PROC of the ICASSP-93 FA 12 Minneapolis, Minnesota, USA, April 1993

OBJECT-ORIENTED ANALYSIS-SYNTHESIS CODING BASED ON SOURCE MODELS OF MOVING 2D- AND 3D-OBJECTS

Hans Georg Musmann

Universität Hannover
Germany

ABSTRACT

For encoding moving images at very low bit rates object-oriented analysis-synthesis coding using source models of moving 2D- and 3D-objects has been investigated. According to this coding concept each moving object of an image is described and encoded by three parameter sets defining the motion, the shape and the surface colour of the moving object.

The parameters to be coded are dependent on the source model being applied. Thus, the coding efficiency of object-oriented analysis-synthesis coding can be evaluated by comparing the encoded parameter bit rates for a fixed picture quality. The results obtained with source models of moving 2D- and 3D-objects are compared to those of a standard H.261 coder at a bit rate of 64 kbit/s.

The source models of flexible 2D- and rigid 3D-objects are more efficient than source models which are based on rigid 2D-objects or on moving blocks as in the case of a H.261 coder.

1. INTRODUCTION

In order to encode moving video signals at 64 kbit/s transmission rate the standardized hybrid coding technique H.261 subdivides each image of a sequence into blocks of 16 x 16 picture elements (pels) and encodes each block by motion compensated predictive and transform coding [1]. Thus this coding technique is based on the source model of moving square blocks which can lead to visible distortions known as blocking and mosquito effects. To avoid these image distortions object-oriented analysis-synthesis coding [2] has been proposed. This coding technique is based on the source model of moving objects instead of moving square blocks. Each object is described and encoded by three parameter sets defining the motion, shape and surface colour of the object.

The parameter sets to be coded vary with the kind of object model. Three kinds of object models have been invetigated, rigid 2D-objects [3], flexible 2D-objects [4] and rigid 3D-objects [5]. In order to evaluate the coding efficiency of object-oriented analysis-synthesis coding the encoded parameter bit rates of the cited source models are compared at a fixed picture quality measured by the signal-to-noise ratio.

After a short explanation of the basic components of an object-oriented analysis-synthesis coder in section 2 the coding of the parameter sets is described in section 3. Finally the influence of the various source models on the coding efficiency is discussed in section 4.

2. GENERAL STRUCTURE OF AN OBJECT-ORIENTED ANALYSIS-SYNTHESIS CODER

The block diagram of an object oriented analysis-synthesis coder is shown in Fig.1. Instead of a frame memory as in block-oriented coding techniques object-oriented coding requires a memory to store the parameter sets $A = \{A_i\}$, $M = \{M_i\}$, $S = \{S_i\}$ defining the motion A_i, shape M_i and colour S_i of each moving object i. The memory for object parameters of the coder and decoder contains the same parameter informations and allows the coder and decoder to reconstruct a transmitted image by image synthesis. The reconstructed image I'_k is displayed at the decoder and used for image analysis of the next input image I_{k+1} at the coder.

The task of the image analysis block in Fig.1 is to estimate the parameter sets A_i, M_i. S_i for each object i of the next input image I_{k+1} by use of the reconstructed image I'_k. At the output of the image analyis block the parameter sets A_i, M_i, S_i are available in PCM representation.

Image analysis distinguishes between two types of objects. Model Compliance objects (MC-objects) can be reconstructed by transmitting only the motion A_i and shape M_i parameter sets and using the colour S_i parameter set stored in the memory. Model failure objects (MF-objects) are image areas where the motion description by the source model fails. MF-objects are encoded by the shape M_i and colour S_i parameter sets, see Table 1.

	MC-Object	MF-Object
Motion Ai	A_i	
Shape M_i	M_i	M_i
Colour S_i		S_i

Table 1. Parameter sets transmitted for MC- and MF-objects

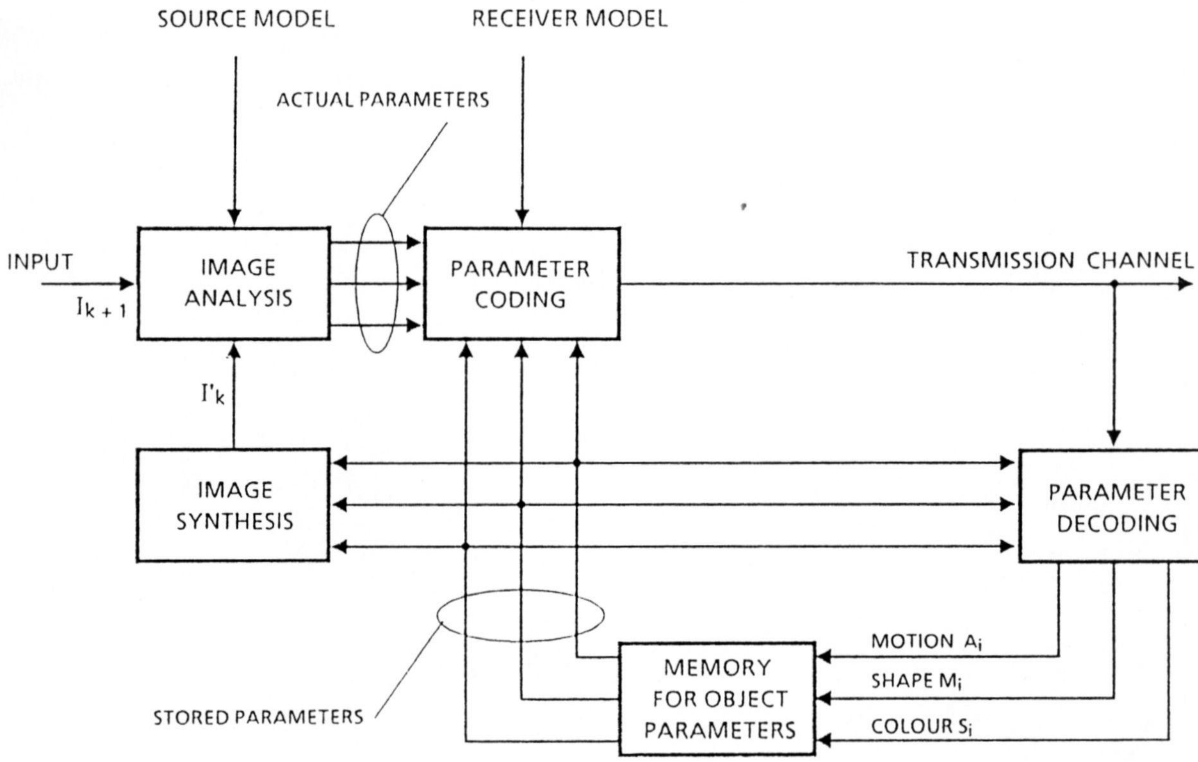

Fig. 1. Block diagram of an object–oriented analysis–synthesis coder

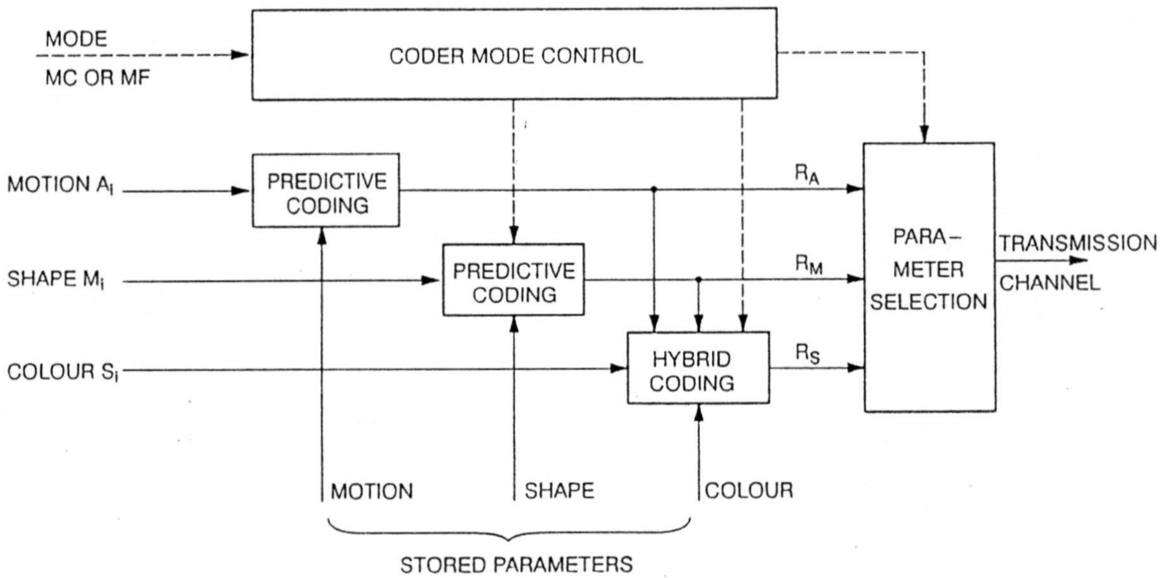

Fig. 2. Block diagram of the parameter coder

The actual parameter sets A_i, M_i, S_i of image I_{k+1} are encoded in the block parameter coding. For efficient coding known predictive and transform coding techniques are applied which make use of the stored parameter sets [4], [5], see Fig.2. Coder mode control decides with help of the MC/MF information which parameter sets of each object will be transmitted.

3. PARAMETER SETS AND PARAMETER CODING

The parameter sets to be coded depend on the source model as indicated in Table 2. In order to achieve an efficient coding of the parameter sets individual coding techniques are applied for each parameter set.

Motion parameter sets of MC–objects are predictively encoded using known DPCM techniques [3], [5]. In the case of rigid 2D–objects threedimensional motion is assumed. Therefore 8 mapping parameters $a_1,..a_8$ describing the motion have to be encoded for each MC–object [2]. In the case of flexible 2D–objects twodimensional translatory motion of each block of an MC–object is assumed [4]. The motion of 3D–objects is described by 3 translatory motion a_1, a_2, a_3 and 3 rotatory motion parameters a_4, a_5, a_6. The shape parameters of 2D–objects describe the silhouette of a MC–object by polygon and spline approximation. The vertices of this representation are predictively encoded using the actual motion and the stored shape information [3], [6]. The shape of 3D–objects is obtained only from silhouette and motion information [8]. Therefore only silhouette information has to be encoded. The accuracy of the shape approximation has to be adapted to the spatial image resolution and the camera aperture. The maximum allowable deviation of the shape approximation from the true shape contour is about 1.5 pel in order to avoid visible distortion at object boundaries.

There are less restrictions for the representation of the boundaries of MF–objects. The maximum deviation should be about 2.1 pel otherwise the bit rate for encoding the colour information of MF–objects may increase too much [4].

The colour information of MF–objects is coded by a block–oriented hybrid scheme where for each block of 8 x 8 pel a DPCM techique is used instead of a DCT whenever it allows a more efficient coding [7].

4. CODING EFFICIENCY

In order to compare the coding efficiency of object–oriented analysis–synthesis coders with different source models the bit rates R_A, R_M and R_S required for encoding the motion, shape and colour parameter sets have to be analysed. Table 3 shows the average bit rates R_A, R_M and R_S per image measured for the CCITT test sequences "Mrs. America" and "Claire" using a spatial resolution of 288 lines with 352 pel per line and 10 Hz frame frequency.

The bit rate R_M required for encoding the motion parameters of rigid objects is less than that for flexible objects, since many displacement vectors $\{a_1, a_2\}$ have to be encoded for only one MC–object.

The bit rate R_M consists of two parts, one part for MC– the other for MF–objects. R_M of rigid 2D–objects is greater than that for flexible 2D–objects since the source model of rigid 2D–objects fails more often and therefore generates more MF–objects. This can also be recognized from the number of pels N of the MF–objects to be coded. The bit rate R_S for encoding the colour parameters $\{S_1,..S_N\}$ results from the number of pels N to be coded multiplied by the bit rate r_s allocated for encoding each pel

$$R_S = N \cdot r_s \quad . \qquad (1)$$

With the choice of r_s the picture quality can be fixed. Thus N is a measure for the transmission rate required for coding the colour parameters and can be used for comparing the coding efficiency of different source models.

Table 3 shows that the two source models of 2D–objects generate almost the same bit rate $R_A + R_M$. Since N is much greater for the source model of rigid 2D–object the source model of flexible 2D–objects provides a higher coding efficiency.

Comparing the model of flexible 2D–objects with that of rigid 3D–objects the bit rate R_A is smaller for rigid 3D–objects since the number of motion parameters is less. The shape information of MC–objects is the same for flexible 2D– and rigid 3D–objects since both are encoded by silhouettes. However, the shape information required for encoding the MF–objects is greater in the case of rigid 3D–objects since the total area N of MF–objects is split into more MF–objects of smaller size. Thus R_M is higher for rigid 3D–objects although N is the same.

To achieve an acceptable picture quality r_s should be greater than 1 bit per pel. Thus the source model of flexible 2D– and rigid 3D–objects allow coding with a total bit rate of 64 kbit/s.

In a block oriented H.261 coder N is much greater in the range of 40000 pels .

5. CONCLUSION

The coding efficiency of object–oriented analysis–synthesis coders based on different source models is evaluated. At a given picture quality the bit rates

$$R = R_A + R_M + R_S \qquad (2)$$

required for encoding the parameter sets for motion R_A, shape R_M and colour R_S are compared. It has been shown that there is only a relatively low degree of freedom for encoding the motion and shape parameters so that $R_A + R_M$ is almost fixed. R_S is linearly dependent on the number N of pels in the areas of model failure at a given picture quality.

The source models of flexible 2D–objects and rigid 3D–objects are almost equivalent and superior to those of rigid 2D–objects and of moving 2D–blocks as in the case of a H.261 coder.

To improve the coding efficiency the source model of flexible 3D–objects has to be introduced [9].

	MC–object		MF–object	
	Motion A_i	Shape M_i	Colour S_i	Shape M_i
Rigid 2D–objects 3D–motion	8 mapping parameters $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8$	vertices of the object silhouette $\{ m_1, m_2 \}$	each pel of a MF–object $\{ s_1, \ldots s_N \}$	vertices of the boundary of the MF–area $\{ m_1, m_2 \}$
Flexible 2D–objects 2D–motion	2 translatory parameters for each block of an object $\{ a_1, a_2 \}$	vertices of the object silhouette $\{ m_1, m_2 \}$	each pel of a MF–object $\{ s_1, \ldots s_N \}$	vertices of the boundary of the MF–area $\{ m_1, m_2 \}$
Rigid 3D–objects 3D–motion	3 translatory and 3 rotary parameters $a_1, a_2, a_3, a_4, a_5, a_6$	vertices of the object silhouette $\{ m_1, m_2 \}$	each pel of a MF–object $\{ s_1, \ldots s_N \}$	vertices of the boundary of the MF–area $\{ m_1, m_2 \}$

Table 2. Parameter sets to be coded per object for various source models

Source model	Motion information R_A	Shape information R_M	Colour information $R_S = N \cdot r_s$
Rigid 2D–objects 3D–motion	600	1 300	$15\,000 \cdot r_s$
Flexible 2D–objects 2D–motion	1 100	900	$4\,000 \cdot r_s$
Rigid 3D–objects 3D–motion	200	1 640	$4\,000 \cdot r_s$

Table 3. Average bit rates R_A, R_M, R_S in bit per image for various source models, N is the totel number of pels of the MF–objects, r_s is the bit rate allocated per pel

REFERENCES

[1] CCITT, "Draft revision of recommendation H.261: video codec for audio visual services at p x 64 kbit/s", Study Group XV, WP/1/Q4, Specialist Group on Coding for Visual Telephony, Cambridge, USA, November 1988

[2] H. G. Musmann, M. Hötter and J. Ostermann, "Object–oriented analysis–synthesis coding of moving images", Signal Processing: Image Communications, Vol.1, No.2, pp. 117–138, October 1989

[3] M. Hötter, "Object–oriented analysis–synthesis coding based on moving two–dimensional objects", Signal Processing: Image Communications, Vol.2, No.4, pp. 409–428, December 1990

[4] M. Hötter, "Optimization and efficiency of an object–oriented analysis–synthesis coder", IEEE Trans. on Circuits and Systems for Video Technology. Accepted for publication

[5] J. Ostermann, "Object–oriented analysis–synthesis coding based on the source model of moving rigid 3D–objects", submitted for publication to Signal Processing: Image Communications

[6] M. Hötter, "Predictive contour coding for an object–oriented analysis–synthesis coder, IEEE International Symposium on Information Theory, San Diego, Californiia, USA, p. 75, January 1990

[7] H. Schiller, M. Hötter, "Colour coding in an object–oriented analysis–synthesis coder", submitted for publication in Signal Processing: Image Communication

[8] J. Ostermann, "Modelling of 3D–moving objects for an analysis–synthesis coder", SPIE/SPSE Symposium on Sensing and Reconstruction of 3D Objects and Scenes '90, B. Girod, Editor, Proc. SPIE 1260, pp.240–250, Santa Clara, California, February 1990

[9] J. Ostermann, "Analysis–synthesis coder based on moving flexible 3D–objects", accepted for publication at Picture Coding Symposium PCS 93, Lausanne, Switzerland, March 1993