CCITT SGXV
Working Party XV/4
Specialists Group on Coding for Visual Telephony

Document #339
March 1988

TITLE: Description of Ref. Model 5 (RM5)

SOURCE: The Netherlands, France, FRG, Italy, Sweden, BTRL

version : March 17, 1988

CONTENTS

1 INTRODUCTION

The scope of this document is twofolded, first the modification of reference model 4 towards reference model 5 is given after which Reference model 5 is completely described.

Reference Model 5 (RM5) is a configuration which has the ability to operate at px64 kbit/s (p=1,..,30)

The objective of this document is to use it as a guideline, new ammendments will be added resulting in a new reference model.

The readers are asked to give comments and corrections to remove ambiguous parts.

2 MODIFICATIONS TO THE REFERENCE MODEL

1. Macro blocks.

   A macro block is defined as a 16·16 Y-block (four 8·8 blocks), an 8·8 U block and a 8·8 V block, corresponding to the same physical picture area.

2. Addressing of macro blocks.

   A non-fixed macro block is addressed with a similar relative addressing as used in RM4.

3. Macro block types.

   Non-fixed macro blocks can be of three types, which are signalled with the following VLC.

MC, coded	1
MC, fixed	01
coded	00

   With "coded" is meant that at least one of the six 8·8 blocks have non-zero transform coefficients. For "coded" macro blocks six EOB words are transmitted (Same 2-D VLC as in RM4).

4. Motion compensation.

   One motion vector is transmitted for each "MC" macro block.

5. Motion compensation of chrominance.

   The motion vector is applied also on the chrominance. To cope with the lower resolution, the Y-MV is devided by two with truncation to integer value.

   Example. Y-MV = (+3,+2) => U,V-MV = (+1,+1)
   Y-MV = (-5,-6) => U,V-MV = (-2,-3)

   (SNR measurements on U and V should be presented in the future).

6. Filter in the loop.

   The filter is applied within 8x8 Y block as well as within the 8x8 U and V blocks.

7. Signalling of filter in the loop.

Filter is applied in "MC" macro blocks.
(No extra side information)

8. Intra.

Intra mode is only used for the first image. The mean values of Y,U and V are first transmitted with 8 bits each (24 bits per macro block). Mean values are calculated with rounding (1.49 = 1 , 1.5 = 2) The residual between original picture and mean value reconstruction is transmitted with inter coding (both DC and AC).
Scene cuts are not handled separately.

9. Quantizer.

The quantizer can have the step sizes 4,6,8,...,64.
(31 different values)

10. Choise of quantizer.

The same quantizer is used in the whole GOB (18 GOBs per picture). Quantizer is selected from buffer content at the start of the GOB according to the following formula.

$$step = 2 * INT(buf\_cont / 200) + 2$$

11. Quantizer at scene cut.

For the first GOB in a scene cut picture the step size is forced to be minimum 16.

12. Buffer overflow.

If the buffer overflows during coding of a macro block, the next macro block is forced to be fixed. (This may give some small buffer overflows, which are not critical for simulations)

13. Number of GOBs.

There are 18 GOBs in a picture, each having 22 macro blocks.

14. Bit rate.

To simulate p·64 kbit/s the following number of bits are used.

30 Hz	p·5	bits / macro block
15 Hz	p·10	bits / macro block
10 Hz	p·15	bits / macro block
7.5 Hz	p·20	bits / macro block

This number of bits are subtracted from the buffer content after each macro block. (For p=1 the simulation bit rate becomes 59.4 kbit/s.)

15. Buffer size.

Buffer size is derived from the bit rate (p·64 kbit/s).

$$\text{buffer size} = p \cdot 6400 \text{ bits}$$

3 DESCRIPTION OF REFERENCE MODEL 5

In the proceeding text reference model 5 will be described. First starting with the basic format parameter choice referred to as common source input format. The spatial sizes are specified where these are most critical where the temporal frequency could be variable.

4 EXPLANATION COMMON SOURCE INPUT FORMAT (C.S.I.F).

The parameters for the C.S.I.F. are:

	Full CSIF	1/4 CSIF
Number of active lines		
Luminance (Y)	288	144
Chrominance (U,V)	144	72
Number of active pixels per line		
Luminance (Y)	360	180
Chrominance (U,V)	180	90

Table 1 : Source format (full CSIF and 1/4 CSIF)

The number of coded pels per line is reduced, because 360 devided by 16 does not yield in an integer value. The obtained format is called significant pel area (SPA).

4.1 Definition Of The Significant Pel Area.

```
             CSIF                                         1/4 CSIF

                360                                          180
 -------------------------------------   ^     ----------------------  ^
 |      |                    |      |    |     |   |            |   |  |
 |      |                    |      |    |     | 2 |    176     | 2 | 144
 |      |                    |      |    |     |   |            |   |  |
 |  4   |        352         |  4   |   288    |   |            |   |  |
 |      |                    |      |    |     ----------------------  V
 |      |                    |      |    |
 |      |                    |      |    |
 -------------------------------------   V
```

```
            180
     ----------------   ^           ---------------   ^
     |  |        |   |  |           | |          | |  |
     | 2|  176   | 2 | 144          |1|    88    |1|  72
     |  |        |   |  |           | |          | |  |
     |  |        |   |  |           ---------------   V
     ----------------   V
```

$$U = B - Y = C_b$$

```
            180
     ----------------   ^           ---------------   ^
     |  |        |   |  |           | |          | |  |
     | 2|  176   | 2 | 144          |1|    88    |1|  72
     |  |        |   |  |           | |          | |  |
     |  |        |   |  |           ---------------   V
     ----------------   V
```

$$V = R - Y = C_r$$

Figure 1 : Definition significant pel area

The number of pixels of the reduced picture formats become:

352 x 288	= 101.376	pixels (Y)	Full CSIF
176 x 144 x 2	= 50.688	pixels (U,V)	
total	= 152.064	pixels/frame (Y,U,V)	

176 x 144	= 25.344	pixels (Y)	1/4 CSIF
88 x 72 x 2	= 12.672	pixels (U,V)	
total	= 38.016	pixels/frame (Y,U,V)	

In table 2 the influence of the frame rate on the number of pixels per second is given. This figure includes the number of pixels for the chrominance as well.

Frame Rate 30 Hz	Frame Rate 25 Hz	skip	number active pixels CSIF	Mbit/s	number active pixels 1/4 CSIF	Mbit/s
15 Hz		1 : 2	2.280.960	18.3	570.240	4.6
	12.5Hz	1 : 2	1.900.800	15.2	475.200	3.8
10Hz		1 : 3	1.520.640	12.2	380.160	3.1
	8.33Hz	1 : 3	1.267.200	10.1	316.673	2.5
7.5Hz		1 : 4	1.140.480	9.1	285.120	2.3

Table 2 Bitrate versus frame rate

The first column of table 2 gives the frame rate and the third column depicts the number of omited frames in the coding process for full CSIF and 1/4 CSIF the different values are tabulated. Applying the number of active pixels in one frame the total number of pixels per second are given in the next column with the corresponding bitrates.

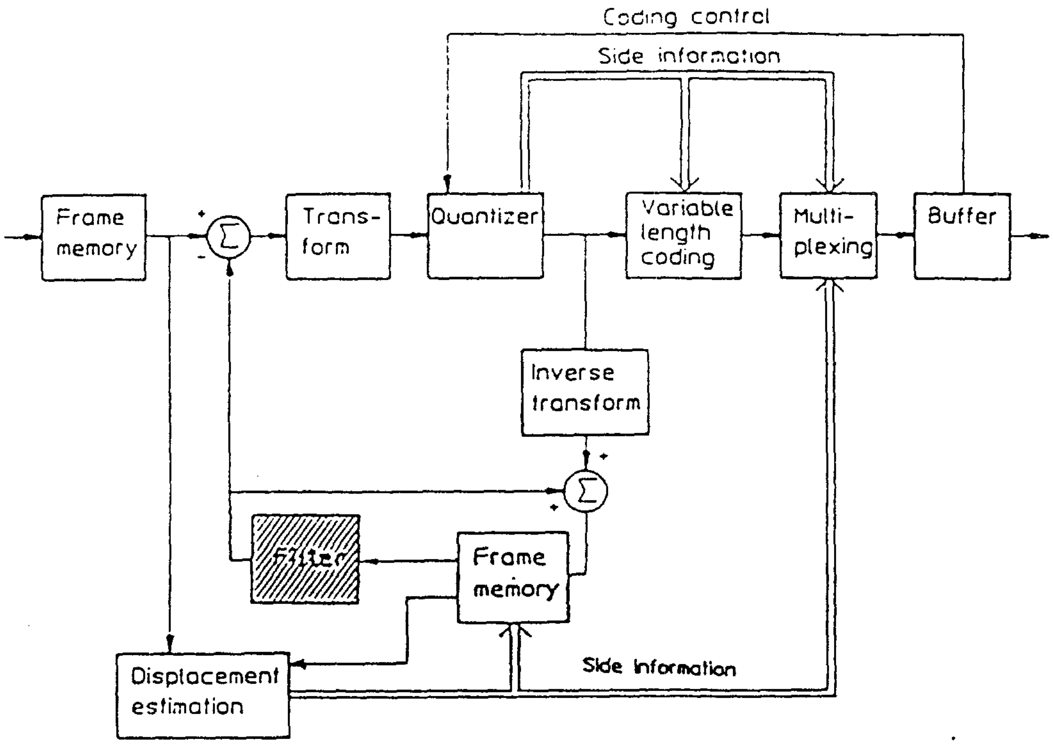


Figure 2 Hybrid transform/DPCM encoder

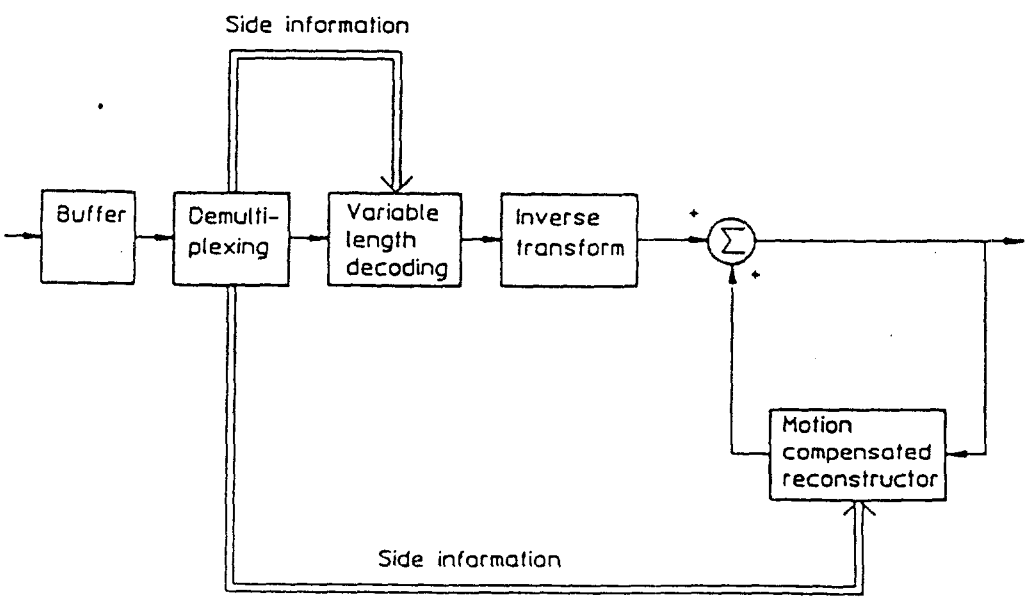


Figure 2 b Hybrid transform/DPCM decoder

5 BASICS OF REFERENCE MODEL 5

5.1 Introduction

The used coding configuration is known as a hybrid DPCM/transform coder. Hybrid denotes a technique which involves more than one redundancy reduction technique, in this case interframe methods where the calculations are performed in pixel and transform domain. This coding procedure requires two transforms, i.e. a forward transform and an inverse transform, which are both located in the coding loop. Due to the usage of a block transform the incoming image is partitioned in non-overlapping blocks of N x N pixels . At the moment the blocksize of the transform is set to N=8).
A simple differential pulse coding modulation loop (DPCM) can be identified as the generic structure of the configuration. This DPCM-loop works in the temporal dimension i.e. interframe. For this purpose a frame memory is included in the loop containing the previously reconstructed image or frame. The generic structure of the reference model depicted in figure 2 is based on:

1. Macro blocks

2. Discrete Cosine Transform (DCT)

3. Variable length coding applying a semi-uniform quantizer

4. A zig-zag scanning of quantized coefficients

5. Displacement estimation

6. Buffer control

Figure 2 Hybrid transform/DPCM encoder.

5.2 Macro Block Approach

A macro block (MB) consists of a 16 x 16 luminance block and the two corresponding 8 x 8 U and V chrominance blocks. The luminance block is divided into four 8 x 8 blocks, i.e. a MB consists of six 8 x 8 blocks.
The construction is depicted in figure 3.

```
      16 * 16                 16 * 8
  ---------------        ---------------
  |      |      |        |      |      |
  | Y00  | Y01  |        |  U   |  V   |
  |      |      |        |      |      |
  ---------------        ---------------
  |      |      |
  | Y10  | Y11  |
  |      |      |
  ---------------
```

Figure 3 : Construction of a Macro Block (MB)

NOTE : A 16 x 16 Luminance block and the two corresponding 8 x 8 U and V chrominance blocks have the same physical size. In table 3 the number of macro blocks and the number of group of blocks are shown:

Format	number of GOB	number of MB in a GOB	total number of MB
CSIF	18	22	396
1/4 CSIF	9	11	99

Table 3 Relation number of Macro blocks and picture format

Y00	Y01	Y10	Y11	U	V

Stucture Macro Block

At a frame rate of 10Hz and a bitrate of 64 kbit/s for each frame one could use 6400 bits (disregarding all bits which should be used). Taking into account a number of overhead bits results in mean bitrate of 15 bits/MB. (396 MB x 15 = 5940 bit)

5.3 Discrete Cosine Transform

The block-differences are transformed with the Discrete Cosine Transform (DCT).
The 2-D DCT is defined as :

$$F(u,v) = \frac{2}{\sqrt{64}} C(u)\ C(v) \sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) \quad \cos\left[\frac{\pi u(2x+1)}{16}\right] \cos\left[\frac{\pi v(2y+1)}{16}\right]$$

with $u = 0,1,2 \ldots\ldots\ldots 7$
$v = 0,1,2 \ldots\ldots\ldots 7$

$$f(x,y) = \frac{2}{\sqrt{64}} \sum_{u=0}^{7} \sum_{v=0}^{7} C(u)\ C(v)\ F(u,v) \quad \cos\left[\frac{\pi u(2x+1)}{16}\right] \cos\left[\frac{\pi v(2y+1)}{16}\right]$$

with $x = 0,1,2 \ldots\ldots\ldots 7$
$y = 0,1,2 \ldots\ldots\ldots 7$

where x,y = spatial coordinates in the pixel domain
u,v = spatial coordinates in the transform domain

$C(u),C(v) = 1/\sqrt{2}$ for $u,v = 0$
1 otherwise

The Luminance blocks and the chrominance blocks are transformed with a blocksize of 8 pixels. It is advisable to exchange the software programs for the DCT and inverse DCT to assure that the different simulations are performed with equal programs.

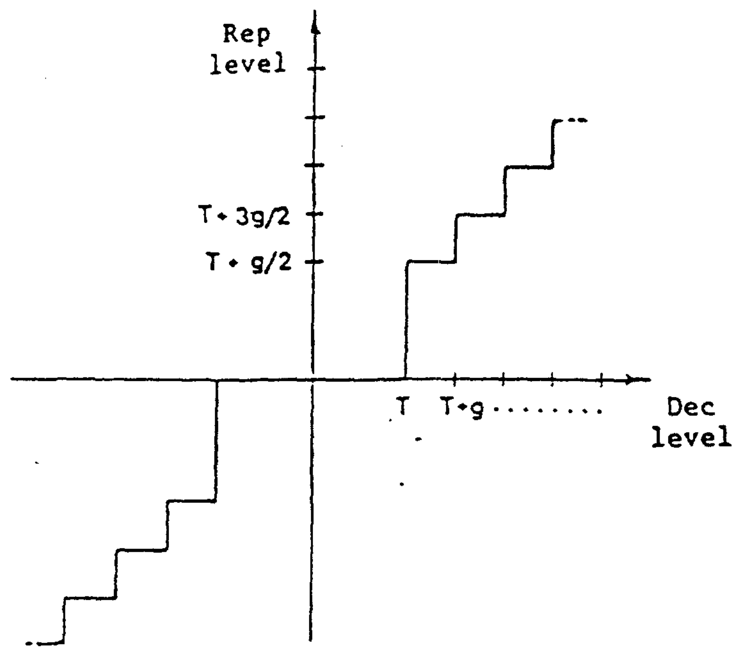


Figure 4 . Characteristic of the quantizer

5.4 Quantization Strategy

The result after the transformation is quantized with an uniform quantizer. The uniform quantizer is defined by a step g and controlled by the buffer fullnes. For RM5 the quantizer threshold has a value T = g.

$$q_{dec}(n) = T + (n-1) g \quad , n \geq 1$$

Taking into account the negative values the expression becomes:

$$q_{dec}(n) = \frac{n}{|n|} \{ T + (|n|-1) g \} \quad , |n| > 1$$

$$q_{rep}(n) = \frac{q_{dec}(n) + q_{dec}(n+1)}{2}$$

with q_{dec} the decision level

q_{rep} the representation level

g the quantizer stepsize

T threshold

In the case of e.g. a transform coefficient F(q(u,v),t)

$$1.0\,g \leq F(q(u,v),t) < 2.0g$$

is quantized to the value of 1.5 g. The characteristic of the quantizer is depicted in figure 4.

Figure 4 . Characteristic of the quantizer

The dynamic range of the coefficients in the case of a blocksize of 8 x 8 is [-2048,2047]. The same quantizer is used both for luminance and chrominance coding.

The quantization stepsize g is transmitted for each group of block. For 64 kbit/s the quantizer levels are scaled by a factor 2 i.e. the maximum stepsize g becomes 64. ($g=4,6,8,..,64$) i.e. 31 different values.

5.5 Coding Of Coefficients

5.5.1 Scanning Technique -

In order to increase the efficiency of capturing the non zero components a zig-zag scanning class has been adopted:

ZIG - ZAG SCANNING :

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

The transmission of the coefficients must stop when the last non zero has been reached.

5.5.2 Coding Of The Scanned Coefficients With A Two Dimensional VLC.

To increase coding efficiency a two dimensional variable length code has been adopted. This means that "events" are coded, where an "event" is defined as :

event : a combination of a magnitude (non-zero quantization index) and a RUN (Number of zero indexes preceeding the current non-zero index)

Coefficients unequal to zero defining the end of the run-length are considered as composite rather than seperate statistical event.

The run-length and the magnitude of composite events define the entries of the 2-D-run-length table which contains the code words for the composite events. Events are coded with Huffman's algorithm. However, events with low probabilities are coded using fixed length codes. These codes consists of the following three parts.

1. Escape (6 bits) for indicating the use of fixed length codes.

2. Run (6 bits)

3. Level (8 bits. Here, it is assumed that the number of quantization indexes is less than 128).

NOTE 1: Note that clipping must be introduced for the quantized coefficients F : $-128\,g \leq F < 128\,g$. The maximum range for the non-zero coefficients is now $+128g$ and $-128g$. The length of the EOB word is two bits.

NOTE 2: $0 \leq \text{run} < 64$ (for blocksize 8) After the last non-zero coefficient an End-Of-Block (EOB) marker is sent indicating that all other coefficients are zero.
An example of the two dimensional VLC is given in figure 5 and the table is annexed.

EVENT = (RUN,LEVEL)

Example: (0,3) (1,2) (7,1) EOB

3	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 5 : Example 2-D VLC

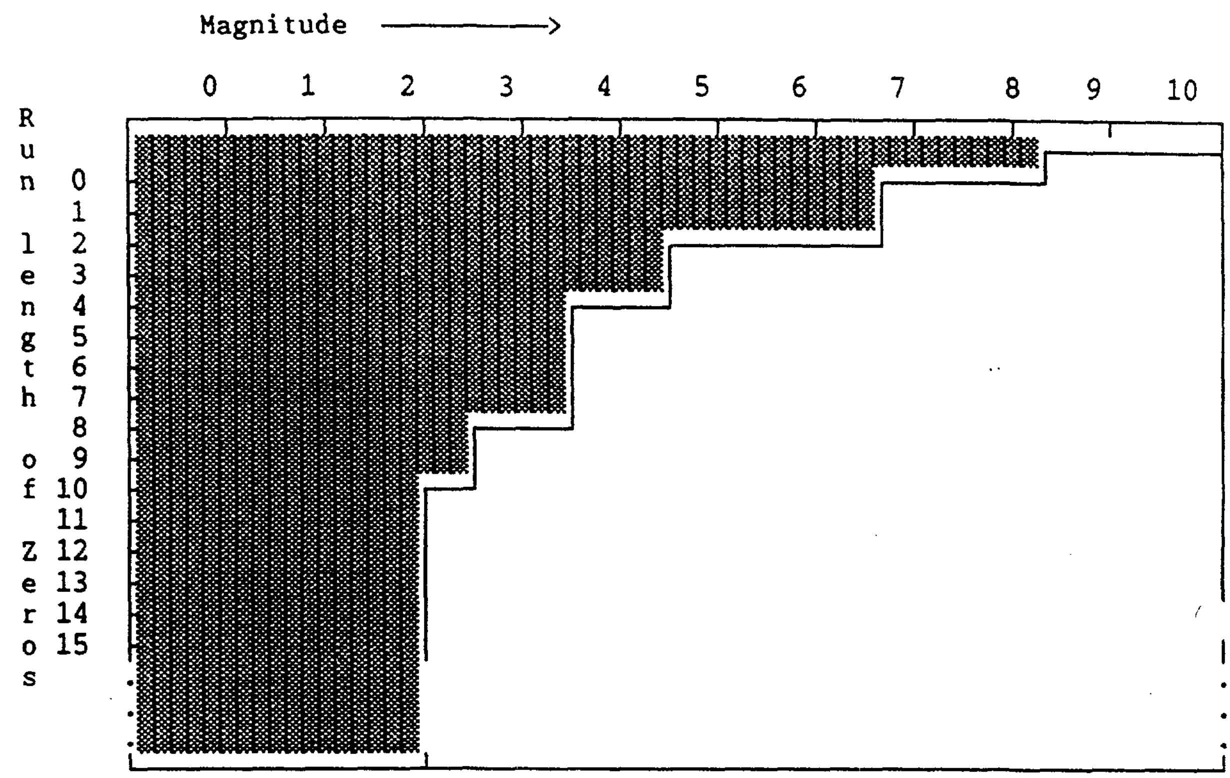
That means:

* (0,3) The DC component which has the value +3

* (1,2) is next non-zero component according to the zig-zag scanning the number of zeroes is 1.

* The next component is 1 preceeded by 7 zeroes, result (7,1)

* EOB is an End of Block marker which indicates that there are no more non zero components.



Example 2-D VLC table

5.6 Coding Strategy And Block Type Discrimination

In RM5 four different block types can be distinguished :

1. no MC not coded
2. no MC coded
3. MC not coded (only motion vectors are transmitted)
4. MC coded

The order in which the block type is determined is depicted in figure 6.

```
                                   ---- Not Coded
                                   |
                    ------ No Mc   |
                    |              |
                    |              ---- Coded
   input block      |
                    |              ---- Not Coded
                    |              |
                    ------ MC      |
                                   |
                                   ----  Coded
```

Figure 6 Decision Tree RM5

If after quantization, all the quantized components of a block are zero, the block is declared to be fixed.
If all six blocks in a MB are fixed, the MB is declared to be fixed. In all other cases the MB is declared to be coded.
When one or more blocks in a coded MB are fixed, only an EOB is sent for these blocks :

e.g. Y00, Y10, Y11, U and V are fixed

Y01 is not fixed

```
Y00        Y01             Y10  Y11   U    V
EOB, EVENT-----EVENT EOB, EOB, EOB, EOB, EOB
```

NOTE : The Data Per Block (DPB) is only transmitted if the macro block is not fixed.

All non-fixed MB are addressed with relative addressing, similar to the relative addressing used in RM4 (table 4) . The other block-types are coded according to the VLC in table 5.

NOTE this example should be made more clear

number of fixed MB's	code	number of fixed MB's	code
0	1	13	0000 0001 0
1	010	14	0000 0001 1
2	011	15	0000 0001 0000
3	0010	16	0000 0001 0001
4	0011	17	0000 0001 0010
5	0001 0	18	0000 0001 0011
6	0001 1	19	0000 0001 0100
7	0000 10	20	0000 0001 0101
8	0000 11	21	0000 0001 0110
9	0000 010	22	0000 0001 0111
10	0000 011		
11	0000 0010		
12	0000 0011		

table 4 : Adopted VLC for relative addressing of non-fixed MB's

Macro Block type		code with rel. addr.	code without rel. addr.
1.	no MC not coded (fixed)	-	1
2.	MC coded	1	01
3.	MC not coded	01	001
4.	no MC coded	00	000

table 5 : Adopted VLC for macro block types

Example 1 :

3 1 1 1 1 1 1 4 2 2 2 2 3 1 1 1 1 1 1 1 1 1

code with rel. addr.	code without rel. addr.	
1	-	Number of fixed MB's : 0
01	001	Block type : MC not coded
00011	111111	Number of fixed MB's : 6
01	000	Block type : no MC coded
1	-	Number of fixed MB's : 0
1	01	Block type : MC coded
1	-	Number of fixed MB's : 0
1	01	Block type : MC coded
1	-	Number of fixed MB's : 0
1	01	Block type : MC coded
1	-	Number of fixed MB's : 0
1	01	Block type : MC coded
1	-	Number of fixed MB's : 0
1	001	Block type : MC not coded
0000 010	111111111	Number of fixed MB's : 9
--------- +	--------- +	
27 bits	32 bits	

Example 2 :

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

code with rel. addr. : 0000 0001 0111 (all fixed MB's)
code without rel. addr. : 1111 1111 1111 1111 1111 11
As depicted in the examples one could notice the small gain which is negligible.

5.7 Filter In The Loop

The introduction of a low pass filter after motion compensation (MC) could have the following advantages:

i. A reduction of high frequency artifacts introduced by MC.

ii. A reduction of quantization noise in the feedback loop.

iii. Adaption according to motion vector without overhead.

The filter could be controlled with:

1. Displacement vector
2. Prediction error

A filter with impulse response as depicted in (1) is applied on a block of 8 x 8 pixels. The filter is applied both on luminance and the chrominance.

$$h(k,l) = \begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{vmatrix} \quad 1/16 \qquad (1)$$

5.7.1 Filtering Inside The Block Boundaries. -

At the block boundaries the filter coefficients need to be adjusted in the case of adopting filtering inside the block.

i. $\begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{vmatrix}$ for pixels inside the block edges

ii. $\begin{vmatrix} 3 & 1 \\ 6 & 2 \\ 3 & 1 \end{vmatrix}$ for pixels on the block edges

iii. $\begin{vmatrix} 9 & 3 \\ 3 & 1 \end{vmatrix}$ for pixels on the block corner positions

In reference model 5 the filter is controlled with the motion vector i.e. if the motion vector is non zero the filter is on. Luminance blocks and chrominance blocks are filtered.

5.8 Buffer

5.8.1 Buffer Control

For RM5 the stepsize varies from 4 to 64 with step 2. The dependence on the buffer fullness is depicted in table 6.

buffer content [bit]	stepsize quantizer
< 400	4
< 600	6
< 800	8
.	.
.	.
.	.
< 6000	60
< 6200	62
> 6200	64

or : step = 2 * INT (buf_cont / 200) + 2

table 6 : quantizer stepsize as a function of the buffer fullness

5.8.2 Buffer Overflow

A buffer size of 6.4 kbit is intended. After each MB the buffercontent can be calculated (mean 15 bit/MB). When the buffer fullness exceeds 6.4 kbits, the coefficients and the motion vector are set to zero in the next macro block (however resulting in a small buffer overflow).

N.B. : When the frame rate is not equal 10 Hz or the bitrate is not equal to 64 kbit/s, the mean number of bits per MB can be computed then according to (3).

bitrate : p * 64 kbit/s

frame rate :	30 Hz	-->	5p bits/MB
	15 Hz	-->	10p bits/MB
	10 Hz	-->	15p bits/MB
	7.5 Hz	-->	20p bits/MB

5.9 Motion Estimation

The prediction error, i.e. the block difference, can be minimized with displacement compensation usually called motion compensation.

The method adopted in RM5 is at the moment the same as used in RM4 the 3-step. The method can be found in appendix A.

The motion estimation is applied on luminance as well as chrominance. Following the lower resolution the chrominance first the vector has to be divided by two and truncated to integer value.

example :

for luminance	-->	for chrominance
(3, 2)		(1, 1)
(-5,-6)		(-2,-3)

5.10 MC/No MC Decision

For the moment we use the characteristic as defined in RM4. The evaluation function for displacement estimation is a sum of absolute differences concerning to all of the pels in a block. The characteristic whether to suppress the displacement vector is depicted in figure 7. The characteristic is determined experimentally.

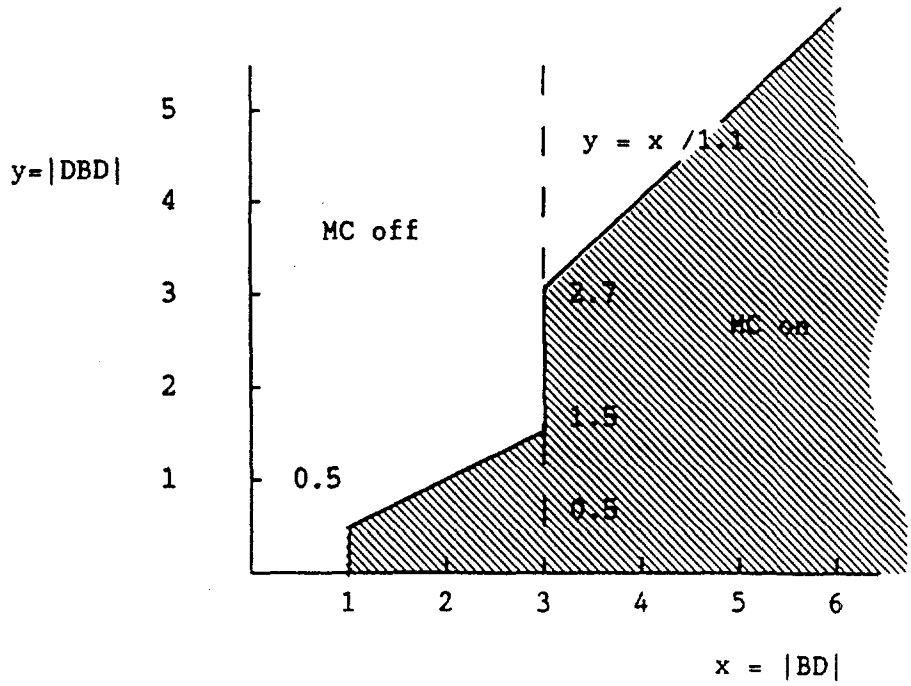


Figure 7. Characteristic MC/ No MC

NOTE: MC off includes the solid line. This characteristic resolves partly the sticking noise in the uncovered background (#107).

5.11 First Picture And Scene Change

In the case of a first picture, the intra-frame mode is used. To overcome the problem of a buffer overflow, the second picture is skipped. This means that for the first picture the amount of available bits is twice compared to the average amount of bits per picture.

original sequence	1	2	3	4	5	6	7	8	(10 Hz)
coded sequence	1	1	3	4	5	6	7	8	(10 Hz)

Table 7 : Temporal reference

The rounding value of the luminance as well as chrominance blocks of a macro block are calculated and transmitted using 8 bits (which mean at least 24 bits/MB for the first picture). Each next incoming block is subtracted from the calculated average and interframe coded with an initial stepsize of 16.

6 PRESENTATION OF RESULTS

INSTITUTE

STATISTICS RM5
SEQUENCE :
MODIFICATION :

DATE :

FRAME RATE:

ITEM			15th Picture	Mean whole seq.
1. RMS for luminance				
2. SNR for luminance for chrominance (u) for chrominance (v)				
3. Mean value of step size				
4. Mean value of the number of non-zero coefficients				
5. Mean value of the number of zeroes before the last NZ-coefficient				
6. Block type of MACRO	FIXED CODED MC FIXED MC CODED			
7. Block type of Y	FIXED CODED MC FIXED MC CODED			
8. Block type of UV	FIXED CODED			
9. Number of bits	Macro attributes			
	End of block			
	Motion vectors			
	Coefficients	Y U V Total		
	Total			

CHECK SUM

ad 2 : The SNR is calculated for luminance as well as chrominance.

ad 9 : The number of bits for the MB attributes also includes the bits for relative addressing.

ad 9 : The number of bits for 'End of block' must be equal to (excluding rounding errors) :

12 (6 blocks) x (#macro coded MC + macro coded)

ad 9 : The number of bits for 'Motion vectors' must be equal to (excluding rounding errors) :

8 x (#macro coded MC + #macro fixed MC)

INDEX

APPENDIX A

3-STEP-ALGORITHM

3-STEP-algorithm.

Assuming a maximum displacement of 7 pixels the step algorithm iterates in three steps to the resulting minimum error.

Step 1 : The actual block B is matched with 9 blocks in the previous window SW at the following positions :

(-4-4)	(0-4)	(4-4)
(-4 0)	(0 0)	(4 0)
(-4 4)	(0 4)	(4 4)

The position of the non-shifted prediction is (0 0). The order of the search has been defined as:

2	3	4
5	1	6
7	8	9

Step 2 : For the second step a new search pattern is used. The best match of step 1 is the centre of this pattern :

(-2-2)	(0-2)	(2-2)
(-2 0)	(0 0)	(2 0)
(-2 2)	(0 2)	(2 2)

Step 3 : The position of the best match in step 2 is the central position of the third and final search pattern :

(-1-1)(0-1)(1-1)
(-1 0)(0 0)(1 0)
(-1 1)(0 1)(1 1)

The best match of step 3 is the resulting minimum match error.

It should be noticed that this algorithm only fits in a limited number of search areas, due to the step-strategy. In this context only five search areas are distinguished (see table 3).

	search window	no matches
0-step-algorithm	8 x 8 pixels	1
1-step-algorithm	10 x 10 pixels	9
2-step-algorithm	14 x 14 pixels	17
3-step-algorithm	22 x 22 pixels	25
4-step-algorithm	38 x 38 pixels	33

table 3 Number of matches for the STEP-algorithm for a limited number of search areas.

An example of the search process for the 3-step algorithm :

```
1       1       1

            2   2   2

1       1   2   1   2
                  3 3 3
            2   2 3 2 3
                  3 3 3
1       1       1
```

APPENDIX B

FIGURES

APPENDIX C

TABLES

APPENDIX D

ADOPTED VARIABLE LENGTH CODES

Word Length of VLC for Two-dimensional Coding

LEVEL (absolute value)

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	. .128
	0	3	5	6	8	9	9	11	13	13	13	13	14	14	14	14	20	20	20	20	20	
	1	4	7	9	11	13	14	14	20	20	20	20	20	20	20	20	20	20	20	20	20	
	2.	5	8	11	13	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	3	6	9	13	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	4	6	11	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	5	7	11	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	6	7	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	7	7	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
R	8	8	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
U	9	8	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
N	10	9	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	11	9	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	12	9	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	13	9	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	14	11	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	15	11	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	16	11	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	17	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	18	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	19	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	20	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	21	13	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	22	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	23	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	24	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	25	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	26	14	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	27	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	28	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
	.																					
	.																					
	63																					

** Word lengths of all other EVENTs (combination of RUN and LEVEL) are 20: ESCAPE CODE(6bits)+RUN(6BITS)+LEVEL(8bits).

** Word length of EOB is 2.

APPENDIX A'

ADOPTED VARIABLE LENGTH CODES

Code	Length	TCOEFF (a)	TCOEFF (b)	BA
1	1	Vo	V1	0
001	3	EOB	not used	1
010	3	V1	V-1	2
011	3	V-1	V2	3
00010	5	V2	V-2	4
00011	5	V-2	V3	5
000010	6	V3	V-3	6
000011	6	V-3	V4	7
0000010	7	V4	V-4	8
0000011	7	V-4	V5	9
00000010	8	V5	V-5	10
00000011	8	V-5	V6	11
000000010	9	V6	V-6	12
000000011	9	V-6	V7	13
0000000011111110	16	V7	V-7	14
0000000011111111	16	V-7	V8	15
0000000011111100	16	V8	V-8	16
0000000011111101	16	V-8	V9	17
0000000011111010	16	V9	V-9	18
0000000011111011	16	V-9	V10	19
0000000011111000	16	V10	V-10	20
0000000011111001	16	V-10	V11	21
....		..	..	..
....		..	..	..
0000000010001010	16	V65	V-65	130
0000000010001011	16	V-65	V66	131
0000000010001000	16	V66	V-66	
0000000010001001	16	V-66	V67	
0000000010000110	16	V67	V-67	
0000000010000111	16	V-67	V68	
0000000010000100	16	V68	V-68	
0000000010000101	16	V-68	V69	
0000000010000010	16	V69	V-69	
0000000010000011	16	V-69	V0	

Notes:

1. TCOEFF. Column (a) is used for the quantiser index of all coefficients except for the one immediately preceding the EOB.

Column (b) is used for the quantiser index of the coefficient immediately preceding the EOB.

2. BA is the block address information

No	RUN	LEVEL	CODE LENGTH	CODE	CODE STRUCTURE
95	22	-1	14	00000000111111	
6	22	1	14	.	
7	23	-1	14	.	
8	23	1	14	.	
9	24	-1	14	.	
100	24	1	14	.	
1	25	-1	14	.	
2	25	1	14	.	
3	26	-1	14	.	
4	26	1	14	.	
5	0	-12	14	.	
6	0	12	14	.	
7	0	-13	14	.	VLC (9bits)
8	0	13	14	.	
9	0	-14	14	.	+
110	0	14	14	.	
1	0	-15	14	.	FLC (5bits)
2	0	15	14	.	
3	1	-6	14	.	
4	1	6	14	.	
5	1	-7	14	.	
6	1	7	14	.	
7	2	-5	14	.	
8	2	5	14	.	
9	3	-4	14	.	
120	3	4	14	.	
1	5	-3	14	.	
2	5	3	14	.	
3	9	-2	14	.	
4	9	2	14	.	
5	10	-2	14	.	
6	10	2	14	00000000100000	
127	EOB WORD		2	01	VLC
128	ESCAPE CODE		6	001000	VLC

** Other EVENTs (combination of RUN and LEVEL) are coded to:
ESCAPE CODE(6bits)+RUN(6bits)+LEVEL(8bits)

No	RUN	LEVEL	CODE LENGTH	CODE	CODE STRUCTURE
47	4	-2	11	00000011111	
8	4	2	11	.	
9	14	-1	11	.	
50	14	1	11	.	
1	15	-1	11	.	
2	15	1	11	.	VLC (7bits)
3	1	-4	11	.	
4	1	4	11	.	+
5	2	-3	11	.	
6	2	3	11	.	FLC (4bits)
7	0	-7	11	.	
8	0	7	11	.	
9	5	-2	11	.	
60	5	2	11	.	
1	16	-1	11	.	
2	16	1	11	00000010000	
3	17	-1	13	0000000111111	
4	17	1	13	.	
5	6	-2	13	.	
6	6	2	13	.	
7	0	-8	13	.	
8	0	8	13	.	
9	3	-3	13	.	
70	3	3	13	.	
1	1	-5	13	.	
2	1	5	13	.	
3	18	-1	13	.	VLC (8bits)
4	18	1	13	.	
5	19	-1	13	.	+
6	19	1	13	.	
7	0	-9	13	.	FLC (5bits)
8	0	9	13	.	
9	20	-1	13	.	
80	20	1	13	.	
1	21	-1	13	.	
2	21	1	13	.	
3	7	-2	13	.	
4	7	2	13	.	
5	2	-4	13	.	
6	2	4	13	.	
7	0	-10	13	.	
8	0	10	13	.	
9	4	-3	13	.	
90	4	3	13	.	
1	8	-2	13	.	
2	8	2	13	.	
3	0	-11	13	.	
4	0	11	13	0000000100000	

Code Set for Two-dimensional Coding of Coefficient
Quantization Index (RM4)

No	RUN	LEVEL	CODE LENGTH	CODE	CODE STRUCTURE
1	0	-1	3	111	VLC
2	0	1	3	110	
3	1	-1	4	1011	VLC
4	1	1	4	1010	
5	2	-1	5	10011	
6	2	1	5	10010	VLC
7	0	-2	5	10001	
8	0	2	5	10000	
9	3	-1	6	001111	
10	3	1	6	001110	
11	4	-1	6	001101	VLC
2	4	1	6	001100	
3	0	-3	6	001011	
4	0	3	6	001010	
5	5	-1	7	0001111	
6	5	1	7	.	VLC (4bits)
7	1	-2	7	.	
8	1	2	7	.	+
9	6	-1	7	.	
20	6	1	7	.	FLC (3bits)
1	7	-1	7	.	
2	7	1	7	0001000	
3	8	-1	8	00001111	
4	8	1	8	.	VLC (5bits)
5	0	-4	8	.	
6	0	4	8	.	+
7	9	-1	8	.	
8	9	1	8	.	FLC (3bits)
9	2	-2	8	.	
30	2	2	8	00001000	
31	10	-1	9	001001111	VLC (6bits)
2	10	1	9	.	
3	0	-5	9	.	+
4	0	5	9	.	
5	1	-3	9	.	
6	1	3	9	.	FLC (3bits)
7	3	-2	9	.	
8	3	2	9	001001000	
9	11	-1	9	000001111	
40	11	1	9	.	VLC (6bits)
1	12	-1	9	.	
2	12	1	9	.	+
3	0	-6	9	.	
4	0	6	9	.	FLC (3bits)
5	13	-1	9	.	
6	13	1	9	0000010000	