CCITT SGXV
Working Party XV/1
Specialist Group on Coding
for Visual Telephony

Doc # 201
March 1987

TITLE: DIFFERENTIAL MOTION VECTORS 2.

SOURCE: UK

OBJECTIVE.

The merits of using a differential motion vector scheme within the 'Okubo Reference Model 3' are further examined. This is compared with an absolute value scheme. Also a modified reference model is used where a full search motion estimation algorithm is employed.

DIFFERENTIAL VECTORS.

Within the hardware specification document [1] the use of differential motion vectors is detailed but not fully specified. An earlier specialist group document [3] describes the study of differential motion vectors within the 'Reference Model 2'. Also refer to [2] for more detail.

CODING OF VECTORS.

Assuming that the absolute motion vectors have a dynamic range of +/-7 then the differential vectors have a dynamic range of +/- 14. It is possible to modify the requirements of the differential vector coding by using the knowledge that any absolute vector must be with the range of +/- 7. Several possible techniques can be used to perform this truncation.

Method 1

This modified coding is known here as truncated differential vector coding (TDVC). It was assumed in [2] that this method was as used in [3], though it appears that this may not be the case. An example illustrates the method. If the previous absolute vector is +7 and the current absolute vector is -7 then the differential is -14. If one assumes that the maximum differential transmitted is +/- 7 then if one were to transmit +7 (which is the inverted absolute vector) in this case the decoder would normally calculate a the current vector as +14. However it is known that the maximum absolute vector is +/7 thus the decoder may re-calculate the vector by inverting the sign of the apparent differential and use this value as the absolute value of the current vector. A few more examples illustrates the mechanism.

Previous vector	Current vector	Diff. vector	tx/rx vector	decoded vector	decoded and modified
7	-7	-14	7	+14	-7
-7	6	+13	-6	-13	+6
-6	7	+13	-7	-13	+7

Method 2

A translation may be obtained for differential vectors with a dynamic range of +/- 14 to +/- 7 by first detecting whether the magnitude of the differential vector is greater than 7, if so then 15 is either added or subtracted such that the resultant vector is in the range +/- 7. The following illustrates the principle.

Differential Vectors		Tx Vector
-14		1
.	+15	.
.		.
-8		7

At the decoder if the decoded vector is found to lie outside the range +/- 7 then 15 is added or subtracted to force the differential to lie within the allowable range.

Method 3

The differential vector range can be limited to -8/+7. The method is identical to Method 2 but '16' is either added or subtracted and the differential thresholds are '+7' and '-8' instead of '7'.

ROBUSTNESS OF CODING

In [2] no attempt was made to assertain the robustness of the codes generated. For each of the sequences used, the associated optimal word allocation was used on the remaining sequences and the resulting efficiency of the coding was assessed. Here the robustness of the codes generated by method 1 (above) are illustrated.

OTHER MOTION ESTIMATION ALGORITHMS

The motion estimation algorithm used in [2] was a 3 stage algorithm which is computationally efficient but not particularly good from a best prediction viewpoint. Here the results of using a full search estimation algorithm is examined. The coding algorithm remains identical to the Reference model 3 except for the motion estimation.

RESULTS

The 'Costi' and 'Miss America' sequences were used to generate results for this document. The results shown represent an average result using the data indicated.

Truncation of Differential Vectors

The results for Method 1 are illustrated in [2].

The entropies and mean word lengths generated Method 2 almost identical to Method 1.

The entropies and mean word lengths generated by Method 3 were marginally worse than Method 1. Typically the entropies and mean word lengths were 0.05 bits greater.

Coding robustness

Table 1 illustrates the change in mean word length (in bits) compared to the minimum mean word length generated by using an optimal code for that sequence. Typically there is less than a 3% loss in coding efficiency of the differential vectors.

Full Search Motion Estimation

Figures 1 to 6 illustrate the distibutions for absolute vectors (Figs 1-2), differential vectors (Figs 3-4) and truncated differential vectors (Figs 5-6) (Method 1) using a full search motion estimation algorithm. These are all similar to the distributions shown in [2]. Table 2 illustrates the entropies generated.

The entropy measures shown in Table 2 can usually never be attained with 'real codes' in practice and therefore some estimate of the attainable bit usage per vector is now given. This is calculated as:

$$\text{Bits required for } V(n) = \text{INT}(-\text{Log}_2 (P(n)) + 0.5)$$

Where V(n) is the vector 'n'.
P(n) is the probability of V(n) occuring
INT(X) implies integer of (X).

Whilst this technique for bit allocation is not truely optimal it is thought to be sufficiently accurate for guidance purposes. For each case examined a bit allocation was generated and a subsequent mean word length for the vectors was calculated. The mean word length (MWL) was calculated using:

$$\text{MWL} = P(n)^{*}\text{INT}(-\text{Log}_2 (P(n)) + 0.5)$$

Where V(n) is the vector 'n'.
P(n) is the probability of V(n) occuring
INT(X) implies integer of (X).

Table 3 illustrates the mean word lengths generated for absolute vectors, differential vectors and truncated differential vectors (Method 1)

DISCUSSION

In general the results illustrated here are similar to the results of [2].

Truncation of Differential Vectors

There was found to be little difference for the three methods examined. Whilst Method 3 appears to be marginally worse in terms of performance than the others it has a significant advantage in hardware realisation, in that no additional processing is required after the generation of the differential vector. By taking the 3 least significant bits of the resulting vector one automatically has the processed result.

Coding robustness

The codes were found to be adequately robust considering the difference in nature of the source material.

Full Search Motion Estimation

The use of full search motion compensation indicated that usiing truncated differential vectors would be worthwhile. Typically 300 bits per frame could be saved by using this technique. This saving of bits is calculated based only on the gains obtainable by differentially coding the motion vectors and not on the gain in coding efficiency by using a better motion estimation algorithm. It is assumed that (in RM3) absolute vectors require 4 bits per vector (or 8 bits per block). Refering to [2] it is clear that the motion estimation algorithm affects the performance of any differential vector coding scheme.

It is clear that the overall efficiency of using relative motion vectors within the 'Reference Model 3' yields only a small gain in coding efficiency (approx 1% overall with full search motion estimation).

CONCLUSION

The current specification allows the provision of differential vectors of two types. The first is a differential with respect to two absolute block vectors. The second is a differential with respect to a global vector and a block absolute vector. If the global vector is zero then the specification has in effect provision to code the motion vectors in a differential or absolute manner. The problems of error propagation can then be investigated more fully in realistic conditions.

Therefore the current recommendation in [1] should remain until further more substantive evidence can be provided.

REFERENCES

[1].......Doc #182 CCITT WP XV/1 Specialist Group on Video Telephony Nov 86 Nurnberg.

[2].......Doc #200 CCITT WP XV/1 Specialist Group on Video Telephony March 87 USA.

[3].......Doc #153 CCITT WP XV/1 Specialist Group on Video Telephony Nov 86 Nurnberg.

		SEQUENCE USED TO GENERATE OPTIMAL CODE		
SEQUENCE CODED	CODED FRAMES	MISSA	SPLIT SCREEN	TREVOR
SPLIT SCREEN	2 – 19	0.0	0.035	–0.12
TREVOR	21 – 48	0.086	0.0	–0.08
MISSA	ALL	0.17	0.099	0.0

TABLE 1 CHANGE IN MEAN WORD LENGTH (BITS)

		ENTROPY (BITS)		
SEQUENCE	CODED FRAMES	ABSOLUTE VECTORS	DIFFERENTIAL VECTORS	TDV
SPLIT SCREEN	2 – 19	3.598	3.398	3.19
TREVOR	21 – 48	3.68	3.47	3.25
MISSA	ALL	3.029	2.75	2.68

TABLE 2 ENTROPY OF VECTORS

		MEAN WORD LENGTH (BITS)		
SEQUENCE	CODED FRAMES	ABSOLUTE VECTORS	DIFFERENTIAL VECTORS	TDV
SPLIT SCREEN	2 - 19	4.098	3.898	3.69
TREVOR	21 - 48	4.18	4.18	3.75
MISSA	ALL	3.52	3.25	3.18

TABLE 3 MEAN WORD LENGTHS OF VECTOR CODES

FIGURE: 1

DATE: 9/2/87
SOURCE: BTRL UK
AUTHOR: G SEXTON

CODER SOFTWARE: CODEC8 V1.1 (OKUBO REFERENCE MODEL3 MODIFIED). RM3V3 V1.0.
MODIFICATIONS: FULL SEARCH MC
SOURCE SEQUENCE: COSTI CIF. SPLIT SCREEN ONLY
FRAME SIZE: 352 * 288
BITRATE: 300KBITS/SEC
BUFFER SIZE: 30000

FRAME RATE: 10 FRAMES / S
CODED FRAMES: 19

PLOT OF ABSOLUTE MOTION VECTOR USAGE

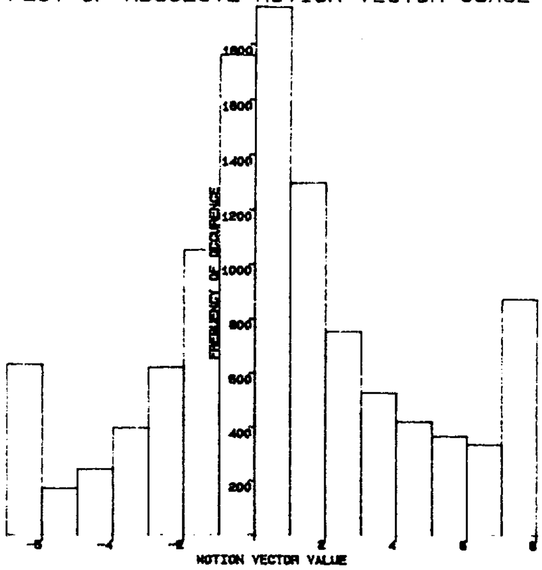

FIGURE: 2

DATE: 9/2/87
SOURCE: BTRL UK
AUTHOR: G SEXTON

CODER SOFTWARE: CODEC8 V1.1 (OKUBO REFERENCE MODEL3 MODIFIED). RM3V3 V1.0.
MODIFICATIONS: FULL SEARCH MC
SOURCE SEQUENCE: COSTI CIF. TREVOR ONLY
FRAME SIZE: 352 * 288
BITRATE: 300KBITS/SEC
BUFFER SIZE: 30000

FRAME RATE: 10 FRAMES / S
CODED FRAMES: 28

PLOT OF ABSOLUTE MOTION VECTOR USAGE

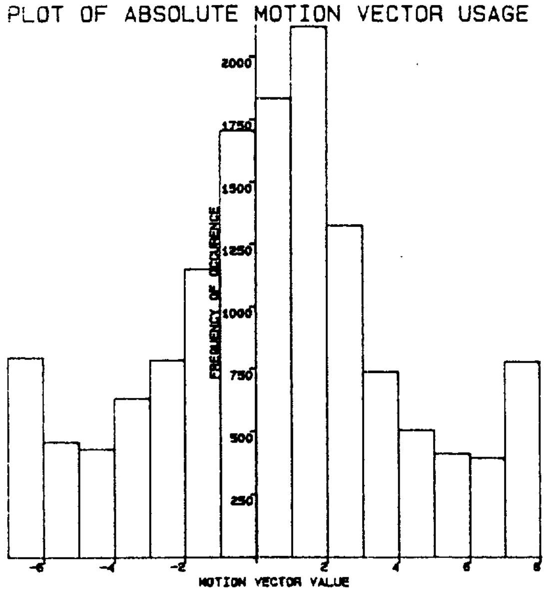

FIGURE: 3

DATE: 9/2/87
SOURCE: BTRL UK
AUTHOR: G SEXTON

CODER SOFTWARE: CODEC8 V1.1 (OKUBO REFERENCE MODEL3 MODIFIED). RM3V3 V1.0.
MODIFICATIONS: FULL SEARCH MC
SOURCE SEQUENCE: COSTI CIF. SPLIT SCREEN ONLY
FRAME SIZE: 352 * 288
BITRATE: 300KBITS/SEC
BUFFER SIZE: 30000

FRAME RATE: 10 FRAMES / S
CODED FRAMES: 19

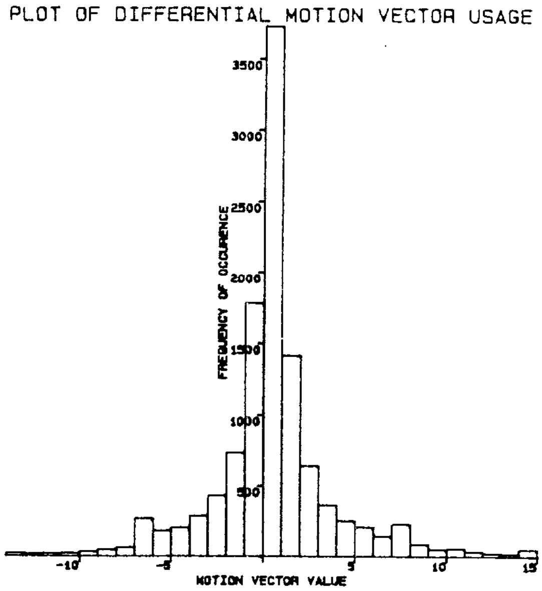

FIGURE: 4

DATE: 9/2/87
SOURCE: BTRL UK
AUTHOR: G SEXTON

CODER SOFTWARE: CODEC8 V1.1 (OKUBO REFERENCE MODEL3 MODIFIED). RM3V3 V1.0.
MODIFICATIONS: FULL SEARCH MC
SOURCE SEQUENCE: COSTI CIF. TREVOR ONLY
FRAME SIZE: 352 * 288
BITRATE: 300KBITS/SEC
BUFFER SIZE: 30000

FRAME RATE: 10 FRAMES / S
CODED FRAMES: 28

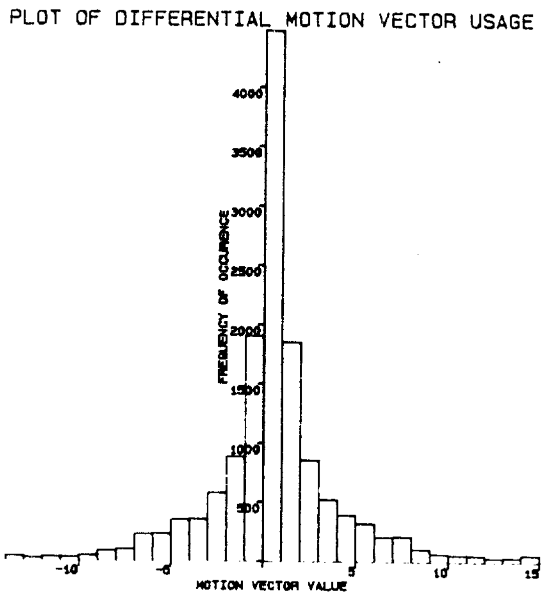

FIGURE: 5

DATE: 9/2/87
SOURCE: BTRL UK
AUTHOR: G SEXTON

CODER SOFTWARE: CODEC8 V1.1 (OKUBO REFERENCE MODEL3 MODIFIED). RM3V3 V1.0.
MODIFICATIONS: FULL SEARCH MC
SOURCE SEQUENCE: COSTI CIF. SPLIT SCREEN ONLY
FRAME SIZE: 352 * 288
BITRATE: 300KBITS/SEC
BUFFER SIZE: 30000

FRAME RATE: 10 FRAMES / S
CODED FRAMES: 19

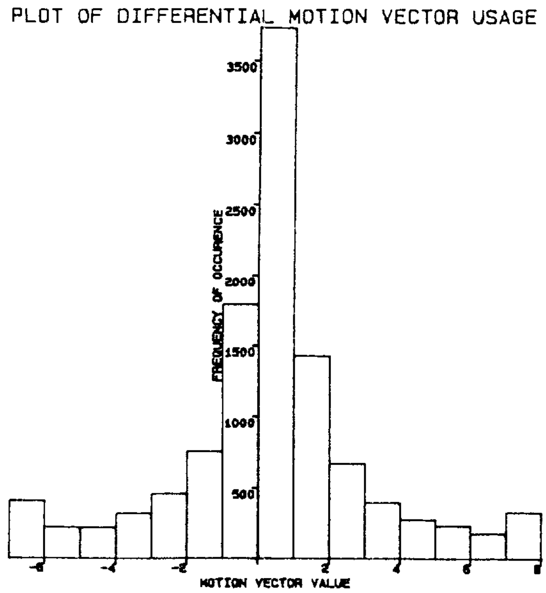

FIGURE: 6

DATE: 9/2/87
SOURCE: BTRL UK
AUTHOR: G SEXTON

CODER SOFTWARE: CODEC8 V1.1 (OKUBO REFERENCE MODEL3 MODIFIED). RM3V3 V1.0.
MODIFICATIONS: FULL SEARCH MC
SOURCE SEQUENCE: COSTI CIF. TREVOR ONLY
FRAME SIZE: 352 * 288
BITRATE: 300KBITS/SEC
BUFFER SIZE: 30000

FRAME RATE: 10 FRAMES / S
CODED FRAMES: 28