. For components U, V the given thresholds can be much more, for example, $\delta_1 = 20, \delta_2 = 30$.

If for any i, j inside the square an inequality $|S(i, j) - f(i, j)| \le \delta_1$ is fulfilled then we suppose that the polynomial f approximates S well enough.

If inside the square there exist i, j such that $|S(i, j) - f(i, j)| > \delta_2$ then we suppose that there is no any approximation.

Now suppose that inside the square always an inequality $|S(i, j) - f(i, j)| \le \delta_2$ is fulfilled, but there exist such points i, j for which $|S(i, j) - f(i, j)| > \delta_1$. Denote by q the number of such “bad” points.

Also it is possible to set some threshold $0 < \lambda < 1$, and suppose that an approximation is good, if $q \le \lambda \cdot K^2$. The sense of the given inequality is that the number of “bad” points should be not too major. In this case as earlier we will suppose that approximation is good.

In addition it is possible to apply the following metric criterion. Namely, a distance between any two “bad” points should be more as some beforehand prescribed values. It means that we disregard a small number of “bad” points dispelled on a square. We do not want to consider a situation when they somewhere group together. Otherwise we suppose that there is no any approximation. Usually metric criterion is omitting. Note when there exists a good approximation we can store only 4 numbers a, b, c, d or e_1, e_2, e_3, e_4 respectively.

5. CONCLUSION

For all videocodecs, there will be considered various questions connected with an elaborating of parallel algorithms, taking into account their possible hardware realization. In particular it is supposed to use a memory of some special form with a parallel access to information. Note that this memory was already successfully applied in practice.

6. REFERENCES

[1] Westwater R., Furht Borko, Furht Borivoje. Real-Time Video Compression: Techniques and Algorithms // The Kluwer International Series in Engineering and Computer Science, Vol. 376, 1996 г., 176 p.

[2] Ivanov M. A. On application of a scaling algorithm to a problem of preliminary anti-aliasing in a coding of videosequences // Fifth International Conference of Memories of Academician A.P. Ershov, “Perspectives of Informatics Systems”, Novosibirsk 2003, 2p.

About the author(s)

Feodor Murzin is a doctor at Institute of Informatics Systems SB RAS, scientific secretary of the Institute and the head of Complex Systems Modeling Group. His contact email is murzin@iis.nsk.su.

Tagir Waleev is a Ph.D. student at Institute of Informatics Systems, Russian Academy of Sciences, Siberian Branch, Complex Systems Modeling Group. His contact email is lany@ngs.ru.