NEW DCT COMPUTATION ALGORITHM FOR VIDEO QUALITY SCALING

Stephan Mietens (1), Peter H.N. de With (1,2) and Christian Hentschel (3)

Eindhoven University of Technology - EESI(1) / CMG Eindhoven(2), P.O. Box 513, 5600 MB Eindhoven, The Netherlands
Philips Research Labs.(3), Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands

ABSTRACT

The application of video coding systems such as MPEG in portable systems like organizers and mobile phones, can be scaled down to reduced complexity that matches with the desired video quality and/or display. In this paper, a new DCT computation algorithm is presented, based on an analysis for optimizing the amount of computations involved at each computing stage, using existing fast DCT calculation algorithms. The analysis is used to scale down the video quality, thereby lowering the computing power and resource usage. Compared to a diagonally oriented computation of coefficients that matches with the conventional MPEG scanning, a 2-4 SNR dB improvement is obtained when scaling down the video quality to halved computing resources.

1. INTRODUCTION

Currently, video applications based on MPEG video compression are widely used in TV and PC systems and the Digital Versatile Disk (DVD) is becoming a popular home video entertainment application. Until today, research in MPEG-based video compression focused primarily on maximizing the video quality for common applications such as broadcast TV and DVD with a data rate of 3–8 Mbit/s and finding efficient implementations in hardware and software.

However, new applications based on Internet connections like web cameras and video conferencing systems have become available, introducing different quality requirements. These requirements can be exploited for further reduction of the algorithmic complexity for regular applications. The following examples indicate that a certain quality loss is acceptable under circumstances. Firstly, if a part of the available general-purpose computation power of a TV can be saved by using algorithms of lower complexity for video applications, the TV is able to perform other tasks in parallel, which otherwise would not be possible (e.g. multi-window TV, see Fig. 1). A second example is found when using small displays, because the observer cannot perceive fine details, and full processing of the signal would be inefficient and costly.

It is our objective to design a scalable MPEG encoding algorithm that features scalable video quality with respect to the desired application. This scalable algorithm is suited for implementation on a generalized and programmable architecture.

Fig. 1. Multi video application with scalable video quality.

2. SYSTEM REQUIREMENTS

Prior to developing the algorithm, a list of system requirements is briefly discussed and presented below.

- Real-time execution: for example, the calculations have to be finished within a given time.
- Limited computing power: a portable system is limited in speed due to power constraints.
- Video resolution: the size of the display or the processed video frame is variable.
- Bandwidth: the available bandwidth for data transmission is limited.
- Energy consumption: less complex algorithms affect the stand-by time of a device.
- Product cost: a low product cost for mobile equipment is a key factor.

With the objective of a scalable MPEG encoder in mind, new algorithms and/or techniques have to be investigated to simplify MPEG coding, while still preserving the quality as good as possible. Furthermore, the scalable quality should match with the required computational resources (i.e. the applications) and available memory, such that a lower quality leads to less required resources accordingly. This objective has been first explored for computation of the Discrete Cosine Transform (DCT), which is a cornerstone of an MPEG encoder. The new DCT algorithm found in this paper can be used for downscaling computing power and video quality.

3. DISCRETE COSINE TRANSFORMATION

The DCT transforms the luminance values and chrominance values of small square blocks of an image to the transform domain. Afterwards, all coefficients are quantized. The signal concentration into a small amount of coefficients enables that the whole image can be represented with less data than the original.

For a given $N \times N$ image block, represented as a 2-D data matrix $\{X[i,j];\ i,j = 0,1,...,N-1\}$, the 2-D DCT matrix $\{Y[m,n];\ m,n = 0,1,...,N-1\}$ is given by

$$Y = C_N^{\top} * X * C_N, \tag{1}$$

where

$$C_N[p,q] = u(p) * \cos \frac{(2p+1)q * \pi}{2N} \tag{2}$$

and $u(p) = \frac{1}{\sqrt{2}}$ if $p = 0$, otherwise $u(p) = 1$.

State-of-the-art DCT algorithms normally concentrate on optimizing a 1-D DCT ($X * C_N$), which is a part of the complete 2-D DCT. The following algorithm examples are based on mathematical transformations to reduce the complexity of the DCT computation.

- The Lee-Huang recursive algorithm [1] reduces the calculation of the cosine matrix C_N to equivalent subproblems of a lower complexity ($C_N = f(C_{\frac{N}{2}})$).
- The Cho-Lee 2-D Fast algorithm [2] is based on data dependencies between both cosine matrices C_N and $C_N^{\top} = f(C_N)$, where the 2-D transformation is reduced to a 1-D transformation. The selection of the 1-D DCT algorithm is free of choice.
- The Arai-Agui-Nakajima 1-D algorithm [3] is deduced from a Discrete Fourier Transformation (DFT) and the calculation is arranged such that several multiplications (8 of 13) can be integrated in a subsequent quantization step.

The following algorithms presented reduce the computation complexity of the DCT to speed up calculation time, whereby a loss of video quality is accepted.

- The Merhav-Vasudev multiplication-free algorithm [4] transforms the mathematical definition of the DCT into several matrices such that the calculation can be performed using binary shifts instead of multiplications. This results in rounding errors, which are mapped onto a generalized error parameter, which is minimized in a subsequent quantization step.
- The Pao-Sun adaptive DCT modeling [5] is based on a statistical analysis of H.263 encoded video sequences. This analysis is used to decide after the motion compensated prediction, if an image block is processed using a complete or partial DCT.

Although the aforementioned algorithms contain useful starting points for low-cost solutions, the computations do not provide scalability between different applications and the required resources. This is explored in the next section.

4. SORTED DCT-COEFFICIENT COMPUTATION

This section presents a novel technique to find a specific computation order of the DCT coefficients, which depends on the fast DCT algorithm used, in order to design a scalable DCT computation algorithm. The constitution of the DCT flow-graph is important to find the best strategy for scaling the algorithm to lower video quality. The algorithm is modified by eliminating several calculations and thus coefficients. Consequently, the output of the algorithm will have less quality, but the processing effort of the algorithm is reduced and thus it can be calculated faster.

The key issue is to identify the computation steps that can be omitted to maximize the number of coefficients in order to obtain the best possible video quality. This is performed by sorting a list of remaining coefficients prior each computation stage, such that in the next step the coefficient is computed having the lowest computational cost.

Since fast DCT algorithms have different computation characteristics (minimum multiplications or arbitrary block sizes), the algorithm implemented for a certain application should be analyzed closely. Although this paper concentrates on calculating a DCT, the technique can be implemented for an IDCT as well.

DCT algorithm analysis

To identify the required number of operations for specific DCT coefficients, data dependencies between calculation nodes within the algorithm are first explored. Using this information, a database is constructed for every calculation step, when going from the input values to the transform coefficients at the output. The database is used to track all computations involved for a specific coefficient and it also provides calculations still needed for another remaining coefficient. If a computation limit is set, it is preferable to calculate coefficients that share calculation steps. The number of coefficients is then maximized with minimum effort.

The above-given approach (or analysis) gives an algorithm-dependent calculation order, which can be improved a)

	0	1	2	3	4	5	6	7
0	1	33	9	37	3	42	11	34
1	49	17	54	25	59	29	61	21
2	10	38	5	45	15	40	7	47
3	53	31	63	23	51	19	56	27
4	4	43	12	35	2	36	13	44
5	58	28	57	20	52	24	64	32
6	14	48	8	41	16	46	6	39
7	50	22	62	30	60	26	55	18

b)

	0	1	2	3	4	5	6	7
0	1	33	9	41	5	44	14	36
1	17	49	21	57	29	63	31	55
2	10	37	3	42	11	39	7	48
3	25	61	26	53	18	51	24	60
4	6	45	15	34	2	35	16	46
5	28	59	23	52	19	54	27	62
6	12	47	8	40	13	43	4	38
7	20	56	32	64	30	58	22	50

Fig. 2. Optimal calculation order of coefficients in a DCT matrix, using [2] and [3]. Variant (b) was found with an additional priority for low-frequency coefficients. The first and the second half of the coefficients are shaded differently.

if the quantization step after the calculation of the DCT is considered. The quantizer weighting function emphasizes the use of low-frequency coefficients in the upper-left corner of the matrix. Therefore, the coefficients can be combined with a *priority* function to favour those coefficients. Other priority functions can favour coefficients that describe block contents like horizontal or vertical lines. However, this does not foreclose choosing a coefficient having a low priority if its computation can be completed with e.g. a single addition.

Within the MPEG standard, a zigzag scanning is used for coding of the DCT coefficients, to start with the most important low-frequency values that remain normally after quantization. If we adopt this as a calculation order, many time-consuming calculations have to be performed at the start of the complete DCT computation to obtain the first coefficients. The reason is that these values depend on different inputs and no intermediate result can be reused. When in this case the computational resources are reduced, less coefficients would be available afterwards. This illustrates that finding the best computation order is useful.

Application

The approach explained in this section has been used to find an optimal calculation order for a *scalable* 2-D DCT algorithm taken from Cho and Lee [2], when inserting the 1-D DCT algorithm by Arai, Agui and Nakajima [3]. Both algorithms were adopted, because their combination provides a highly efficient computation (104 multiplications and 466 additions). The optimal calculation order found when using our reordering algorithm, is shown in Figure 2(a). The corresponding Figure 2(b) is a modified calculation order, which was found by prioritizing low-frequency coefficients. The different shaded areas mark the first and the second half of coefficients within the given calculation order.

Comparison

The calculation order shown in Figure 2(a) uniformly selects the coefficients from the matrix, which is useful for image blocks missing a clear structure. Vertical block details are clearly favoured by the calculation order shown in Figure 2(b). It can be verified, that the number of coefficients in the top-left corner of the DCT matrix having a

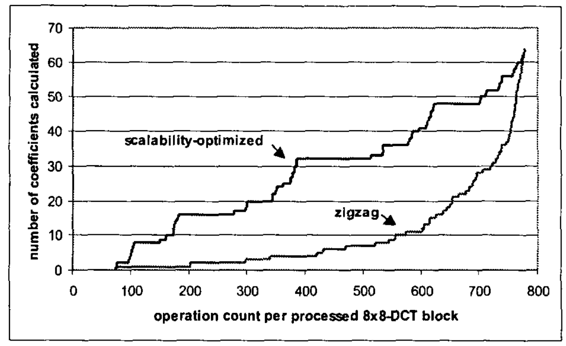

Fig. 3. Comparison of zigzag-ordered computation vs. scalability-optimized order for 8×8-DCT blocks.

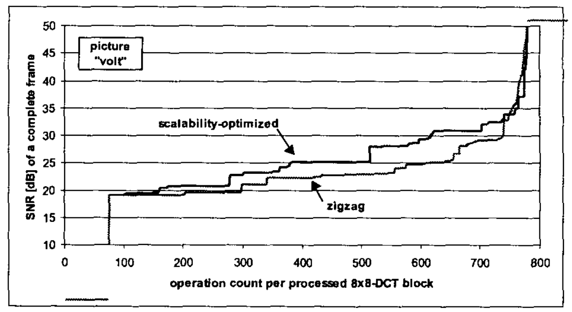

Fig. 4. SNR obtained as a function of the available nr. of operations demonstrated using a frame of sequence "voit".

low ranking number using calculation order (b) is equal or higher than using order (a) without priority function. Note that if a zigzag order would be preferred right from the start, another DCT algorithm could be selected for optimization with this criterion and yield better results (we took low-cost operation as a starting point).

The operations count for a given number of coefficients using the MPEG-zigzag order has been compared with the scalability-optimized order given in Figure 2(b). For comparison, we assume that one addition is one operation and a multiplication is three operations (an alternative cost function can also be choosen). The results are given in Figure 3. It can be noticed that the proposed scalability-optimized order leads to significantly more coefficients calculated, resulting in an improved video quality. The SNR improves between 1-5 dB, depending on the amount of available operations (see Figure 4, and 5, which shows the SNR difference between the usage of both calculation orders). It can be seen that the SNR values increase slowly in the beginning of the computation and augment faster at the end. However, the perceptual quality increases inversely, because at the start,

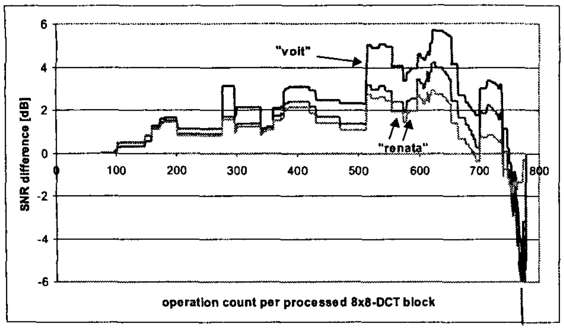

Fig. 5. Improvement expressed in SNR difference (gain) using a scalability-optimized order instead of the zigzag order using a frame of sequence "voit" and sequence "renata". The lowest "renata" curve is caused by including a standard MPEG quantization step.

every additional coefficient contributes much to improve the quality. At the end, the perceived video quality hardly improves, although the SNR grows significantly.

Using a quantizer step between the DCT and IDCT does not change the characteristic of the curves shown in these Figures. However, the SNR difference is reduced when quantization is added. As an example, Figure 5 contains two curves from the "renata" sequence, where the light grey curve is caused by a standard MPEG quantization. Finally, Figure 6 shows a picture pair (based on zigzag and scalability-optimized order) from the "renata" sequence sampled after 300 operations (representing low-cost applications). It can be seen that the scalability-optimized order results in an improved quality (compare sharpness and readability). A standard MPEG quantization has no visible effect here.

5. CONCLUSIONS

We have presented a new technique for computing the DCT coefficients in a re-ordered way to maximize the number of coefficients obtained within a given computation constraint. The optimization is achieved by analyzing each calculation step of a DCT algorithm to find the next coefficient that should be computed with minimum effort. The new DCT computation method provides a scalability-optimized order for a 2-D DCT algorithm, where two fast algorithms from literature having a low operations count were used.

Computer simulations have shown that the proposed new method maximizes the SNR of the picture during the computation by obtaining a high amount of DCT coefficients. Moreover, the achieved results show that the reconstruction quality grows consistently with the amount of available operations. The SNR obtained by using the scalability-optimized order surpasses the zigzag order used in MPEG coding with a 1-5 dB improvement, depending on the stage of the computation.

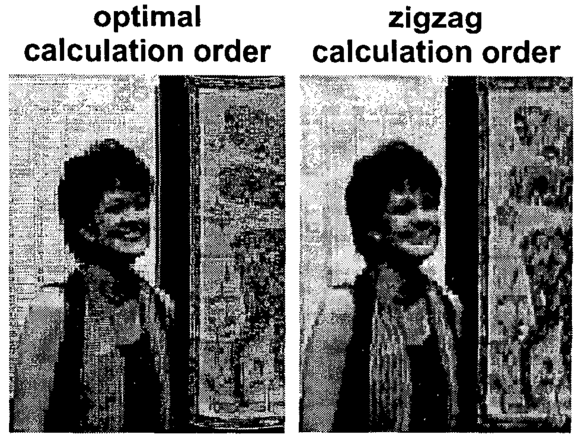

Fig. 6. A video frame from the "renata" sequence to compare the scalability-optimized order (left) and the zigzag order (right) Index $m(n)$ means m ops. for n coefficients.

The proposed technique is highly attractive for scalable MPEG encoding, where the quality for various portable applications should match with the constrained availability of resources. Ways for further improvement of the results have been indicated. For example, the method can be enhanced by various features such as a priority function, which e.g. favours the computation of low-frequency coefficients, so that it fits better with MPEG coding after performing a DCT. The technique can implemented for an IDCT as well.

6. REFERENCES

[1] P.Z. Lee and F.-Y. Huang, "Restructured Recursive DCT and DST Algorithms," *IEEE Trans. Sign. Proc.*, vol. 42, no. 7, pp. 1600–1609, July 1994.

[2] N.I. Cho and S.U. Lee, "Fast Algorithm and Implementation of 2-D Discrete Cosine Transform," *IEEE Trans. Circ. & Systems*, vol. 38, no. 3, pp. 297–305, Mar. 1991.

[3] Y. Arai, T. Agui and M. Nakajiama, "A Fast DCT-SQ Scheme for Images," *Trans. on IEICE*, vol. 71, no. 11, pp. 1095, Nov. 1988.

[4] N. Merhav and B. Vasudev, "A Multiplication-Free Approximate Algorithm for the Inverse Discrete Cosine Transform," *IEEE Proc. Int. Conf. Image Proc., Kobe, Japan*, Oct. 1999.

[5] I. Pao and M. Sun, "Modeling DCT Coefficients for Fast Video Encoding," *IEEE Trans. Circ. & Systems Video Technol.*, vol. 9, no. number 4, pp. 608–616, June 1999.