MODELING OF REAL TIME VIDEO COMPRESSION SYSTEM
Three-dimensional Discrete Cosine Transform

Tomas Fryza
Department of Radio Electronics, Brno University of Technology, Purkynova 118, Brno, Czech Republic

Keywords: Video signal compression, 3-D DCT, complexity, implementation, DSP TMS320C6711, C language, linear assembler.

Abstract: One of the methods used for the video signals' compression is the Three Dimensional Discrete Cosine Transform. The aim of this block-based method is to combine intraframe and interframe coding into a single transform coding, therefore no motion compensation and motion prediction have to be implemented. The paper deals with the practical ways of the 3-D DCT computing. It will be proof, the transform coding could be used for encoding of video sequences in real time domain.

1 INTRODUCTION

A video compression method has two purposes: a) reduce spatial redundancy between adjacent picture elements by intraframe coding and b) reduce temporal redundancy between adjacent frames with help of interframe coding. Individual compression algorithms use different mechanisms to achieve these principles. The 3-D DCT (Three Dimensional Discrete Cosine Transform) is based on a consolidation of both coding into a single transform coding where several frames are being encoded simultaneously. The substance of 3-D DCT is similar to the JPEG standard (Wallace, 1992). Each group of N frames is divided into small segments (so-called video cubes). From these video cubes, the frequency coefficients and output binary stream are being formed.

The contribution presents the fundamental properties of the 3-D DCT and mainly the conditions for utilization such a method in real time processing domain. The paper is divided into three parts. In Section 2 the mathematical background of different 3-D DCT variations is presented. The Section 3 discuss the minimal hardware demands for real time operation and final results are given in Section 4.

2 TRANSFORM CODING

As mentioned above in 3-D DCT compression scheme the succession of input frames is divided into groups of N frames. The value of N controls not only the amount of allocated memory space for captured samples or the number of repetition per second but also the mathematical complexity of the encoder itself.

Entire encoding scheme could be split into a preprocessing phase where the input samples could be resampled or represented into different color space. Next part of the encoder is the transform coding itself, and finally a postprocessing phase with a quantizer and threshold processes. Last part compounds of the entropy encoding and forming of the output bit stream.

The analysis of the complexity proportion of individual encoder parts proves the 3-D DCT transform coding is the most consuming process of entire encoding stage. Thus, in the next text only the transform coding is analyzed by reason of the estimation of real time processing possibilities.

For video cube dimensions of N, the forward 3D DCT for a grey-scale video sequence is defined in the following way (Rao and Yip, 1990)

$$D_{u,v,w} = \gamma_u\gamma_v\gamma_w \cdot \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\sum_{z=0}^{N-1} f_{x,y,z} \cdot \cos\frac{\pi u(2x+1)}{2N} \cdot \cos\frac{\pi v(2y+1)}{2N} \cdot \cos\frac{\pi w(2z+1)}{2N} \quad (1)$$

where $D_{u,v,w}$ represents DCT (frequency) coefficient of a picture element intensity $f_{x,y,z}$ while $u,v,w = 0,1,\ldots,N-1$. The constants γ could be expressed as follows

$$\gamma_{u,v,w} = \begin{cases} 1/\sqrt{2} & : \quad u,v,w = 0 \\ 1 & : \quad u,v,w \neq 0. \end{cases}$$

Fryza T. (2008).
MODELING OF REAL TIME VIDEO COMPRESSION SYSTEM - Three-dimensional Discrete Cosine Transform.
In *Proceedings of the International Conference on Signal Processing and Multimedia Applications*, pages 208-211
DOI: 10.5220/0001938102080211

The 3-D DCT definition from (1) implies high number of arithmetical operations. Therefore, the 3-D transform could be re-write in a form of successive three 1-D transforms with obvious impact in reducting of mathematical operations. In the next text two fast algorithms for evaluating of 1-D transform with different value of N are outlined.

2.1 8-point Discrete Cosine Transform

Let $N = 8$, then one dimensional discrete cosine transform for a single color palette is defined by (2), while $u = 0, 1, \ldots, 7$.

$$D_u = \gamma_u \cdot \sum_{x=0}^{7} f_x \cdot \cos \frac{\pi u(2x+1)}{16} \qquad (2)$$

In purpose of reducing the mathematical operations, the multiplying by constant γ_u could be compound into a quantizer block, i.e. into the last part of video encoder (post processing part). In practice, the evaluation of 1-D DCT is obtain with help of Discrete Fourier Transform and through the set of modified equations (Gonzalez and Wintz, 1987):

$$\begin{aligned}
\Re\{F_0\} &= 2 \cdot (f_0 + f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7) \\
\Re\{F_1\} &= f_0 - f_7 + C_0 \cdot I_0 + C_1 \cdot I_1 - C_2 \cdot (I_1 - I_2) \\
\Re\{F_2\} &= f_0 - f_3 - f_4 + f_7 + C_0 \cdot I_3 \qquad (3) \\
\Re\{F_3\} &= f_0 - f_7 - C_0 \cdot I_0 - C_3 \cdot I_2 - C_2 \cdot (I_1 - I_2) \\
\Re\{F_4\} &= f_0 - f_1 - f_2 + f_3 + f_4 - f_5 - f_6 + f_7 \\
\Re\{F_5\} &= f_0 - f_7 - C_0 \cdot I_0 + C_3 \cdot I_2 + C_2 \cdot (I_1 - I_2) \\
\Re\{F_6\} &= f_0 - f_3 - f_4 + f_7 - C_0 \cdot I_3 \\
\Re\{F_7\} &= f_0 - f_7 + C_0 \cdot I_0 - C_1 \cdot I_1 + C_2 \cdot (I_1 - I_2)
\end{aligned}$$

where $\Re\{F_u\}$ represents real part of Fourier coefficients and constants C_i are defined as follows

$$\begin{aligned}
C_0 &= \cos(4\pi/16) \qquad (4) \\
C_1 &= \cos(2\pi/16) + \cos(6\pi/16) \\
C_2 &= \cos(2\pi/16) \\
C_3 &= \cos(2\pi/16) - \cos(6\pi/16)
\end{aligned}$$

and combinations of input samples I_i are given by

$$\begin{aligned}
I_0 &= f_1 + f_2 - f_5 - f_6 \qquad (5) \\
I_1 &= f_2 + f_3 - f_4 - f_5 \\
I_2 &= f_0 + f_1 - f_6 - f_7 \\
I_3 &= f_0 + f_1 - f_2 - f_3 - f_4 - f_5 + f_6 - f_7.
\end{aligned}$$

According to the (3)-(5), 5 products operations and 29 sums operations have to be performed in order to evaluate eight 1-D coefficients. In every video cube, this 1-D transform have to be repeated 192 times to obtain 512 frequency coefficients. Imaging a test grayscale video sequence with dimensions of 720×576 picture elements and length of 24 frames. Therefore the minimal number of arithmetical operations for encoding such a sequence is 18,662,400 products and 108,241,920 sums (Fryza and Hanus, 2003).

2.2 4-point Discrete Cosine Transform

According to (1), the one dimensional 4-point forward discrete cosine transform could be express as follows

$$D_u = \sum_{x=0}^{3} f_x \cdot \cos \frac{\pi u(2x+1)}{8} \qquad (6)$$

where $u = 0, 1, \ldots, 3$. Applying the similar procedures mentioned in (Gonzalez and Wintz, 1987), the set of modified equations for fast calculations of 4-point 1-D DCT could be evaluated. Using the extension of 4 input samples in term of $f_x = f_{7-x}$ and using the Discrete Fourier Transform defined by (7), the 1-D DCT could be expressed by (8).

$$F_u = \sum_{x=0}^{7} f_x \cdot \exp(-2\pi j u \cdot x/8) \qquad (7)$$

$$\sum_{x=0}^{3} f_x \cdot \cos \frac{\pi u(2x+1)}{8} = \frac{\Re\{F_u\}}{2 \cdot \cos(\frac{\pi u}{8})} \qquad (8)$$

The left part of equation (8) is equal to the 1-D DCT. Therefore, 1-D transform could be evaluated by real parts of Fourier coefficients divided by the real constant $2 \cdot \cos(\frac{\pi u}{8})$. Likewise the γ values, the constant could be also incorporated in quantizer block of the encoder. Hence, the only task is to enumerate the real parts of F_u. It could be done by the set of equations defined below

$$\begin{aligned}
\Re\{F_0\} &= 2 \cdot (f_0 + f_1 + f_2 + f_3) \qquad (9) \\
\Re\{F_1\} &= f_0 - f_3 + \cos \frac{2\pi}{8} \cdot (f_1 - f_3 - f_2 + f_0) \\
\Re\{F_2\} &= f_0 - f_2 + f_3 - f_1) \\
\Re\{F_3\} &= f_0 - f_3 - \cos \frac{2\pi}{8} \cdot (f_1 - f_3 - f_2 + f_0)
\end{aligned}$$

It can be seen the number of necessary arithmetical operation for 4-point 1-D DCT is 1 product and 9 sums. Nevertheless, according to the smaller blocks of input samples encoded in one moment, the total number of operations for transforming the tested video sequence from Subsection 2.1 ($720 \times 576 \times 24$) is 7,464,960 products and 67,184,640 sums, i.e. lower number.

3 COMPLEXITY VERIFICATION

In this section, the models of evaluated transform coding are being verified in order to verify the real time processing possibilities. According to the simplification applied in Section 2, the criterion for examination of real time processing is only the calculation of 3-D DCT.

The algorithms mentioned in Subsections 2.1 and 2.2 were programed for digital signal processor TMS320C6711 from Texas Instruments in C language and in so-called linear assembler. The linear assembler is an interstage between high level C language and low level assember code. The floating-point processor TMS320C6711 contains eight functional units such as hardware multiplier unit or unit for memory accessing, 32 32-bit registers and it is controlled by the clock signal with a relatively low frequency of 150MHz. The basic tool for evaluating of the algorithm velocity is a total number of CPU cycles. Every instruction of DSP has a specific number of needed cycles and the number depends on the type of instruction. In general the most consuming instructions are the accessing the memory and multiplication in double or even in extended floating-point precisions.

The total number of needed CPU cycles for 8-point and 4-point 1-D DCT is shown in Table 1 and in Table 2 respectively. The estimated time for encoding the grey scaled video sequence with dimensions of $720 \times 576 \times 24$ are shown as well. The parameters -o0, -o1, -o2 and -o3 correspond to the level of source code optimizing. Parameter -o0 enables the register level optimizing, -o1 and -o2 starts function level optimizing and parameter -o3 corresponds to the optimizing on file level. The optimizing could be done by the Code Composer Studio development software from Texas Instruments.

An example of using the 4-point version of DCT encoder for real video sequence processing is shown in Fig. 1, where one frame from original sequence and three details with different quality levels are presented.

Table 1: CPU cycles for 8-point 1-D DCT and duration of transforming a grayscale video sequence ($720 \times 576 \times 24$, $f_{CPU} = 150MHz$).

	C code		ASM code	
Param.	Cycles	Time [s]	Cycles	Time [s]
no opt.	57,655	22.42	10,284	4.00
-o0	52,759	20.51	10,284	4.00
-o1	26,226	10.20	5,206	2.02
-o2	15,527	6.04	2,144	0.83
-o3	15,527	6.04	2,144	0.83

Table 2: CPU cycles for 4-point 1-D DCT and duration of transforming a grayscale video sequence ($720 \times 576 \times 24$, $f_{CPU} = 150MHz$).

	C code		ASM code	
Param.	Cycles	Time [s]	Cycles	Time [s]
no opt.	5,470	17.01	1,054	3.28
-o0	4,615	14.35	1,054	3.28
-o1	2,975	9.25	702	2.18
-o2	1,583	4.92	417	1.30
-o3	1,583	4.92	417	1.30

Figure 1: A frame of tested sequence "high jump" encoded by 4-point encoder version with different quality levels.

It can be seen the combination of lower level programming languages and the optimizing tools are unavoidable to achieved the code effective applications. Also it can be seen the only possibility for encoding a grey scaled sequence with PAL resolutions on DSP TMS320C6711 (controlled by $f_{CPU} = 150MHz$) is using the 8-point fast algorithm version with maximal level of optimizing.

4 CONCLUSIONS

The contribution was focused into the video compression domain, and mainly into the modeling of the real time 3-D DCT encoding system. This 3-D system is used to replace two ways of video compression, i.e. intraframe and interframe coding. In the paper two versions of fast 1-D algorithms were outlined, for 8-point and 4-point DCT system. It was proved the total number of needed mathematical operations and memory demands declines with the selected *N*-point version. The better picture quality of encoded video sequences could be reached by low-point DCT version as well (Fryza and Hanus, 2004). The practical verification of the 3-D DCT calculation on floating-point digital signal processor TMS320C6711 was also described. The criteria was the total number of CPU cycles. Only the high capability of the Texas Instruments optimizing tools cause the 3-D DCT transform based on 8-point 1-D DCT could be usable in real time processing domain.

ACKNOWLEDGEMENTS

The author would like to thank for the financial support of the Czech Ministry of Education, Youth and Sports under grants no. MSM 002 163 0513, 1880250/2008 and of the Czech grant agency of Science Academy grant no. KJB 208 130 704.

REFERENCES

Fryza, T. and Hanus, S. (2003). Algorithms for fast computing of the 3d dct transform. *Radioengineering*, 12(1):23–26.

Fryza, T. and Hanus, S. (2004). Dissimilarity detection in video sequences using variation in time domain. In *Radioelektronika 2004 Conference Proceedings*. FEI STUBA.

Gonzalez, R. and Wintz, P. (1987). *Digital Image Processing*. Addison Wesley Publishing Company, Boston, 2nd edition.

Rao, K. and Yip, P. (1990). *Discrete Cosine Transform. Algorithms, Advantages, Applications.* Academic Press, Inc., San Diego, 1nd edition.

Wallace, G. (1992). The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv.