Spatiotemporal Video Representation and Compression[1]

S. Kuleshov, A. Zaytseva, and A. Aksenov

St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences
14th line, 39 Saint-Petersburg 199178 Russia
e-mail: kuleshov@iias.spb.su

Abstract—In this paper is presented a novel approach for video representation and compression based on spectral compression of spatiotemporal cube. This solution is inspired by the spatiotemporal representation of video sequences.

Keywords: video, data compression, videocube, codec.

DOI: 10.1134/S1054661813010082

INTRODUCTION

Compression of multimedia data is a field within computer vision that is still in progress. It deals with the problem of data representation which gives the highest level of compression at entropy encoding. The range of applications of video compression includes storing multimedia data on media, digital TV, cellular TV.

Method of compression, giving good compression level at the conservation of subjective quality of expressing is still actual.

In this paper is offered a possible solution for this problem.

VIDEOCUBE

In compression and processing the video data is usually represented as sequences of separate frames, which are encoded almost independently. There is also the essentially different way of data representation—the discrete space. The idea of transition from traditional representation to representation in discrete space can be seen in Fig. 1.

This 3D video representation allows to save local particularities both within one frame, and between frames, so it becomes possible the compression video using redundancy within one frame and between frames.

One of the papers that first mentioned the VideoCube and the "motion as orientation" effect of the paths formed in the t direction, was done [1]. It is oriented to visual perception, and proposes to detect motion models based on energy and impulse response filters.

In the corresponding horizontal plane in the VideoCube one can easily identify (nearly straight) line patterns (see Fig. 2). These lines are related to the relative apparent displacement of the objects due to motion of one.

Presence of these curves also provides an opportunity of compression of the data on a time line.

The name VideoCube we will give to a 3D discrete space, which elements are pixels of consequent of frames from source video data.

As element of the VideoCube we shall name an element of 3d representation of the points of video in size $N \times N \times N$ above which a series of operations on transformation is made. Each $N \times N \times N$ block is input, makes its way through each processing step, and yields output in compressed form into the data stream.

PROCESSING STEPS FOR STVC CODING

Figures 3 and 4 show the key processing steps which are the central part of the Spatiotemporal Video Compression (STVC) modes of operation.

RGB to YUV Transforming

At this stage the image is translated from RGB color space with the components responsible for red, green and blue parts of the point color into YCrCb color space (YUV version) where Y is brightness and Cr, Cb components mean chromatic red and chromatic blue.

Because the human eye is less sensitive to color, than to brightness, there is an opportunity to compress data for Cr and Cb component with the big losses and, accordingly, the greater degree of compression. Similar transformation already for a long time is used in TV where on the signals responsible for color, narrower band is allocated.

[1] The article is published in the original.

Received March 03, 2011

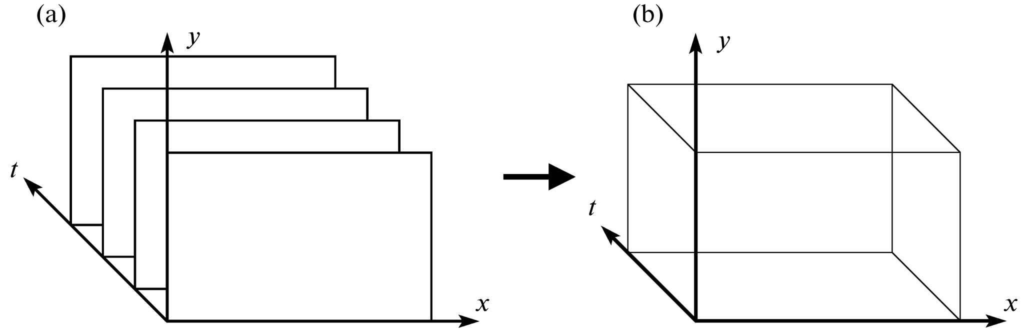

Fig. 1. Sequence of frames when represented as sequence (a) and spatiotemporal space (b).

The following equation is mathematical definitions of transform from color space RGB in color space YCrCb:

$$\begin{vmatrix} Y \\ Cr \\ Cb \end{vmatrix} = \begin{vmatrix} 0.299 & 0.587 & 0.114 \\ 0.500 & -0.418 & -0.081 \\ -0.168 & -0.331 & 0.500 \end{vmatrix} * \begin{vmatrix} R \\ G \\ B \end{vmatrix} + \begin{vmatrix} 0 \\ 128 \\ 128 \end{vmatrix}$$

Inverse transform from color space YCrCb to color space RGB is:

$$\begin{vmatrix} R \\ G \\ B \end{vmatrix} = \begin{vmatrix} 1 & 0 & 1.402 \\ 1 & -0.344 & -0.714 \\ 1 & 1.772 & 0 \end{vmatrix} * \left(\begin{vmatrix} Y \\ Cb \\ Cr \end{vmatrix} - \begin{vmatrix} 0 \\ 128 \\ 128 \end{vmatrix} \right).$$

Sampling

At a stage of sampling (see Fig. 5) is made splitting of the initial discrete parallelepiped into elements of a VideoCube—cubes $N \times N \times N$ where N is usually equal 8 (empirical value). 3 working matrixes of 8 bits of each element are formed separately for every component for the further transformation.

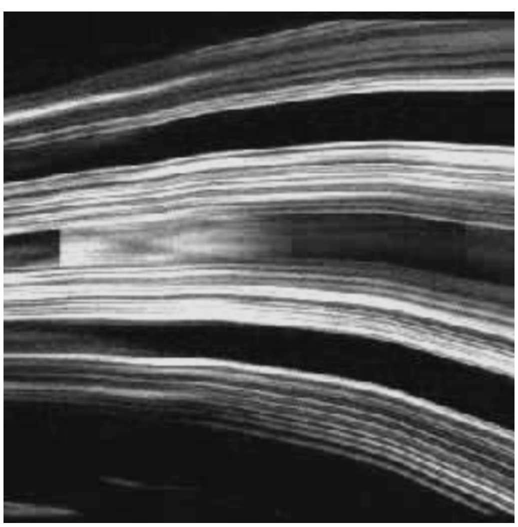

Fig. 2. Example of section of a VideoCube plane XOT (horizontal plane). Spatiotemporal curves.

On this step the adaptive choice of the working element size also can be used about it is in more detail told below.

For increasing the compression degree can be used only the part of the data about a color a component of the image. In this case the data of component Y is used completely, and for Cr and Cb components matrixes are get through a line and through a column. So from an initial cube of size 16 × 16 × 16 only one working matrix turns out.

3D DCT and IDCT

With the purpose of maintenance of a high degree of a video compression it is required, by means of certain homomorphous transformations, to form a structure of data, which will effectively be entropy compression.

One of such transformations is translation of the data in spectral area, for example a discrete cosine transformation (DCT) on 3 dimension. With this transformation lowfrequency and highfrequency coefficients of the video data are located in opposite corners of VideoCube element. Whereas, majority of real graphic images consists of lowfrequency information, possible reduce accuracy of presentation highfrequency coefficients in step of quantization without essential visual worsening a quality.

The following equations are mathematical definitions of the 8 × 8 × 8 DCT:

$$D_{imp} = C_m C_n C_p \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sum^{P-1} S_{kji} \cos\left(\frac{\pi(2k+1)m}{2M}\right) \times \cos\left(\frac{\pi(2j+1)n}{2N}\right) \cos\left(\frac{\pi(2i+1)p}{2P}\right).$$

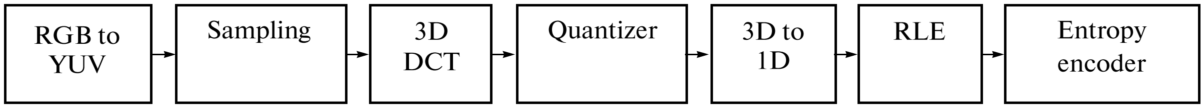

Fig. 3. STVC encoder processing steps.

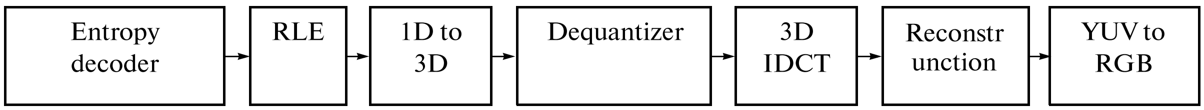

Fig. 4. STVC decoder processing steps.

Matrixes sizes will be assigned by values M, N, P. When sampling of VideCube on elements $8 \times 8 \times 8$ values $M = N = P$ are equals to 8.

Quantization

Quantization is processed like quantization in JPEG [2]. After output from the DCT, each of the 512 DCT coefficients is uniformly quantized in conjunction with a 512 element Quantization Table, which must be specified by the application (or user) as an input to the encoder. Each element can be any integer value from 1 to 255, which specifies the step size of the quantizer for its corresponding DCT coefficient. The purpose of quantization is to achieve further compression by representing DCT coefficients with no greater precision than is necessary to achieve the desired image quality. Stated another way, the goal of this processing step is to discard information which is not visually significant.

Quantization is defined as division of each DCT coefficient by its corresponding quantizer coefficient, followed by rounding to the nearest integer:

$$D(x, y, t) = \left\lfloor \frac{S(x, y, t)}{q} \right\rfloor,$$

where S—source matrix (before quantization), D—destination matrix (after quantization), q—value of quantizer coefficient.

Dequantization is the inverse function, which in this case means simply that the normalization is removed by multiplying by the step size, which returns the result to a representation appropriate for input to the IDCT:

$$D(x, y, t) = qS(x, y, t).$$

Transformation of Elements of Space to Sequence (ZigZag mapping)

Finally, all of the quantized coefficients are ordered into the "zigzag" sequence, also shown in Fig. 7.

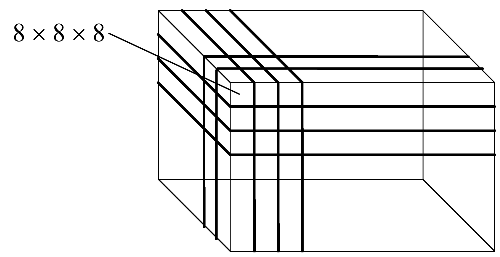

Fig. 5. Sampling stage.

This ordering helps to facilitate entropy coding by placing lowfrequency coefficients (which are more likely to be nonzero) before highfrequency coefficients. In the Figure 6 is shown data ordering in cube $8 \times 8 \times 8$ (from 1 to 512) for "zigzag" sequence.

LZSE Coding

Long zero sequences encoding (LZSE) is a kind of RLE compression (Run-Length Encoding). Each sequence of certain length, which ends by zeroes, is encoded to the following structure: ⟨nonzero_counter, data_without_zeroes⟩ So, the sequence 42 3 0 –2 0 0 0 0 0 0 will be transformed to the sequence 4 42 3 0 –2. Such modification of Run-Length Encoding algorithm gives high compression degree for data gotten after ZigZag mapping.

This compression algorithm is more powerful with separated storage of high and low parts of values like in following sequence ⟨hi_val1, hi_val_2, ..., hi_valN⟩ and ⟨lo_val1, lo_val_2, ..., lo_valN⟩. In this case

Comparison of codecs parameters

Method of compression	Compression object	Size of data, byte
STVC	512 frames of video sequence, size 256 × 256 pixels	1192512
DivX		1118208

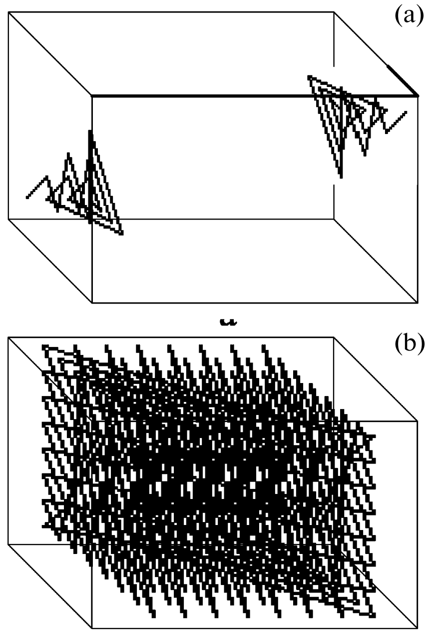

Fig. 6. "Zigzag" sequence curve (a—starting and ending fragments of curves, b—full "zigzag" curve).

sequence of high parts of values will be compressed with the highest degree.

Entropy Coding

The final STVC encoder processing step is entropy coding. This step achieves additional compression losslessly by encoding the quantized DCT coefficients more compactly based on their statistical characteristics. The most effective are two entropy coding methods—Huffman coding [3] and arithmetic coding [4]. These methods reduce an average length of code word for preprocessed video data. Arithmetic coding has produced better compression than Huffman's method for many of the video sequences.

High efficiency of compression, which gives this algorithm, based on that fact that in the matrix of frequency coefficients, forming from source matrixes after the DCT, lowfrequency components located portably in the corner parallelepiped.

OTHER ASPECTS OF THE STVC

The most advanced stage of a compression of an element of a VideoCube takes place in case of the static image or smooth movement, and the worst—at change of a plot. This feature allows to use dynamic change of the size of an elementary VideoCube on an axis of time for improvement of parameters of the codec.

CONCLUSION

Advantages of described method of compression are:

—simplicity of software and hardware realization;

—absence of "key frames" (i.e. possibility of free positioning);

—decoding a series of frames for one iteration of algorithm;

—possibility of stream transmitting and bitrate control;

—symmetry of codec (equal time of coding/decoding);

—absence of visual defects in the manner of "artifacts";

—homogeneity of flow (low dependence of peak load on the processor from the contents of expressing).

Disadvantages of described method of compression are:

—required performance of CPU higher, than MPEG4. Comparison of codecs parameters, under the alike subjective quality evaluation of expressing by the expert, are provided in the table.

REFERENCES

1. E. H. Adelson and J. R. Bergen, "Spatiotemporal Energy Models for the Perception of Motion," J. Opt. Soc. Am. A **2** (2), 284–299 (1985).
2. G. K. Wallace, "The JPEG Algorithm for Image Compression Standard," Commun. ACM **34** (4), 30–44 (1991).
3. D. A. Huffman, "A Method for the Construction of Minimum Redundancy Codes," Proc. IRE **40**, 1098–1101 (1962).
4. I. Witten, R. M. Neal, and G. Cleary, "Arithmetic Coding for Data Compression," Comm. ACM **30**, No. 6, 520–540 (1987).

Sergey V. Kuleshov. Graduated from the Kurgan State University in 2003, subject area: Software Engineering. He received his PhD degree in 2006, subject: Automated system of the semantic analysis and construction of visual dynamic glossaries development. Currently works as senior researcher in the Laboratory of Research Automation of St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences. Area of scientific interests: Infology information systems, analytical Internet monitoring, video data streams processing, applications of software defined radio (SDR). Author of 14 scientific publications, including 2 monograph, in the area of pattern recognition and image analysis.

Alexandra A. Zaytseva. Graduated from the St. Petersburg State Polytechnical University in 2001. She received her PhD degree in 2009, subject: Adaptive dynamic segmentation and semantic selection of videodata. Currently works as researcher in the Laboratory of Research Automation of St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences. Area of scientific interests: video data streams and image processing, development and comparing of semantic criteria for image quality. Author of 11 scientific publications in the area of pattern recognition and image analysis.

Alexey J. Aksenov. Has graduated from the Saint Petersburg Electrotechnical University "LETI" in 2001, was educated in SPIIRAS Post-graduate school. Currently works as junior researcher in the Laboratory of Research Automation of St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences. Area of scientific interests: signal processing, application of software defined radio (SDR). Author of 5 scientific publications in the area of pattern recognition and image analysis.