COMPRESSION METHODS USING 3D DCT

Jiří Hermany
Doctoral Degree Programme (1), FEEC BUT
E-mail: xherma09@stud.feec.vutbr.cz

Supervised by: Stanislav Hanus
E-mail: hanus@feec.vutbr.cz

ABSTRACT

This paper describes the main process of a video sequence compression method using the 3D DCT and its main difference in comparison with 2D DCT. The article describes a video cube creation from the video sequence and the principle of the separately proceeded 3C DCT computation. Also the quantization and entropy coding processes are presented. The main problems: a cutting recognition and treatment, a video cube "zig-zag" reordering and a 3D DCT algorithm simplifying, arising from used method are outlined.

1. INTRODUCTION

JPEG and MPEG are recently the most used image and video compression methods. These methods are not lossless. With growing compression ratio (CR) the quality of the output image or video sequence decreases as a result of removing redundant data which is allowed by using the human eye imperfection.

Both methods use Two-Dimensional Discrete Cosine Transform (2D DCT) to produce a kind of spatial frequency spectrum. The magnitudes of low frequency components can be stored with lower accuracy accordingly to different sensitivity of human vision to colour or brightness changes in large areas than to the high frequency brightness variations.

Considering that MPEG is based on compression of the sequence of single images, there could be a possibility to reach good compression ratio using the Three-Dimensional Discrete Cosine Transform (3D DCT). This idea differs by using "video cube" which is a cube of N × N × N video elements.

The 3D DCT can make the account of the neighbouring pictures correlation in the video cube the same as the 2D DCT uses the correlation of the neighbouring pixels in 2D matrix.

The aim of my dissertation thesis is to verify the possibility of implementing the 3D DCT compression algorithm into Digital Signal Processor (DSP). This article describes some necessary basics for introducing into this field.

2. COMPRESSION PROCESS

The compression process can be divided into three main parts (Figure 1). The video sequence is on the input, compressed video sequence on the output.

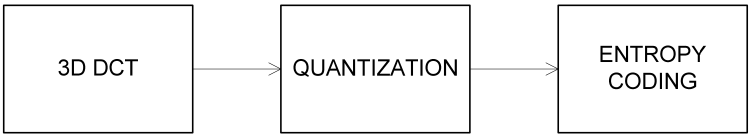

Figure 1: Compression process

The input video sequence can be divided into so-called video cubes. The principle of a video cube composition from video sequence frames is shown in the Figure 2, each video cube contains 512 video elements.

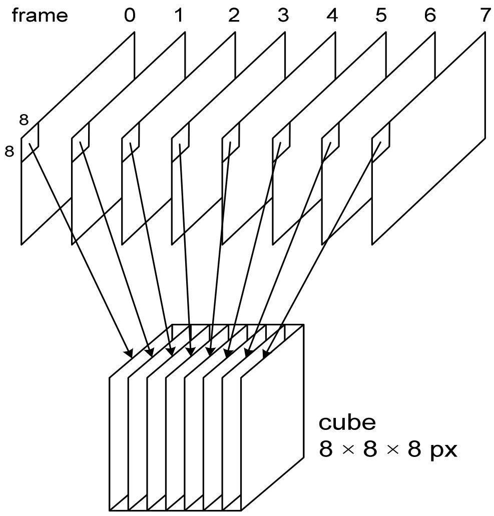

Figure 2: Video cube of 8 × 8 × 8 pixels

If the monochromatic video sequence is on the input, the first part of the compression process is the 3D DCT block which is the most time consuming.

2.1. Discrete Cosine Transform

The three-dimensional variant of the DCT is a composition of three 1D DCT along each dimension. The formal definition is

$$X(l,m,n) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\sum_{k=0}^{N-1} x(i,j,k)\cdot C_{li}\cdot C_{mj}\cdot C_{nk}\ , \qquad (1)$$

where $x(i,j,k)$ is a value of the video cube element which is positioned at the coordinates i,j,k, $X(l,m,n)$ is a 3D DCT coefficient at the position l,m,n and indexes take values $l,m,n = 0,1,\ldots,N-1$.

The $C_{li}\cdot C_{mj}\cdot C_{nk}$ multiplication is the 3D DCT base function which can be defined as

$$C_{li}\cdot C_{mj}\cdot C_{nk} = \cos\left[\frac{\pi}{N}\left(i+\frac{1}{2}\right)l\right]\cdot\cos\left[\frac{\pi}{N}\left(j+\frac{1}{2}\right)m\right]\cdot\cos\left[\frac{\pi}{N}\left(k+\frac{1}{2}\right)n\right]. \qquad (2)$$

A cube of frequency components is the product of the 3D DCT. It contains one DC coefficient $X(0,0,0)$ at zero coordinates, the remaining 511 coefficients are called AC coefficients. The most important elements of the signal are concerned near this DC coefficient.

2.2. QUANTIZATION

The human eye characteristics allow removing a lot of redundant information in higher frequency coefficients. It can be done by dividing each frequency component by a suitable constant and by consecutive rounding to the nearest integer. As a result, many of higher frequency coefficients are rounded to zero. The quantized 3D DCT coefficients can be computed with

$$X_q(l,m,n) = \frac{X(l,m,n)}{Q(l,m,n)} \tag{3}$$

where $X(l,m,n)$ are the frequency coefficients before quantizing, $X_q(l,m,n)$ are the frequency coefficients after quantizing and $Q(l,m,n)$ are the quantizing coefficients (as in [2]).

This operation is lossy as a result of this division therefore these components can not be restored in the decompression process. However it causes decreasing amount of data to store. It is necessary to decide which constant will be used for quantizing each frequency component therefore the quantization cube must be defined. Its segments determine the compression ratio and the quality of output video sequence.

2.3. ENTROPY CODING

Entropy coding is a lossless data compression. One of the most common is the Huffman coding which is also used in JPEG and MPEG.

Data in the quantized cube must be rearranged into a “zig-zag” order. The more zeros will be in the straight-line the less data will be necessary to store. Consequently it also influences the final compression ratio.

3. MAIN PROBLEMS

Using the 3D DCT in the video compression has some advantages for video processing. However, some problems must be necessarily solved:

- cutting recognition and treatment,
- video cube “zig-zag” reordering,
- 3D DCT algorithm simplifying.

3.1. CUTTING TREATMENT

The 3D DCT is deriving benefit from the correlation of the neighbouring frames. In spite of that, if the cutting (a scene change) will be in the video cube (Figure 3), it would cause a so-called cross-fade effect. After decompression, this will display fragments of both scenes included in the video cube.

One of the proper solutions may be to ensemble the video cube with pictures from the first scene if the scene change is present in the video cube. The second could be to darken nearest pictures of the ending scene and also to raise the nearest pictures of the beginning scene as in [2]. Another way how to remove the cross-fade effect is the coder modification. It is based on separation of different scenes in video cube and their individual transformation.

First two 1D DCT transforms are used as usual and the third (along temporal axis) is modified [1].

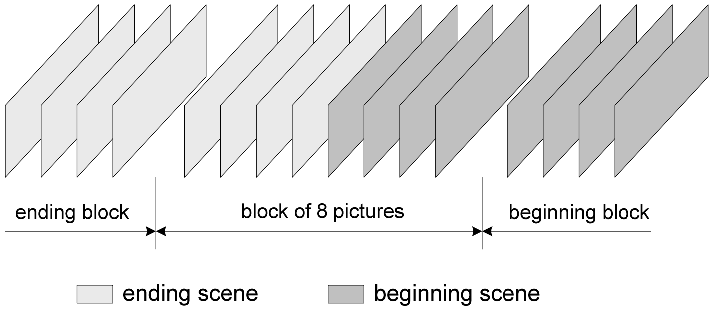

Figure 3: Cutting (scene change) in video cube

3.2. "ZIG-ZAG" REORDERING

This procedure is important for maintenance of the resulting video sequence compression ratio. The more zero coefficients will be in the reordered sequence the higher the compression ratio would be. The "zig-zag" reordering method used in JPEG is known [1], yet the method for the video cube is necessary to model and verify.

3.3. 3D DCT SIMPLIFYING

Because the 3D DCT is the most time demanding part of the whole compression process, it should be simplified as possible. The 3D DCT can be decomposed into the three 1D DCTs as shown in Figure 4. One way to simplifying the 3D DCT computation process could be using the similarity of the DCT with Discrete Fourier Transform (DFT) as described in [1].

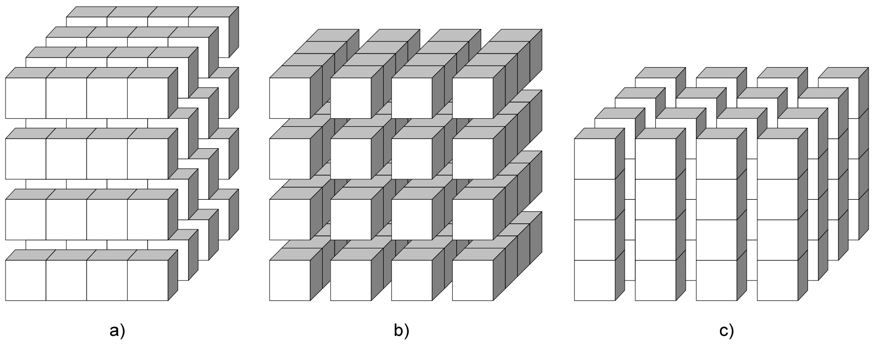

Figure 4: 3D DCT decomposition into three 1D DCTs along each dimension

4. COMPUTATION DEMANDS

Finding out the amount of processed data is the first step to solving this area. PAL (Phase Alternating Line) is a colour model commonly used in the broadcast television systems. Its digital version has a resolution of 720×576 pixels. We will use this standard as an initial for the following calculations. The monochromatic sequence frame of 720×576 pixels contains 6480 (90×72) 2D matrixes with 8×8 pixels each. Using 8 frames and considering 24 frames per second mean the number of 19,440 (6,480×3) video cubes in the sequence per second.

The 3D DCT computation for each video cube is necessary. Because of its decomposition into three 1D DCTs, the number of cubes must by multiplied by 64 columns in 3 dimensions. It gives 3,732,480 1D DCT computations per second. According to [1], 29 sum and 5 product operations are necessary for one 1D DCT computation. That means 108,241,920 sum and 18,662,400 product operations per second only for 3D DCT compression without processing quantization or entropy coding.

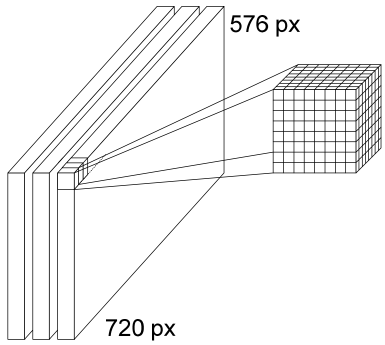

Figure 5: Three sets of 8 frames – 24 frames of video sequence

This is very high number for signal processing. For colour sequences, more (about three times) operations are needed because the colour sequence (or picture) is usually composed from three different components. Is could be red, green and blue frames for the RGB colour model or luminance and two chrominance frames for the YUV colour model.

These results will have some consequences. Firstly, very fast DSP must be chosen because of the great number of compression process operations. Next consequence could be the input signal under-sampling if there won't be as much efficient DSP as we would need.

5. CONCLUSION

The compression method using the 3D DCT was examined and its progress was described. Main tasks which will be necessary to work out and their possible solutions were discussed. The future steps will be deeper research of outlined problems. Because the DSP implementation should be the main task of further work, the investigation will aim this way.

ACKNOWLEDGEMENT

This work has been supported by the research project of the GA CR (Czech Science Foundation) No. 102/07/1295 "Models of Mobile Networks and their Parts" and by the research program No. MSM 0021630513 "Advanced Electronic Communication Systems and Technologies" (ELCOM).

REFERENCES

[1] FRÝZA, T. *Komprimace obrazových signálů pomocí transformace 3D DCT*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2005. 178 s. Vedoucí disertační práce Doc. Ing. Stanislav Hanus, CSc.

[2] CHROMÝ, I. *Komprese digitálních obrazových signálů*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2001. 81 s. Vedoucí disertační práce Doc. Ing. Stanislav Hanus, CSc.