CCITT Study Group XV
Working Party XV/1
Specialists Group on Coding for Visual Telephony

Doc.# 153
November 1986

Source: NTT, KDD, NEC, FUJITSU

Title: MOTION VECTOR CODING

---

1. Introduction

This document discusses motion vector encoding methods. First, absolute and relative(inter block differential) forms are compared. Then, comparison is made with respect to motion vector code sets.

2. Absolute and Relative Expression for Motion Vectors

Two motion vector encoding methods, absolute and relative are compared with respect to information amount for motion vectors.

Simulation results for RM2 are shown in tables on pp.3-4. For "Checked Jacket" as an example of moderate motion, the absolute method is more efficient than the relative method. For active picture "Split-Trevor", the relative method is more efficient, though it was reported in Doc.# 88 (FRG) that the relative method produced a greater information amount when Jain and Jain's method was used for the sequence.[1] When RM2 is used, the difference is small with respect to both entropy and mean code length of Huffman codes, for the distribution of these motion vectors. For most active picture such as panning, efficiency of the relative method is considered to be much better.

In addition, the relative method may be more useful when future technique such as object matching is employed.

Therefore, it is confirmed that the relative method is more suitable than the absolute one for motion vector encoding.

3. One-dimensional Encoding Method vs Two-dimensional Method

Three encoding methods are compared in what follows.

A. Two-dimensional Coding
A motion vector is represented with a single code word.

B. One-dimensional Coding(1)
A motion vector is represented by two code words. Horizontal component and vertical component of motion vector are coded with different code sets.

C. One-dimensional Coding(2)

A motion vector is represented by two code words. Horizontal component and vertical component of a motion vector are coded with the same code set.

In table on pp.3-4, simulation results are shown for RM2. With respect to entropy value and mean code length, the two-dimensional code is slightly more efficient than the one-dimensional code(1). The latter is slightly more efficient than the one-dimensional code(2).

Both differences are very small, less than 100 bits per frame. One-dimensional code(2) is simplest of the three and can be designed most easily.

Therefore, it will be better to use this one dimensional code(2).

4.Conclusion

From the view point of efficiency and future expandability, it has been confirmed that the relative vector encoding method is more suitable than the absolute method.

For simplicity, a one-dimensional coding method is proposed, where horizontal component and vertical component of motion vector are coded with the same code set.

[1] Document # 88 (Federal Republic of Germany) in Tokyo, 1986.

SIMULATION RESULT

In Table 1, entropy values are calculated for each method. Calculation method is as follows.

A. Entropy of two-dimensional coding:
Entropy is calculated from the two-dimensional distribution of motion vectors excluding zero vectors.

B. Entropy of one-dimensional coding(1):
Entropy value is calculated each from the distribution of vertical or horizontal motion vector component Sum of the two entropy value is evaluated.

C. Entropy of one dimensional coding(2):
Entropy is calculated from the distribution of motion vectors, where vertical and horizontal components are combined. Sum of the two entropy is evaluated.

Three statistics used in entropy calculation are generated using Reference Model 2.

In Table 2, mean code lengths are calculated. Code sets are designed with the distribution of motion vectors, which is generated using Reference Model 2 for "Miss America", "Checked Jacket" and "Split-Trevor" all together. Code length examples for designed code sets are shown in Table 3.

TABLE 1: ENTROPY OF MOTION VECTORS

Absolute motion vectors

	Two dimensional entropy	One dimensional entropy(1)	One dimensional entropy(2)
Miss America	6.34	6.53(=3.44+3.09)	6.62(=3.31*2)
Checked Jacket	5.83	6.09(=2.75+3.34)	6.18(=3.09*2)
Split Screen	7.14	7.27(=3.63+3.64)	7.30(=3.65*2)
Trevor	7.26	7.42(=3.64+3.79)	7.46(=3.73*2)
Average	6.89	7.04(=3.54+3.50)	7.06(=3.53*2)

Relative motion vectors

	Two dimensional entropy	One dimensional entropy(1)	One dimensional entropy(2)
Miss America	6.50	6.68(=3.37+3.31)	6.68(=3.34*2)
Checked Jacket	6.24	6.46(=2.97+3.49)	6.54(=3.27*2)
Split Screen	6.93	7.08(=3.55+3.54)	7.08(=3.54*2)
Trevor	7.07	7.22(=3.57+3.65)	7.22(=3.61*2)
Average	6.76	6.92(=3.45+3.48)	6.92(=3.46*2)

TABLE 2: MEAN CODE LENGTH OF MOTION VECTORS

Absolute motion vectors

	Two dimensional entropy	One dimensional entropy(1)	One dimensional entropy(2)
Miss America	6.57	6.77(=3.54+3.23)	6.79
Checked Jacket	6.25	6.37(=2.96+3.41)	6.37
Split Screen	7.31	7.44(=3.71+3.73)	7.48
Trevor	7.46	7.67(=3.74+3.93)	7.63

Relative motion vectors

	Two dimensional entropy	One dimensional entropy(1)	One dimensional entropy(2)
Miss America	6.59	6.81(=3.43+3.38)	6.81
Checked Jacket	6.40	6.65(=3.10+3.55)	6.65
Split Screen	7.01	7.16(=3.60+3.57)	7.17
Trevor	7.14	7.30(=3.60+3.70)	7.31

Table 3: Code length examples of motion vectors

3-1 Two dimensional code for absolute motion vector

vertical \ horizontal	0	+-1	+-2	+-3	+-4	+-5	+-6	+-7
+-7	8	9	9	9	9	9	10	9
+-6	8	9	9	9	9	10	10	10
+-5	7	8	9	9	9	10	10	10
+-4	7	8	9	9	8	10	10	10
+-3	7	7	8	8	8	9	10	10
+-2	6	7	7	8	8	10	10	10
+-1	5	5	6	7	8	8	10	9
0		5	6	6	6	8	8	8

(bit)

3-2 One dimensional code(1) for absolute motion vector

	0	+-1	+-2	+-3	+-4	+-5	+-6	+-7
horizontal	2	3	4	4	4	5	6	6
vertical	2	3	4	4	5	5	5	5

(bit)

3-3 One dimensional code(2) for absolute motion vector

0	+-1	+-2	+-3	+-4	+-5	+-6	+-7
2	3	4	4	4or5	5	6	6

(bit)

3-4 Two dimensional code for relative motion vector

vertical \ horizontal	0	+-1	+-2	+-3	+-4	+-5	+-6	+-7
+-7	7	8	9	9	9	10	10	8
+-6	8	9	10	10	9	10	10	10
+-5	8	8	9	10	9	10	10	10
+-4	7	8	8	9	9	10	10	10
+-3	6	8	8	8	9	10	10	9
+-2	6	7	7	8	8	10	10	9
+-1	4	5	7	7	8	8	9	8
0		5	6	7	7	8	8	7

(bit)

3-5 One dimensional code(1) for relative motion vector

	0	+-1	+-2	+-3	+-4	+-5	+-6	+-7
horizontal	2	3	4	4	4	5	5	5
vertical	2	3	4	4	5	5	6	4or5

(bit)

3-6 One dimensional code(2) for relative motion vector

0	+-1	+-2	+-3	+-4	+-5	+-6	+-7
2	3	4	4	5	5	5	5

(bit)