Abstract—A novel technique is proposed to reduce the computational complexity of discrete cosine transform (DCT)-based video encoders. The proposed method merges the DCT and quantization into a single procedure, which is referred to as the novel quantized DCT (NQDCT), such that the DCT output does not need to be explicitly quantized. Thus, a lot of computations related to quantization can be saved. The video encoder’s performance using NQDCT is evaluated by comparing it with those using the traditional separate DCT and quantization method and the quantized DCT (QDCT) method. Simulation results demonstrate that the proposed NQDCT outperforms the other two methods in improving the real-time performance for video encoding.

Index Terms—Discrete cosine transform (DCT), quantization, video encoder optimization.

I. INTRODUCTION

The discrete cosine transform (DCT) is widely used in a number of video coding standards, such as H.263 [1], H.264 [2], and MPEG-4 [3]. This is because the DCT exhibits a good energy compaction property, which is close to that of the optimum Karhunen–Loeve transform for the motion-compensated video signal. Therefore, after the DCT coefficients are quantized, high compression and low distortion can be achieved for video encoding. Fast algorithms [4] exist in the literature for the computations of DCT. However, even with an efficient fast DCT algorithm, the computational complexity of most video coding systems is still too high, which limits their applications. Various algorithms have been proposed to lower the computational complexity of video encoding. The quantized DCT (QDCT) [5] has shown to be an efficient algorithm that lowers the computational complexity of DCT-based coding systems. To distinguish from QDCT, we will use the notation DCT/Q to represent the two separate processes of DCT and quantization used in traditional video coding systems. The QDCT combines DCT with quantization into a single procedure and thus reduces the number of computations. However, the gain in computational efficiency through QDCT is traded off with the quality of the encoded video and the system memory requirements. For example, 31 different QDCT coefficient sets are required to be stored, which correspond to 31 possible quantization step sizes allowed by the MPEG-4 video encoder.

Even if multiple QDCT coefficient sets are used, the QDCT is still restricted to the scalar quantization process, which can be merged with DCT. The QDCT cannot be applied with nonuniform quantization matrices. In order to solve this problem, a novel quantized DCT (NQDCT) method is proposed in this letter, which allows merging the quantization process with a quantization matrix that has variable quantization step sizes into the DCT. Furthermore, appropriate scaling techniques are proposed to allow integer arithmetics to be used for NQDCT to gain further reduction in computational complexity.

The rest of this letter is organized as follows. The NQDCT is proposed in Section II to remedy the problem caused by QDCT. Simulation results are presented in Section III. Finally, Section IV concludes this letter.

II. PROPOSED NOVEL QUANTIZED DCT

A. Combination of DCT and Quantization

In this letter, we will consider the $8 \times 8$ DCT that has been widely used in the H.263 [1] and MPEG-4 [3] encoders. The two-dimensional (2D) $8 \times 8$ DCT coefficients $y_{uv} = [Y]_{u,v}$, $0 \leq u$, and $v \leq 7$ can be defined as

$$Y = C X C^T$$

where $x_{mn} = [X]_{m,n}$, $0 \leq m$, and $n \leq 7$ are either the image pixel intensity levels in intracoding mode or motion compensated image residues in intercoding mode; $C$ is the cosine basis transformation matrix given by

$$C = \begin{bmatrix}
C_4 & C_4 & C_4 & C_4 & C_4 & C_4 & C_4 & C_4 \\
C_1 & C_3 & C_5 & -C_7 & -C_5 & -C_3 & -C_1 & -C_4 \\
C_2 & -C_6 & -C_2 & -C_2 & -C_6 & -C_2 & -C_2 & -C_6 \\
C_3 & -C_7 & -C_3 & -C_5 & -C_5 & -C_3 & -C_1 & -C_4 \\
C_4 & -C_3 & -C_1 & -C_5 & -C_7 & -C_5 & -C_3 & -C_4 \\
C_5 & -C_5 & -C_1 & -C_3 & -C_7 & -C_5 & -C_1 & -C_5 \\
C_6 & -C_2 & -C_6 & -C_2 & -C_6 & -C_2 & -C_2 & -C_6 \\
C_7 & -C_5 & -C_3 & -C_1 & -C_3 & -C_5 & -C_5 & -C_7
\end{bmatrix}$$

where $C_n = \cos(n\pi/16)$, and $n = 1, \ldots, 7$. Quantization is then applied to $Y$ for compression using an $8 \times 8$ quantization matrix $Q$, which is either uniform (i.e., the quantization matrix has constant elements) or nonuniform. The quantized DCT outputs $y'_{i,j} = [Y']_{i,j}$ are given as

$$Y' = \left[ Y/s \right]$$

where $\div$ represents the element-to-element division between two matrices.
In [5], the QDCT is proposed to embed the quantization into DCT transformation. If a uniform quantization matrix is applied, the QDCT can be defined as

$$Y^q = \left( C^r_n \right) X \left( C^c_m \right)^T$$

(4)

where $C^r_n$ and $C^c_m$ are the corresponding quantized matrices for the row and column transforms, which are given as

$$C^r_n = C_n / Q^r_n$$

and

$$C^c_m = C_m / Q^c_m,$$

(5)

where $Q^r_n$ and $Q^c_m$ are intermediate DCT results; $Q^r_n$ and $Q^c_m$ are the corresponding quantized matrices for the row and column transforms, which are given as

$$C^r_n = C_n / Q^r_n$$

and

$$C^c_m = C_m / Q^c_m,$$

(5)

$\odot$ indicates that each element of $Q^j$ is multiplied by the element in the same position in the matrix $Q^j$. On the other hand, if the quantization matrix is nonuniform such as the MPEG-4 ‘method1’ quantization matrices [3], the QDCT as described above cannot be applied. Instead, the QDCT proposes to approximate $Q$ by

$$Q \approx q QT$$

(6)

where $q = [q_0, q_1, \ldots, q_j]^T$. Therefore, QDCT has the limitation that a factorizable quantization matrix is required; otherwise, the approximation by (6) will result in performance losses. When the nonuniform quantization matrices are used, we find that the fast row-column DCT algorithm [6] cannot be applied in QDCT. This is because the rows of $C$ in (2) are divided by the corresponding elements of $q$, for example, the third row of $C$ is divided by $q_3$; thus, $C$ cannot be described by only seven coefficients but by $22$ coefficients. Since the fast row-column algorithm [6] can be applied, depending on the structure of $C$, the modification of $C$ causes the fast row-column DCT algorithm to be no longer applicable to QDCT. As a result, QDCT is not suitable for fast video encoding when nonuniform quantization matrices are used.

B. Novel Quantized DCT

In order to solve the problem caused by QDCT, we propose a NQDCT method, which can be applied with both the uniform and nonuniform quantization matrices. Consider the last stage of the one-dimensional DCT signal flow graph in Fig. 1(a), which has the general form of

$$y_k = C_n a + C_m b$$

(7)

where $a$ and $b$ are intermediate DCT results; $C_n$ and $C_m$ are the corresponding cosine coefficients entering the last stage of the DCT signal flow graph. The DCT output $y_k$ is quantized by a quantization element $q$ as

$$y^q_k = \frac{(C_n a + C_m b)}{q} = \frac{(C_n)}{q} a + \frac{(C_m)}{q} b$$

(8)

As a result, the quantization can be embedded into DCT by using quantized cosine coefficients, such as $C^r_n$ and $C^c_m$ in (8), at the last stage of the DCT signal flow graph [see Fig. 1(b)]. As for the two-dimensional DCT, which is usually implemented by the fast row-column DCT algorithm [6], our NQDCT method only modifies the cosine coefficients at the last stage of the column transform by incorporating the elements of the quantization table. For example, considering the MPEG-4 ‘method1’-based intercoding and the quantization parameter $Q_p$, the final quantized DCT output $y^q_{k,j}$ in position $(i,j)$ after column transform can be written as

$$y^q_{k,j} = \left[ \left( \frac{16C_n}{2Q_p \times Q_{i,j}} \right) a + \left( \frac{16C_m}{2Q_p \times Q_{i,j}} \right) b \right] \odot \left[ C^r_n a + C^r_m b \right]$$

(9)

where $Q_{i,j}$ is the $(i,j)$th element of a nonuniform inter quantization matrix.

Therefore, when the nonuniform quantization matrix is applied, for each of the 64 quantized DCT outputs, two quantized cosine coefficients at the last stage of DCT column transform are precomputed and stored depending on $Q_{i,j}$ and $Q_p$. Hence, for each $Q_p$ value, a total of 128 quantized cosine coefficients must be stored. Because the fixed-point implementation is applied (see Section II-C), we can only store the quantized cosine coefficient sets corresponding to 16 odd $Q_p$ values: 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, and 31. Other sets can be easily obtained by using the shift operation, e.g., the quantized cosine coefficients corresponding to $Q_p = 28$ can be obtained by two right shifts of the quantized cosine coefficients corresponding to $Q_p = 7$. On the other hand, if the uniform quantization matrix is applied, for each $Q_p$ value, only seven quantized cosine coefficients are stored for use. Thus, not only the uniform but also the nonuniform quantization matrices can be applied in our NQDCT method. The proposed NQDCT method is described as follows.

Step 1) The row transform is performed using the normal DCT with the signal flow graph shown in Fig. 1(a).

Step 2) The column transform is applied with the modified DCT as shown in the signal flow graph in Fig. 1(b): using the quantized cosine coefficients to compute each of the 64 quantized DCT outputs.
C. Fixed-Point Implementation and Error Analysis

Similar to the fixed-point transformation in [7], the proposed NQDCT can be implemented by fixed-point operations to further reduce the computational complexity after appropriate scaling. Since NQDCT is different from DCT at the last stage of DCT computation, the fixed-point NQDCT can be obtained similar to DCT, except at the last stage.

The floating-point DCT cosine coefficients are transformed to the fixed-point form through scaling as

$$C_{n,f} = \lfloor C_n \times 2^s + 0.5 \rfloor$$

where $s$ determines the precision used in fixed-point DCT computations. Similarly, the fixed-point NQDCT cosine coefficients $C_{n,f}^q$ are given by

$$C_{n,f}^q = \left\lfloor \frac{C_n \times 2^s}{q} + 0.5 \right\rfloor.$$  \hspace{1cm} (11)

At the last stage of the column transform of DCT computation in NQDCT, the final quantized DCT output $y_f$ in (8) is given by

$$y_f = \left( C_{n,f}^q a + C_{m,f}^q b \right) \gg s$$ \hspace{1cm} (12)

where $\gg$ indicates the binary shift right. Compared with NQDCT, the final quantized DCT value $y_f^q$ by DCT/Q can be written as

$$y_f^q = \left\lfloor \frac{C_{n,f} a + C_{m,f} b}{q} \right\rfloor \gg s.$$ \hspace{1cm} (13)

We implement extensive experiments using benchmark video sequences based on the MPEG-4 video encoder [8] to test the difference between the final quantized DCT outputs $y_f^q$ and $y_f^q$. The results show that in most cases (95%), there is no difference between the final outputs of NQDCT and DCT/Q, while in all the other cases, the difference is 1. As a result, the encoded video qualities are almost the same for NQDCT and DCT/Q, and hence, no visual difference is observed in the video encoded by both methods.

III. SIMULATION RESULTS

In this letter, the XVID codec [8], which is an MPEG-4-compliant video codec, is implemented to evaluate the performance of the proposed NQDCT. Although extensive simulations have been performed with various benchmark video sequences, due to limited space, presented herein are the encoding results of two benchmark video sequences Foreman and Coast Guard, each of which is in CIF format. Both the MPEG-4 ‘method1’ and ‘method2’ quantization methods are applied to evaluate the proposed NQDCT.

A. Results Based on MPEG-4 ‘Method2’ Quantization

In MPEG-4 ‘method2’-based quantization, the uniform quantization matrix is applied. Therefore, we can compare our NQDCT with both the DCT/Q and QDCT methods. The encoding time spent in the DCT and quantization processes is studied. The comparison results demonstrate that both the NQDCT and QDCT methods can reduce the encoding time in DCT and quantization processes by around 40% when compared with the DCT/Q method.

The comparative results of the entire encoding time are shown in Fig. 2 for the Foreman sequence, where we can observe that the entire encoding time of NQDCT and QDCT are always lower than that of DCT/Q, as expected. Also, for some particular $Q_p$ values, QDCT has noticeable performance drop when compared with that of NQDCT. This is due to the reason that the calculation errors introduced by QDCT lead to an increase in the processing time of other procedures, such as the motion estimation (ME). The number of searching points per macroblock during ME for the Foreman sequence is plotted in Fig. 3. From this plot, we can see that the average number of searching points increases sharply for QDCT when $Q_p = 11, 13, 19, 23, 25, 27, 29$. This explains why there are some abnormal increases in the entire encoding time in Fig. 2 when the QDCT method is applied. The reason that the number of searching points increases is due to the fact that the error introduced by QDCT is so large that the reference frame reconstructed by IQ and IDCT is much different from the subsequent frame. As a result, it becomes more difficult for ME to locate the best matched block, and hence, the average number of searching points increases. On the other hand, our proposed NQDCT can reduce the entire encoding time while introducing negligible errors. Thus, the negative effects about NQDCT on other encoding procedures can be neglected, and
the proposed NQDCT is definitely better than the QDCT in improving the encoding real-time performance.

The rate-distortion (RD) curves are shown in Figs. 4 and 5. We can see from these two figures that the proposed NQDCT has almost the same RD performance as the original DCT/Q method, and both the DCT/Q and NQDCT methods have better RD performances than that of QDCT.

B. Results Based on MPEG-4 ‘Method1’ Quantization

In MPEG-4 ‘method1’-based quantization, the nonuniform quantization matrices are utilized. Therefore, the QDCT method cannot be applied. The proposed NQDCT method is only compared with the DCT/Q method. The improvements of the entire encoding time and the time consumed in DCT and quantization stages achieved by NQDCT have shown that the proposed NQDCT method can consistently reduce the encoding time in DCT and quantization processes by around 58% and reduce the entire encoding time by around 17% when compared with the DCT/Q method. The RD curves are shown in Figs. 6 and 7, where again we can see that the proposed NQDCT has almost the same RD performance as the DCT/Q method.

IV. CONCLUSION

In this letter, a novel method NQDCT is presented to reduce the computational complexity of DCT-based video encoders. The proposed NQDCT merges the quantization process into the DCT computation to speed up video encoding. Different from the QDCT method, the NQDCT permits the application of nonuniform quantization matrices, which cannot be applied with QDCT. Thus, the NQDCT is more flexible than QDCT. Simulation results have demonstrated that the proposed NQDCT can greatly reduce the encoding time and keep almost the same RD performance when compared with the DCT/Q method. Moreover, the NQDCT is superior to QDCT in terms of the encoding time and RD performance.

REFERENCES