INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11

CODING OF MOVING PICTURES AND ASSOCIATED AUDIO

ISO/IEC JTC1/SC29/WG11
MPEG93/283
Mar. 1993

Title: Protection from IDCT Mismatch

Purpose: Discussion

Source: Yasuo Katayama (GCT)

1 Introduction

There are two sources of IDCT mismatch: errors induced by finite precision implementations, and systematic errors realized in the algorithm itself.

In this paper IDCT mismatch due to algorithmic error is discussed, several protection methods are compared experimentally, and a simple protection of IDCT mismatch is proposed.

In the situation of high quality coding in MPEG-2, the conventional protection from IDCT mismatch (independent oddification of each coefficient in the de-quantization) is insufficient, and modification error cannot be ignored.

1.1 IDCT Mismatch

The two dimensional 8× 8 IDCT is defined as below, when $X_{k,l}$ are DCT coefficients and $x_{i,j}$ are values in real space.

$$x_{i,j} = (1/4) \sum_{k,l=0,0}^{7,7} C(k)C(l)X_{k,l} \cos(\frac{k(2i+1)\pi}{16}) \cos(\frac{l(2j+1)\pi}{16}) \tag{1}$$

$$C(n) = \begin{cases} 1/\sqrt{2} & \text{if (} n = 0 \text{)} \\ 1 & \text{(else)} \end{cases}$$

which can be written as below,

$$x_{i,j} = \sum_{k,l=0,0}^{7,7} C_{i,j,k,l} X_{k,l} \tag{2}$$

$$C_{i,j,k,l} = (1/4)C(k)C(l) \cos(\frac{k(2i+1)\pi}{16}) \cos(\frac{l(2j+1)\pi}{16})$$

and $C_{i,j,k,l}$ can be written as below, because k(2i+1) is zero only if k=0.

$$C_{i,j,k,l} = C_{k(2i+1)} C_{l(2j+1)}/4 \tag{3}$$

$$C_n = \begin{cases} 1/\sqrt{2} & \text{if (} n = 0 \text{)} \\ \cos(n\pi/16) & \text{(else)} \end{cases}$$

The $|C_n|$ are equal to one of the $C_0 - C_7$, and these are non-zero irrational numbers: $C_0 = 1/\sqrt{2}, C_4 = 1/\sqrt{2}$, $C_2 = \frac{1}{2}\sqrt{2+\sqrt{2}}$, $C_6 = \frac{1}{2}\sqrt{2-\sqrt{2}}$, $C_1 = \frac{1}{2}\sqrt{2+\sqrt{2+\sqrt{2}}}$, $C_7 = \frac{1}{2}\sqrt{2-\sqrt{2+\sqrt{2}}}$, $C_3 = \frac{1}{2}\sqrt{2+\sqrt{2-\sqrt{2}}}$, $C_5 = \frac{1}{2}\sqrt{2-\sqrt{2-\sqrt{2}}}$.

But in equation (3), $C_{i,j,k,l}$ has a product of two C_n.

Therefore $C_{i,j,k,l}$ can be rational numbers, especially a fraction over an even number. For example, when $X_{0,4} = \pm 4, \pm 12, \pm 20, ...$, $x_{0,0}$ can be the exact integer +0.5. This is one of the reasons for IDCT mismatch.

1.2 IDCT mismatch by single coefficient

The Four coefficients, $\{X_{0,0}, X_{4,0}, X_{0,4}, X_{4,4}\ \}$ have their $C_{i,j,k,l} = \pm 1/8(k = 0,4, l = 0,4)$, because $C_0 = C_4 = 1/\sqrt{2}$ as described before (Sign $\pm$ depends on i,j position).

Therefore these four coefficients can cause IDCT mismatch solitarily, (i.e. one of the four coefficients is non-zero and all other coefficients are zero), when its value is 8n + 4.

Any other coefficients have $C_{i,j,k,l}$ as non-zero irrational values, and therefore never cause mismatch alone.

1.3 Relation of 4 coefficients

The four coefficients, $\{X_{0,0}, X_{4,0}, X_{0,4}, X_{4,4}\}$ have the same absolute $C_{i,j,k,l}$, 1/8, and so are related as:

$$x_{i,j} = (X_{0,0} \pm X_{4,0} \pm X_{0,4} \pm X_{4,4})/8 + ...$$

This means that we can not decide separately whether four coefficients cause the mismatch.

Independent oddification may leave mismatches, typical examples of which are shown in Table 3.

1.4 Restriction of the relation

But actually, only 4 forms of 8 forms of $\pm$ in the upper equation are possible.

$C_{i,j,0,0} = C_0C_0/4$, $C_{i,j,4,0} = C_{8i+4}C_0/4$, $C_{i,j,0,4} = C_0C_{8j+4}/4$, $C_{i,j,4,4} = C_{8i+4}C_{8j+4}/4$, $|C_{8i+4}| = |C_{8j+4}| = C_0 = 1/\sqrt{2}$.

$C_{i,j,4,0}$ and $C_{i,j,4,4}$ change their signs simultaneously by changing the sign of C_{8i+4}, and $C_{i,j,0,4}$ and $C_{i,j,4,4}$ also change their signs simultaneously by changing the sign of C_{8j+4}.

Therefore we have only four equations those examples are shown as below,

$$\begin{aligned}
x_{0,0} =& \quad (X_{0,0} + X_{4,0} + X_{0,4} + X_{4,4})/8 + ...\\
x_{1,0} =& \quad (X_{0,0} - X_{4,0} + X_{0,4} - X_{4,4})/8 + ...\\
x_{0,1} =& \quad (X_{0,0} + X_{4,0} - X_{0,4} - X_{4,4})/8 + ...\\
x_{1,1} =& \quad (X_{0,0} - X_{4,0} - X_{0,4} + X_{4,4})/8 + ...
\end{aligned}$$

According to the above discussion, mismatches by 4 coefficients can be summarized as below, When one of I_i(i=0 − 3) in equations below is 8n+4, mismatches can appear. In fact, when all other coefficients are zero, a mismatch actually appears as shown in Table 3.

There is a simple protection method against this situation. If I_0 is oddified, other I_i(i=1 − 3) are oddified automatically, because the differences from I_0 are even values. (This is the method (6) described in the experimental section.)

$$\begin{aligned}
I_0 =& \quad X_{0,0} + X_{4,0} + X_{0,4} + X_{4,4}\\
I_1 =& \quad X_{0,0} - X_{4,0} + X_{0,4} - X_{4,4}\\
I_2 =& \quad X_{0,0} + X_{4,0} - X_{0,4} - X_{4,4}\\
I_3 =& \quad X_{0,0} - X_{4,0} - X_{0,4} + X_{4,4}
\end{aligned}$$

As described before, any other non-zero coefficients alone have a function of protection of mismatch rather than causes of mismatch. However, two or more non-zero coefficients do not protect mismatch, but rather can be a cause of mismatch, which is discussed in the next section.

1.5 Mismatches by two coefficients

A rare but nevertheless impossible to ignore phenomena is mismatches by two coefficients, that have an equal value, which is to be called "coefficient pairing".

The most frequent three examples are ($X_{1,3}$ and $X_{3,1}$), ($X_{1,5}$ and $X_{5,1}$), and ($X_{3,5}$ and $-X_{5,3}$).

- The value of $(C_{i,j,1,3}+C_{i,j,3,1})$ is exactly 1/8 when (i,j)is (1,0) etc., so when $X_{1,3} = X_{3,1}$ is 8n+4, a mismatch can appear.

$$C_{1,0,1,3} + C_{1,0,3,1} = (C_3C_3 + C_9C_1)/4 = ({C_3}^2 - C_7C_1)/4 = 1/8$$

- With the same reason, $(C_{i,j,1,5} + C_{i,j,5,1})$ is $-1/8$ when (i,j)is (1,0)etc., so when $X_{1,5} = X_{5,1}$ is 8n+4, a mismatch can appear.

$$C_{1,0,1,5} + C_{1,0,5,1} = (C_3C_5 + C_{15}C_1)/4 = (C_3C_5 - {C_1}^2)/4 = -1/8$$

- $(C_{i,j,3,5}-C_{i,j,5,3})$ is just $-1/4$ when (i,j) is (3,0) etc., so when $X_{3,5} = -X_{5,3}$ is 4n+2, a mismatch can appear.

$$C_{3,0,3,5} + C_{3,0,5,3} = (C_{21}C_5 - C_{35}C_3)/4 = (-{C_5}^2 - {C_3}^2)/4 = -1/4$$

When pairs of coefficients in Table 1 have an equal value of 8n+4, there can be a mismatch. The relevant equation is one of the six below: (Suffixs in Table 1 are exchangeable, if $X_{k,l}$ and $X_{m,n}$ make a pair, $X_{l,k}$ and $X_{n,m}$ also make a pair.)

$$\begin{aligned} {C_6}^2 + C_2C_6 &= 1/2, & {C_2}^2 - C_2C_6 &= 1/2 \\ {C_7}^2 + C_3C_5 &= 1/2, & {C_1}^2 - C_3C_5 &= 1/2 \\ {C_5}^2 + C_1C_7 &= 1/2, & {C_3}^2 - C_1C_7 &= 1/2 \end{aligned}$$

Table 1. Pairing Coefficients

$X_{1,3}$	$X_{3,1}, X_{5,7}$
$X_{1,5}$	$X_{5,1}, -X_{3,7}$
$X_{3,7}$	$X_{7,3}$
$X_{5,7}$	$X_{7,5}$
$X_{1,1}$	$-X_{3,5}$
$X_{2,2}$	$\pm X_{2,6}$
$X_{3,3}$	$-X_{1,7}$
$X_{5,5}$	$X_{1,7}$
$X_{6,6}$	$\pm X_{2,6}$
$X_{7,7}$	$X_{3,5}$

When pairs of coefficients in Table 2 have an equal value of 4n+2, there can be a mismatch. The relevant equation is one of the three below: (Table 2 does not include four coefficients. Suffixs of Table 2 are also exchangeable.)

$${C_1}^2 + {C_7}^2 = 1, \qquad {C_2}^2 + {C_6}^2 = 1, \qquad {C_3}^2 + {C_5}^2 = 1$$

Table 2. Even Pairing Coefficients

$X_{1,1}$	$X_{7,7}$
$X_{2,2}$	$X_{6,6}$
$X_{3,3}$	$X_{5,5}$
$X_{1,5}$	$X_{7,3}$
$X_{1,3}$	$-X_{7,5}$
$X_{1,7}$	$-X_{7,1}$
$X_{2,6}$	$-X_{6,2}$
$X_{3,5}$	$-X_{5,3}$

1.6 Mismatches by three or four coefficients

Three coefficients can cause mismatches (for example, $X_{4,2} = -X_{5,3} = X_{5,5}$ and, $X_{0,2} = -X_{1,3} = X_{1,5}$ etc). Four coefficients can cause a mismatch when there are two even pairs (each of odd value).

2 Experiments on IDCT mismatch

Test conditions:

- TM2, M= 3, Frame structure
- Frame/field prediction and frame/field DCT
- Rate Control: includes Step 3
- Sequence: FG, MC, BC, FT, and BS (0 - 149)
- Bitrate: 4 Mb/s, 8 Mb/s and 16 Mb/s

2.1 Detection of Algorithmic IDCT mismatches

IDCT calculations are done via double-precision arithmetic. Algorithmic IDCT mismatches are detected by the following inequality, and mismatches are detected when there are some $x_{i,j}$ in the section below,

$$(integer + 0.5 - 10^{-10}) < x_{i,j} < (integer + 0.5 + 10^{-10})$$

This inequality detects true algorithmic mismatches because the inequality below has the exact same result as above inequality at no protection 4 Mb/s.

$$(integer + 0.5 - 10^{-9}) < x_{i,j} < (integer + 0.5 + 10^{-9})$$

2.2 Protections from IDCT mismatch

(1) No Protection

(2) DC only oddification

(3) 4 coefficients oddification

(4) All coefficients oddification (the conventional protection)

(5) Modification only DC by 1 to oddify sum of all coefficients

(6) Modification only DC by 1 to oddify sum of the 4 coefficients

(7) Modification only DC by 1 to oddify sum of the 4 coefficients and protecting against 2 pairs of coefficents ($X_{1,3} = X_{3,1}$ and $X_{1,5} = X_{5,1}$), with including two comparisons and conditional additions (or exclusive OR).
if($X_{1,3} == X_{3,1}$) sum = sum + $X_{1,3}$
if($X_{1,5} == X_{5,1}$) sum = sum + $X_{1,5}$

3 Results

The number of blocks for 150 frames in which mismatches are detected is shown in Table 1.

- (3) and (4) give nearly equal results. And (5) and (6) give nearly equal results.
- (5) and (6) are better than (3) and (4).
- (5),(6) and (7) are methods whose MSE $\leq 1/64$ because they modify one coefficient by only 1.
- (5) and (6) are simple and effective for protecting against IDCT mismatch.
- Which method is simpler, depends on the architecture of data processing. That is, (5) is simpler when all coeff. are processed successively, (it needs one bit register and one exclusive OR) and (6) is simpler when all coeff. are processed in parallel. (it needs 3 exclusive ORs.)
- (6) is the essential protection, and so can be extended, such as to (7).

Table 1. The number of blocks in which detected IDCT mismatch

Table 1.a (4 Mb/s)

Methods	FG	MC	BC	FT	BS
(1)No Protection	2276	1516	6283	6957	4747
(2)DC only oddify	860	848	1481	687	885
(3)4 coeff oddify	361	298	620	1505	904
(4)All coeff odd	343	256	651	1507	851
(5)sum of all	4	8	2	2	4
(6)sum of 4	1	11	1	1	2
(7)4 with 2 pairs	0	0	0	0	0

Table 1.b (8 Mb/s)

Methods	FG	MC	BC	FT	BS
(1)No Protection	6888	11447	9980	19835	10519
(2)DC only oddify	705	695	646	2478	1894
(3)4 coeff oddify	876	699	1147	2905	2206
(4)All coeff odd	914	719	1197	2963	2189
(5)sum of all	1	9	4	10	23
(6)sum of 4	4	19	3	8	19
(7)4 with 2 pairs	0	0	0	0	0

Table 1.c (16 Mb/s)

Methods	FG	MC	BC	FT	BS
(1)No Protection	2975	3086	7359	2847	2875
(2)DC only oddify	1293	1785	1007	1817	2178
(3)4 coeff oddify	1311	1845	1174	1867	2060
(4)All coeff odd	1331	1836	1207	1848	2053
(5)sum of all	17	19	6	29	30
(6)sum of 4	10	17	5	20	19
(7)4 with 2 pairs	0	0	0	1	1

The maximum luminance SNR achievable is shown in Table 2, which is in the situation of perfect reconstruction of coefficients without quantization. The method (4) decreases maximum SNR about 8 dB and the methods (5) and (6) slightly decrease maximum SNR (-0.7dB).

Table 2. Maximum SNR for each protection

Methods	FG	MC	FT
(1)No Protection	58.92	58.91	58.92
(4)All coeff odd	51.32	50.62	51.50
(5)sum of all	58.23	58.21	58.21
(6)sum of 4	58.23	58.23	58.21

The effect on IDCT mismatch of the above algorithmic errors have not yet been compared with those due to finite-precision errors. If it is the case that finite-precision errors are large compared with these algorithmic errors, we might ignore them. However, if algorithmic errors have the same or greater order than precision effects, we cannot ignore them.