Prototype Prediction for Colour Update in Object-Based Analysis-Synthesis Coding

Michael Wollborn

Abstract— **In an object-based analysis-synthesis coder each image of an image sequence is subdivided into arbitrarily shaped moving objects. The objects are described by three parameter sets defining their motion, shape and colour, where the colour parameters denote the luminance and chrominance values of the object surface. In order to increase the efficiency of the coder, a predictive technique for the coding of colour parameters is presented using image pattern prototypes for prediction. The prototypes are extracted from previously coded images and identically stored at the coder and the decoder. They can be used for the prediction of the colour parameters of an object by applying a suitably defined mapping function. The prediction error is coded by an adaptive combination of DCT and Intraframe DPCM. For the reconstruction of the colour parameters at the decoder the address of the image pattern prototype and the parameters of the mapping function have to be transmitted as side information. Prototype Prediction is only applied, when its data rate is lower than that of an extended hybrid scheme. At a transmission data rate of 64 kbit/s Prototype Prediction increases the PSNR by 2 dB in the areas of colour update.**

Index Terms—**Object-based image coding, videophone, colour, parameter coding, prototype prediction.**

I. Introduction

FOR the coding of image sequences at low bit rates between 64 kbit/s and 2 Mbit/s a hybrid coding concept has been standardized by the CCITT where each image of an image sequence is subdivided into square blocks of $N \times N$ picture elements (pels) [3], [4]. Each block is coded separately using 2D motion compensated prediction and transform coding [20]. In order to avoid the typical coding artefacts caused by this coding technique, known as blocking and mosquito effects, the concept of object-based analysis-synthesis coding (OBASC) aiming at bit rates of 64 kbit/s and below has been proposed [16].

A coder based on this concept subdivides each image of an image sequence into arbitrarily shaped moving objects and describes each object by three parameter sets defining its motion, shape and colour, where the colour parameters denote the luminance and chrominance values of the object surface [16]. For those objects where the description of the object motion by the source model is sufficiently exact, only shape and motion parameters have to be transmitted to the receiver. Objects which cannot be described sufficiently exact by the source model are denoted as MF-Objects; for these objects colour parameters are transmitted instead of motion parameters. While motion and shape parameters can be coded very efficiently [9], [10], most of the data rate needed for image transmission is used for colour parameter coding, which is currently performed by an extended hybrid scheme [21].

A first complete implementation of an object-based analysis-synthesis coder based on the source model of "moving flexible 2D objects with 2D motion" has been developed by Hötter [11]–[13]. This implementation shows an image quality where blocking and mosquito artefacts are nearly completely reduced when compared to conventional hybrid coding. Further improvements of the coding efficiency can be reached by applying more sophisticated source models, as described by Ostermann [18], [19], Harashima [6] or Welsh [24], as well as by applying more efficient colour coding techniques. In this paper an improved colour coding technique using a new prediction concept called "Prototype Prediction" [25] is presented. To demonstrate the efficiency of this technique it is combined with the object-based analysis-synthesis coder described by Hötter [11], [12].

The basic idea of prototype prediction is to exploit the temporal correlation of specific image parts in an image sequence. A similar technique has been proposed by Kato [15] for a block oriented hybrid scheme. Investigations of an object-based analysis-synthesis coder have shown that MF-Objects from a current image are strongly correlated with MF-Objects from previous images [25]. If MF-Objects from previously coded images are stored in a memory, they can be used for the prediction of an MF-Object currently to be coded. Thus, the proposed scheme consists of two algorithms: one is to generate and store up image pattern prototypes (IPPs) containing typical patterns of MF-Objects from previously coded images and the other is to use these prototypes for prediction, applying a suitably defined mapping function. In order to guarantee that the efficiency of the colour coding is always increased, prototype prediction is only applied if its data rate is lower than that of the extended hybrid scheme [21].

The outline of the paper is as follows. In Section II the principle of object-based analysis-synthesis coding is described. In Section III the basic idea of Prototype Prediction, the generation of image pattern prototypes and the prediction of colour parameters of an MF-Object using an IPP memory and a suitable mapping function are described in detail. Also the coding of the side information is explained. In Section IV experimental results are presented and discussed.

Manuscript received October 15, 1993; revised February 21, 1994 and May 1, 1994. It was recommended by Kou-Hu Tzou.

The author is with the Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung, Universität Hannover, Appelstraße 9A, 30167 Hannover, F.R. Germany.

IEEE Log Number 9403081.

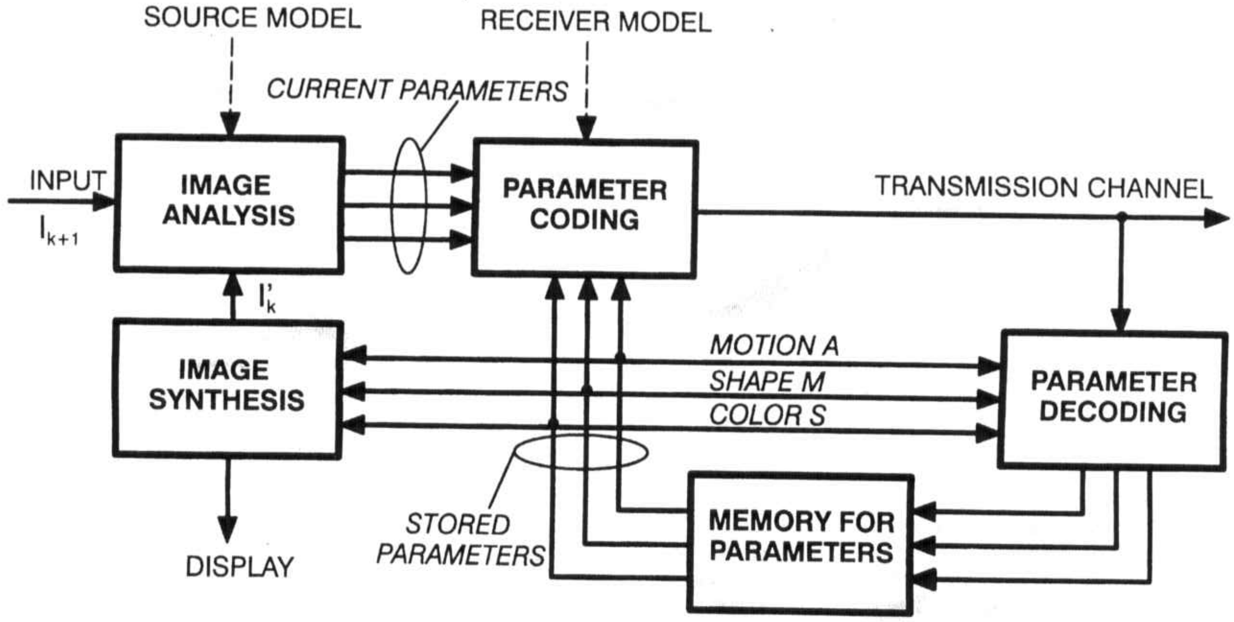

Fig. 1. Block diagram of an object-based analysis-synthesis coder.

II. Principle of Object-Based Analysis-Synthesis Coding

In this section a short description of the principle of object-based analysis-synthesis coding (OBASC) [16] is given. Fig. 1 shows the block diagram of an object-based analysis-synthesis coder. Input of the coder is a sequence of images. The image analysis subdivides each image into arbitrarily shaped moving objects and estimates for each object three parameter sets defining its motion, shape and colour [7], [8]. The estimated parameters depend on the applied source model. Here, the source model "moving flexible 2D objects with 2D motion" [10] is used as an example throughout the paper. For this source model, the colour parameters S denote the luminance and chrominance values of the object surface. The shape parameters M describe the silhouette of the object using a combined polygon/spline approximation [9]. The motion parameters A describe the object motion by a displacement vector field.

The estimated parameter sets are coded, transmitted, decoded and stored in a memory at the coder and at the decoder. Since the parameter memories at the coder and the decoder contain the same information, the image I'_k, also denoted as synthesized image, can be identically reconstructed by the image synthesis [13], [22]. The synthesized image I'_k is displayed at the decoder and is used for image analysis of the next input image I_{k+1} at the coder. Additionally the stored parameters can be used for the coding and decoding of the parameter sets of the next input image I_{k+1}allowing for temporal prediction techniques.

By the source model, the image analysis is based on, temporal luminance changes between the synthesized image I'_k and the current image I_{k+1} are supposed to be caused only by 2D motion of flexible 2D objects. Objects where this assumption holds are referred to as MC-Objects (MC: Model Compliance). Objects which can not be described sufficiently exact by the source model are denoted as MF-Objects (MF: Model Failure) [13]. In order to reduce the overall data rate for image transmission, for MC-Objects only shape and motion parameters are coded, since they are sufficient to describe the object using the applied source model. For MF-Objects, besides the shape parameters colour parameters have to be coded. Since for these objects the motion description by the source model fails, motion parameters are not coded.

To outperform the efficiency of block-oriented hybrid coding, the parameter sets have to be coded efficiently. Therefore, individual parameter coding techniques are applied for each type of parameter set. The motion parameters A are predictively coded using known DPCM techniques [13]. The shape parameters M are predictively coded using the current motion parameters and the stored shape parameters from the previous image [9]. For coding of the colour parameters S, an extended block-oriented hybrid scheme combining DPCM and transform coding is used [21]. Finally, for each object its class (MC or MF) has to be coded and transmitted to the receiver.

In the current implementation of the coder the areas of model failure are about 4% of the total image area [10], [11] for typical videotelephone sequences. Only for these areas colour parameters have to be transmitted, but due to the complexity and the high signal energy of these areas the data rate required for the coding of colour parameters is about 70% of the overall needed data rate, assuming sequences in CIF (Common Intermediate Format) with a reduced frame frequency of 10 Hz and a transmission rate of 64 kbit/s. Thus, a significant increase in colour coding efficiency will also significantly increase the efficiency of the whole coding scheme.

III. Colour Coding Using Prototype Prediction

In order to increase the coding efficiency of an object-based analysis-synthesis coder, in this work the efficiency

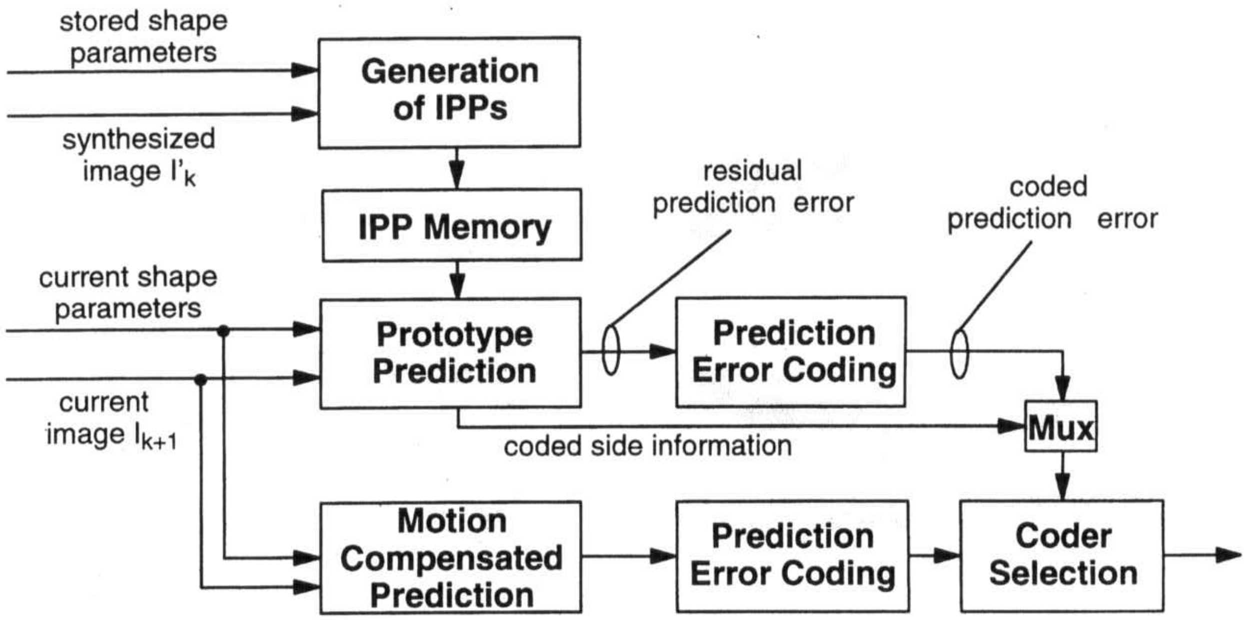

Fig. 2. Block diagram of a colour coder using prototype prediction.

of the colour coding will be improved. Investigations of an object-based analysis-synthesis coder have shown that in an image sequence MF-Objects are strongly correlated [25] (e.g. for opening and closing eyes and mouth in a videotelephone sequence). Thus, the idea is to store these parts of the current image in a memory at the coder and the decoder and to use this memory for prediction of MF-Objects of the following images. The stored image parts, denoted as image pattern prototypes (IPPs), contain typical patterns of MF-Objects.

Since the IPPs are generated using the synthesized image, no additional information has to be transmitted to the receiver. To guarantee that the memory at the coder and the decoder contain identical information, the applied algorithm for generating IPPs has to be the same at the coder and the decoder. In this section the basic structure of a colour coder using prototype prediction is described as well as the generation of IPPs, the control of an IPP memory and the prediction using this memory and an appropriate mapping function.

A. Basic Structure of a Colour Coder Using Prototype Prediction

The block diagram in Fig. 2 is now used to describe the concept of prototype prediction. At the beginning of the image transmission there are no image pattern prototypes at the coder and the decoder. The colour parameters are coded using the extended hybrid scheme [21], *i.e.*, the lower branch in the block diagram. As soon as images are transmitted, IPPs are generated using the synthesized image I'_k and stored in a memory. From then on, colour parameters of MF-Objects can be coded either by applying prototype prediction or motion compensated prediction, each followed by prediction error coding. The decision which scheme to use is performed for each MF-Object by coder selection, depending on which scheme requires less data rate R_S. Then, in the case of prototype prediction the side information, i.e. the IPP memory address of the prototype used for prediction and the calculated mapping parameters, is transmitted to the receiver. For the motion compensated prediction, the side information consisting of shape and motion of the MF-Object is not sent to the receiver explicitly. The shape parameters have already been transmitted by shape parameter coding and the motion parameters (which are not transmitted in the case of MF-Objects) are derived from the underlying MC-Object or set to zero in case of uncovered background. Additionally, for each MF-Object the coder selection information is transmitted to the receiver.

For both schemes the prediction error is coded and transmitted to the receiver. For prediction error coding an adaptive combination of DCT and Intraframe DPCM is used [21]. The area of the MF-Object is subdivided into square blocks and for each block either DCT or DPCM is used, whichever allows a more efficient coding. So the lower branch, i.e. motion compensated prediction and coding of the prediction error with DCT/DPCM, corresponds to the colour coding technique which is exclusively used at the moment, referred to as extended hybrid scheme in the introduction.

B. Generation of IPPs and Control of the IPP Memory

A prediction using an IPP memory assumes that both the memory at the coder and that at the decoder contain identical information. Using a current image from the camera for generating image pattern prototypes causes a drastic increase of the required data rate, because the generated IPPs at the coder have to be coded and transmitted to the decoder. So instead of the current image I_{k+1} the previously synthesized image I'_k is used for the generation of IPPs and thus no additional information has to be transmitted to the receiver.

To guarantee good prediction results using an IPP memory, the IPPs should contain typical patterns of MF-Objects. Because the areas of model failure can have an arbitrary shape, a special algorithm is applied for generating IPPs which extracts

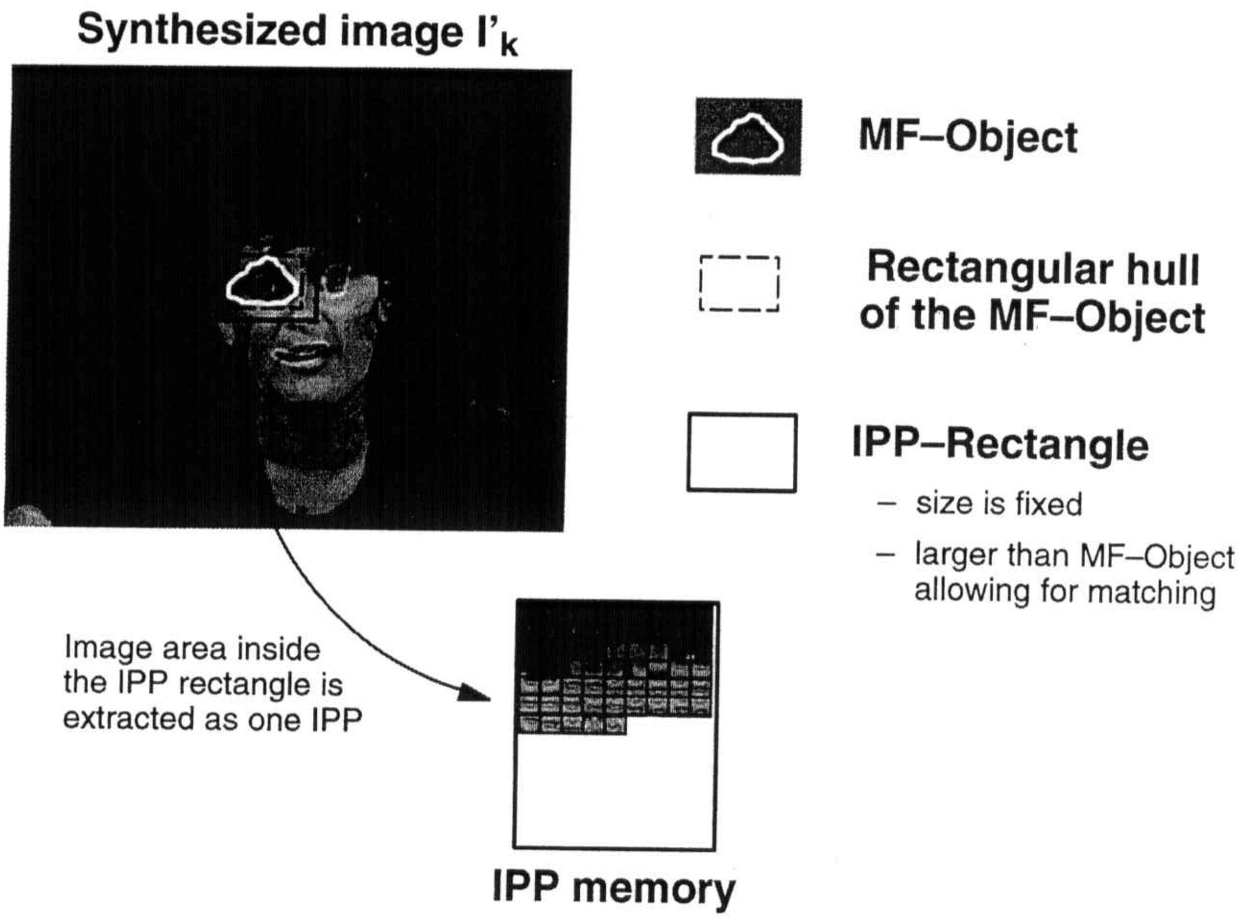

Fig. 3. Generation of an IPP using the synthesized image I'_k. The image area inside the IPP rectangle is extracted as one IPP and stored in the IPP memory.

a rectangular region from the synthesized image containing the whole MF-Object or only parts of it as an IPP (Fig. 3). The algorithm works as follows: first the circumscribing rectangle of the MF-Object is found, then a second rectangle of fixed size, denoted as *IPP rectangle*, is placed in such a manner that the centers of both rectangles coincide. The image area within the IPP rectangle represents an image pattern prototype and is stored in the memory at the coder and the decoder. The size of the IPP rectangle is fixed and, as a rule, larger than an MF-Object, allowing for matching MF-Objects inside the IPP. It has not been optimized yet, but the simulations have shown that a size of 66 pels/line by 54 lines leads to good results for the used test sequences in CIF (Common Intermediate Format: 288 lines, 352 pels/line). The generated prototypes usually contain the MF-Objects plus some surrounding area. Thus it is possible to perform a good prediction for various similar MF-Objects using the corresponding prototype by variation of the mapping parameters.

The control of the IPP memory includes two aspects. First, as the size of the memory has to be limited for implementation reasons, only a limited number of IPPs can be stored. If the memory is full, further prototypes that are generated during the successive image transmission can either be rejected or stored prototypes can be replaced by new ones, applying an appropriate update strategy. Rejecting further prototypes causes the memory or at least parts of it to become out of date, i.e. the stored prototypes are too old and thus no longer suitable for a good prediction. Thus, in the described algorithm old, unsuitable IPPs are discarded and replaced by new ones. This guarantees that the memory is always "up to date" and thus it is possible to adapt the memory to slow variations of the image content. Since the same update strategy is applied at the coder and the decoder, this can be done without additional transmission of information and consequently without increasing the data rate. As a first update strategy the frequency of applications of an IPP is stored and that IPP with the minimum frequency is replaced.

The second aspect of control is the classification of the IPPs, which has to be implemented for two reasons. First, during the above described updating of the memory, a new prototype should replace a "similar" old one, whenever possible; as an example for videotelephone sequences, a new "eye"-prototype should replace an old "eye"-prototype, not a "mouth". Secondly, the classification is to speed up the selection of the optimal IPP for prediction. When an unsorted memory is used, all available IPPs have to be checked, i.e. the prediction has to be performed for all prototypes in the memory. Instead of this full search strategy, the classification of IPPs allows to reduce the number of prototypes that have to be checked.

To assess the similarity of two IPPs, an appropriate similarity criterion has to be found, i.e. the IPPs have to be classified using a suitable classification feature. Using such

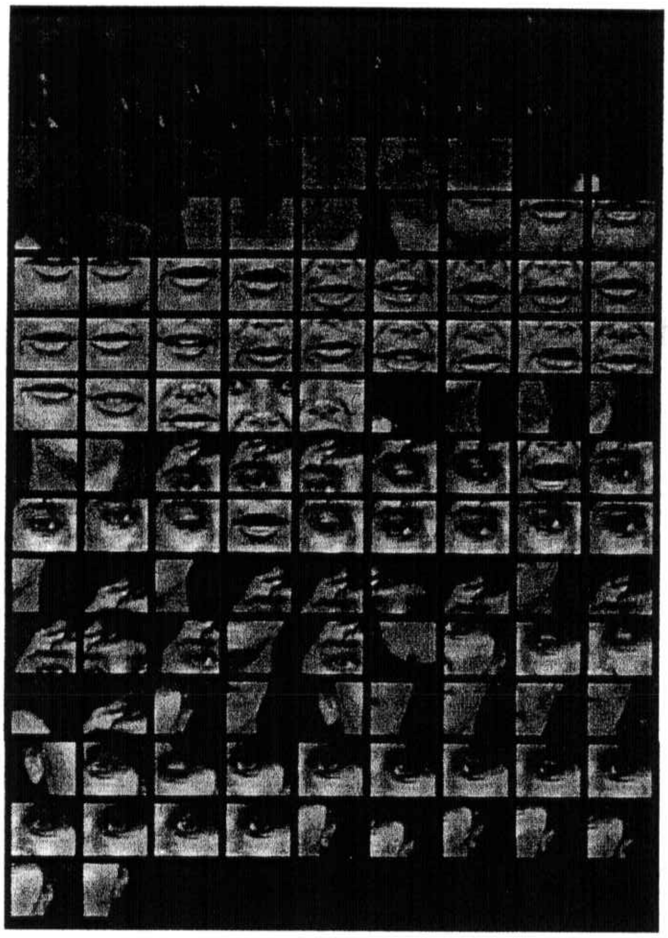

Fig. 4. IPP Memory, extracted from *the test sequence Miss America* (CIF, 10 Hz) and sorted due to a combination of mean and variance of the IPP luminance values.

a classification, it is possible to organize the IPP memory in subsets. Each subset contains a special class of IPPs. In this work a combination of the statistical parameters mean and variance of the IPP's luminance values is used as classification feature. To organize the memory, in a first step the IPPs inside the memory are arranged in ascending order according to the variance of their luminance values. Then they are divided into a number of subgroups using relative thresholds. In a second step, for each subgroup all IPPs in the subgroup are arranged in ascending order according to their mean luminance values.

The subset of IPPs corresponding to a new IPP or to an MF-Object to be coded is found as follows: first that IPP inside the memory is found which is most similar to the new IPP or to the MF-Object, using the ordering algorithm described above. Then those IPPs within a given range "around" the most similar IPP form the corresponding subset that is used for update or for prediction. Fig. 4 shows an example of an IPP memory containing 128 prototypes (36 lines by 44 pels/line) generated during the transmission of the test sequence *Miss America* [2]. The memory has been organized using the algorithm described above.

Using such an organized IPP memory, the exchange of an old IPP for a new one is done only inside the subset of IPPs corresponding to the new prototype. For the prediction, only prototypes of the same class as the MF-Object to be coded, denoted as *candidates*, are checked. This leads to a decrease of about 70% in computational load, concerning an IPP memory with 256 prototypes and for image sequences in CIF at a transmission rate of 64 kbit/s.

C. Prediction of MF-Objects Using an Appropriate Mapping Function

For the prediction of the colour parameters of an MF-Object, all candidates are mapped onto the MF-Object while optimizing the mapping parameters for each candidate. As quality criterion the mean square error (MSE) between the mapped candidate and the MF-Object is used, where the MSE is only evaluated in the area of the MF-Object. The candidate leading to the least MSE is used for prediction. The applied mapping function has a great influence on the prediction quality. As a first basic mapping function a displacement with one pel amplitude resolution was used. In a second step, the resolution of the displacement was increased to halfpel, allowing for more accurate mapping results. The third mapping function consists of a halfpel displacement combined with an additional luminance transformation. In the following, the pel and halfpel resolution displacement and the applied luminance transformation will be described in detail.

Displacement as Mapping Function To find the optimal displacement vector $\vec{D}_{\mathrm{opt}}$ between an IPP and the MF-Object, in a first step the initial position has to be determined, corresponding to a displacement vector $\vec{D} = \vec{0}$. Therefore the MF-Object and the IPP are positioned in such a way that the centers of the IPP and the circumscribing rectangle of the MF-Object coincide. Then, the optimal displacement vector $\vec{D}_{\mathrm{opt}}$ is calculated, using a full search inside an area of $\pm k$ pel in each direction. In Fig. 5, an example for the optimization of the displacement vector $\vec{D}$ for a specific IPP is shown. Here, the mean squared luminance difference $\mathrm{MSE}(\vec{D})$ between the MF-Object O_{MF} and the displaced prototype T(IPP), depending on the displacement vector $\vec{D}$, is described by

$$\mathrm{MSE}(\vec{D}) = \frac{1}{A_{O_{\mathrm{MF}}}} \cdot \sum_{(x,y)\in O_{\mathrm{MF}}} [L_{O_{\mathrm{MF}}}(x,y) - L_{T(\mathrm{IPP})}(x,y)]^2 \tag{1}$$

where the area of model failure $A_{O_{\mathrm{MF}}}$ and the mapped prototype $T(\mathrm{IPP})$ are

$$\begin{aligned} T(\mathrm{IPP}) &= \{(x,y)|(x - D_x, y - D_y) \in \mathrm{IPP}\}, \\ A_{O_{\mathrm{MF}}} &= \sum_{(x,y)\in O_{\mathrm{MF}}} 1 \end{aligned} \tag{2}$$

In (1), the luminance values of the MF-Object and of the displaced IPP at the position (x,y) are denoted as $L_{O_{\mathrm{MF}}}(x,y)$ and $L_{T(\mathrm{IPP})}(x,y)$, respectively. The displacement vector $\vec{D}$ leading to the minimal $\mathrm{MSE}(\vec{D})$ is the optimal displacement vector $\vec{D}_{\mathrm{opt}}$.

This procedure is performed for each candidate, i.e. for each prototype in the memory that has to be checked. The candidate which leads to the least MSE, compared to all other candidates, is selected for prediction, taking into account the calculated optimal displacement vector $\vec{D}_{\mathrm{opt}}$. Thus, the side information

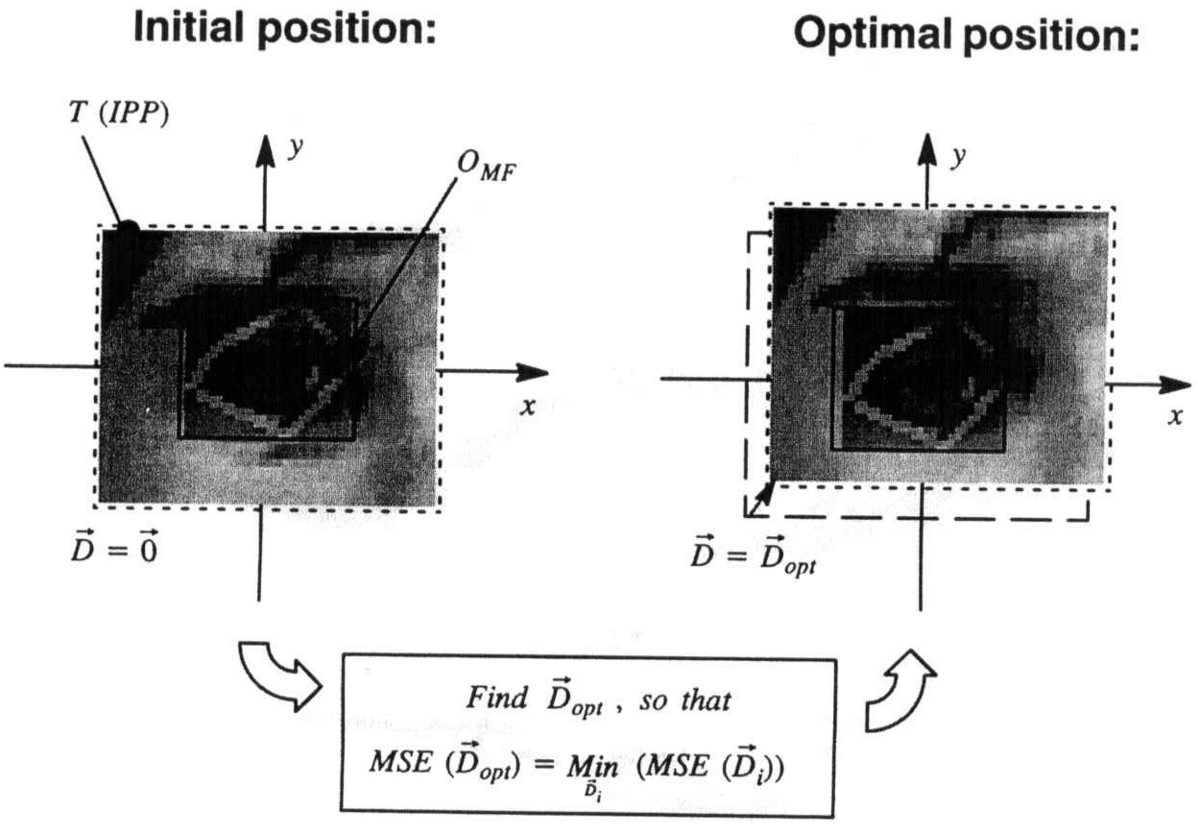

Fig. 5. Calculation of the optimal displacement vector between an IPP and an MF-Object.

for this mapping function consists of the memory address of the selected IPP and the corresponding optimal displacement vector $\vec{D}_{\rm opt}$.

Luminance Transformation as Additional Mapping Function To increase the quality of the prediction and thus to increase the efficiency of the colour coding, an additional luminance transformation λ is applied. It serves to adjust the contrast and the brightness of the selected and already displaced IPP to the contrast and brightness of the MF-Object. There is a pair of parameters $(a_{\rm opt}, b_{\rm opt})$ that minimizes the squared error between the luminance values $L_{O_{\rm MF}}$ of the MF-Object and the luminance values $L_{T({\rm IPP})}$ of the selected and already displaced IPP, according to

$$\begin{aligned} e^2(L_{O_{\rm MF}}, \lambda(a, b, L_{t({\rm IPP})})) &= \frac{1}{A_{O_{\rm MF}}} \sum_{(x,y)\in O_{\rm MF}} \left[L_{O_{\rm MF}}(x,y) - (aL_{T({\rm IPP})}(x,y) + b)\right]^2 \\ &\stackrel{!}{=} \min \end{aligned} \tag{3}$$

To find the minimum of the error function e, the partial derivations of $e^2(a, b)$ must be set to zero. This leads to the optimal parameters

$$a_{\rm opt} = \frac{C_{\rm MF,IPP}(0,0)}{\sigma_{\rm IPP}^2} \qquad b_{\rm opt} = \mu_{\rm MF} - a_{\rm opt} \cdot \mu_{\rm IPP} \tag{4}$$

In (4), $C_{\rm MF,IPP}$ denotes the covariance between $L_{O_{\rm MF}}$ and $L_{T({\rm IPP})}$. The mean and the variance of the luminance values $L_{T({\rm IPP})}$ and $L_{O_{\rm MF}}$ are denoted as $\mu_{\rm IPP}, \sigma_{\rm IPP}^2, \mu_{\rm MF}$ and $\sigma_{\rm MF}^2$, respectively. Because the IPP is known at the decoder, only the covariance $C_{\rm MF,IPP}$ and the mean luminance value $\mu_{\rm MF}$ have to be quantized and transmitted to the decoder. Thus, in case of additional luminance transformation the side information consists of the address of the selected IPP, the optimal displacement vector $\vec{D}_{\rm opt}$, the covariance $C_{\rm MF,IPP}$ and the mean luminance value $\mu_{\rm MF}$.

After the prediction has been performed, the resulting prediction error is coded and send along with the needed side information to the receiver. At the receiver the colour parameters of the MF-Object can be reconstructed using the selected IPP, the displacement vector $\vec{D}_{\rm opt}$, the optimal luminance transformation parameters $a_{\rm opt}$ and $b_{\rm opt}$ (if luminance transformation is applied) and the coded prediction error. The initial position for the displacement of the IPP is determined in the same way as at the coder.

D. Coding of Prediction Error and Side Information

The residual prediction error is coded using a combination of DPCM and DCT coding techniques. Therefore the area of the MF-Object is subdivided into square blocks and each block is coded either using DCT or DPCM, whichever allows more efficient coding. This scheme is also used for the prediction error coding when motion compensated prediction is used, i.e. in the lower branch of the block diagram in Fig. 2.

The side information consists of the memory address of the IPP, the displacement vector $\vec{D}_{\rm opt}$ and, in case of luminance transformation, of the covariance $C_{\rm MF,IPP}$ and the mean luminance value $\mu_{\rm MF}$. Because the data rate for the transmission of

TABLE I
NUMBER OF MF-OBJECTS FOR WHICH PROTOTYPE PREDICTION WAS APPLIED IN PERCENT OF ALL CODED MF-OBJECTS. RESULTS ARE SHOWN FOR SIMULATIONS USING DIFFERENT MAPPING FUNCTIONS FOR PROTOTYPE PREDICTION AND WITH DIFFERENT IMAGE QUALITIES (MEASURED IN PSNR) IN THE AREA OF MF-OBJECTS

Applied mapping Function	Image quality in the area of MF-objects 34dB	36dB	38dB
pel displacement	25%	30%	40%
halfpel displacement	42%	50%	58%
halfpel displacement plus luminance transformation	40%	46%	52%

the side information is very low, all side parameters are PCM coded, using linear quantization. Applying entropy coding techniques as e.g. huffman or arithmetic coding may decrease the data rate and will be applied in future.

IV. EXPERIMENTAL RESULTS

To assess the efficiency of colour coding using prototype prediction, several simulations have been done. Therefore, the described scheme has been implemented in an object-based analysis-synthesis coder using the source model of "2D flexible objects with 2D motion" as described in [11] [12]. Using the image format CIF (Common Intermediate Format) with a reduced frame frequency of 10 Hz, the test sequences *Miss America* [2] and *Claire* [5] have been coded at a data rate of about 64 kbit/s. For prototype prediction three different mapping functions as described in Section III were used, the maximum size of the IPP memory was set to 256 prototypes and the size of the IPPs was 66 pels/line by 54 lines.

To judge the efficiency of prototype prediction, two aspects can be considered. First, the number of MF-Objects for which prototype prediction has been applied (and not motion compensated prediction, i.e. the lower branch in the block diagram) gives an impression of the prediction quality, i.e. the more prototype prediction is applied, the better the prediction. The results for three different mapping functions, i.e. pel displacement, halfpel displacement and halfpel displacement with additional luminance transformation are shown in Table I.

As can be seen, the best prediction results are achieved using halfpel displacement. The additional luminance transformation does not increase the selection frequency of prototype prediction, though the prediction quality is slightly increased. This result shows, that the main prediction gain is achieved by the geometric transformation, i.e. the halfpel displacement. Thus, if the prediction quality due to the displacement outperforms the prediction quality by motion compensation (i.e. the lower branch in the block diagram), prototype prediction is selected, regardless of whether an additional luminance transformation is applied or not. Vice versa, if the prediction quality due to the displacement does not outperform the motion compensated prediction, the additional luminance transformation is not sufficient in order to increase the prediction quality of prototype prediction as much as necessary to outperform motion compensated prediction.

Secondly, the different mapping functions have different amount of side information which has to be sent to the receiver. Thus, the efficiency of the whole colour coding scheme can only be assessed by the decrease of the data rate required for the whole image transmission, maintaining the same image quality or, vice versa, by the increase in image quality, maintaining the same transmission data rate. The corresponding results are shown in a diagram in Fig. 6, showing the image quality in the area of the MF-Objects, measured in peak signal to noise ratio (PSNR), over the needed data rate for image transmission. Three simulations using prototype prediction with three different mapping functions are mutually compared. Additionally, a simulation with colour coding using exclusively the extended hybrid scheme is used as a reference.

As can be seen from the diagram, all three simulations where prototype prediction is applied outperform the efficiency of the colour coding without prototype prediction. This is due to the strategy of applying prototype prediction only for those MF-Objects which are coded more sufficiently this way. Further it can be seen, that the most efficient mapping function is not the halfpel displacement with additional luminance transformation, but the halfpel displacement without luminance transformation. This is caused by two reasons: first, the side information is increased by the additional mapping parameters of the luminance transformation. Secondly, the results show that in most cases the data rate required for coding of the prediction error is increased when the luminance transformation is applied. Since the prediction quality, measured in PSNR, is slightly increased by the luminance transformation, this result is unexpected. It could be due to the fact that by the luminance transformation the prediction error is shifted into higher frequency bands. Since the analysis of the prediction error signal has not been finished, this explanation is not proved yet.

Further the results show, that for some MF-Objects the luminance transformation really decreases the data rate required for prediction error coding. Taking into account this effect, the overall efficiency will be increased if an additional luminance transformation is only applied in these cases. This strategy has been applied, but the increase in efficiency is very small. Thus, the corresponding curve in the diagram would be nearly the same as the curve for the halfpel displacement without luminance transformation.

Finally, the results of applying prototype prediction for colour coding are compared to a simulation where only the extended hybrid scheme is used for colour coding, i.e. without prototype prediction , see also Fig. 6. For this comparison the simulation using prototype prediction with an IPP memory of 256 IPPs of 66×54 pels and a halfpel displacement as mapping function is used. The results show that the image quality in the area of MF-Objects, measured in PSNR, is increased by 2 dB when prototype prediction is applied, maintaining the same transmission rate of 64 kbit/s. Vice versa, the required data rate for image transmission is decreased by about 10%, maintaining the same image quality in the area of MF-Objects.

To demonstrate the improvement of the subjective image quality due to the increased PSNR of 2 dB, in Fig. 7 different coded image areas including eye and mouth regions are shown.

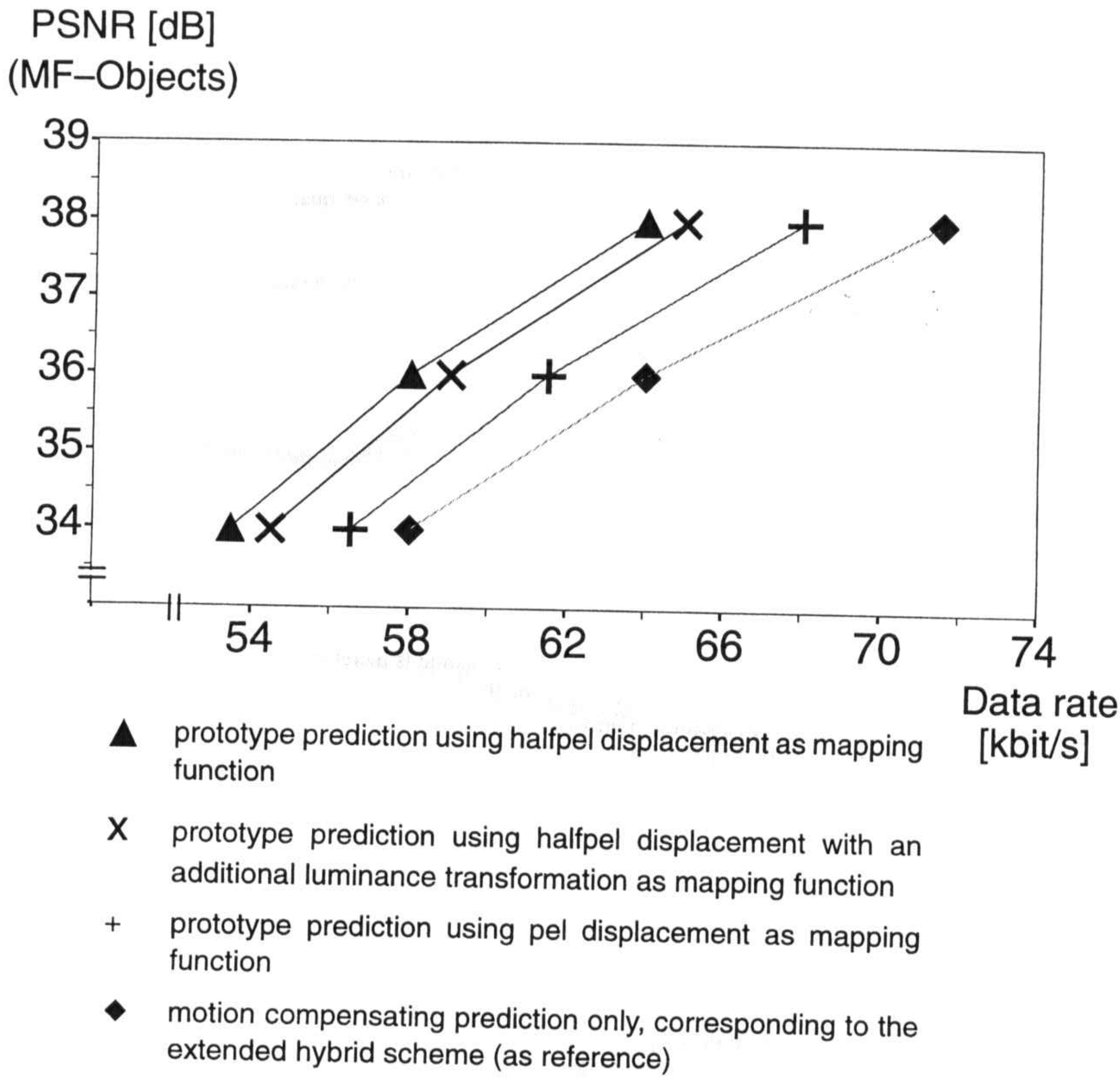

Fig. 6. Simulation results applying different mapping functions for prototype prediction mutually compared and additionally compared to colour coding without prototype prediction. The diagram shows the image quality in the area of MF-Objects over the data rate needed for image transmission.

Both eye and mouth regions were colour updated with an image quality measured in PSNR of 36 dB, corresponding to colour coding only using the extended hybrid scheme, and 38 dB, corresponding to colour coding applying prototype prediction. The image areas have been extracted from the test sequences "Miss America" and "Claire" in CIF with a frame rate of 10 Hz, coded at a data rate of 64 kbit/s. As can be seen, the subjective image quality of these critical regions is improved when prototype prediction is applied. The image looks sharper due to the reduced quantization noise. Also the contouring effects, which are visible around the mouth and eye regions with a PSNR of 36 dB, are not visible in the image with 38 dB. Because the areas of MF-Objects, especially mouth and eye regions, are very crucial for human observers, the improved subjective image quality of these regions leads to an improved overall subjective image quality of the coded image sequence.

V. Conclusion

In this paper a coding scheme for coding the colour parameters of MF-Objects in an object-based analysis-synthesis coder is described. The core of the scheme is a prediction technique called prototype prediction, which is taking into account the spatial and temporal similarities of MF-Objects in successive images. For specific image sequences like videotelephone sequences, MF-Objects of the current image are often very similar to MF-Objects of previously coded images. To exploit these similarities, during the image transmission image pattern prototypes (IPPs) are generated, containing typical patterns of MF-Objects. Since the IPPs are generated from transmitted images, they can be stored in a memory at the coder and the decoder without additional transmission of information. Thus prototype prediction allows a prediction from previously coded images, i.e. from more than one preceding image, without

(a) (b)

Fig. 7. Simulation results of colour coding applying prototype prediction compared to colour coding without prototype prediction. Extracted image areas from the test sequences *Miss America* and *Claire* in CIF/10 Hz, coded at a data rateof 64 kbit/s. Both eye and mouth regions were colour updated with an image quality of: (a) PSNR = 36 dB corresponding to colour coding without prototype prediction; (b) PSNR = 38 dB corresponding to colour coding applying prototype prediction.

storing the whole images but only those image areas which are useful for lateron prediction. The residual prediction error is coded using a combination of DCT and DPCM and sent along with the coded side information to the receiver. To guarantee that the efficiency of the colour coding is always increased, prototype prediction is only applied when its data rate is lower than that of an extended hybrid scheme, as it is has been used exclusively for colour coding up to now.

The first aspect of prototype prediction is the generation of IPPs and the control of the IPP memory. Therefore an algorithm is applied which generates IPPs containing typical patterns of MF-Objects from a transmitted image. For image sequences in CIF (Common Intermediate Format) image pattern prototypes of 66 pels/line by 54 lines are used. In order to simplify the prototype prediction the IPP memory is organized in subsets. Each subset contains a special class of IPPs. A class of IPPs is defined by a classification feature, i.e. all IPPs with a similar classification feature belong to one class. Here, statistical parameters like the mean and the variance of the luminance values of an IPP are used as classification features. This leads for an IPP memory of 256 image pattern prototypes to a number of 6 to 10 subsets, each containing 20 to 30 IPPs. The classification of IPPs reduces the number of prototypes to be checked for prediction. Only IPPs of the same class as the MF-Object to be coded are checked, leading to a decrease of about 70% in computational load. Since the size of the IPP memory is limited, another task of the control is to update the memory when its maximum size is reached, i.e. to exchange old prototypes for new ones. As a first update strategy the frequency of applications of an IPP is stored and that one with the minimum frequency is exchanged. The exchange is done inside the subset of IPPs corresponding to the new prototype. Because of an insufficient length of the test sequences, the optimal maximum size of the IPP memory has not been found yet. A size of about 256 prototypes gives good results for the used test sequences. Also the update strategy has not been optimized due to the same reason.

The second aspect of prototype prediction is to find an appropriate mapping function for the prediction. Three different mapping functions have been investigated. Two of them are displacements, one with pel and the other with halfpel amplitude resolution. The third mapping function consists of a halfpel displacement with an additional luminance transformation, adjusting the contrast and the brightness of the IPP to that of the MF-Object. The results have shown that this additional luminance transformation leads only to a slightly improved prediction quality. But also the side information is increased and the prediction error signal is influenced in such a way, that the coding of the prediction error requires more data rate than without luminance transformation for most MF-Objects. Thus, the most efficient mapping function for the prediction is the halfpel displacement.

Finally, the simulation results of colour coding using prototype prediction are compared to an implementation where only the extended hybrid scheme is used. For prototype prediction an IPP memory of 256 IPPs of 66×54 pels was used, applying a halfpel displacement as mapping function. It is shown that applying prototype prediction increases the image quality in the area of MF-Objects, measured in PSNR, by 2 dB while maintaining the same data rate of 64 kbit/s. Vice versa, the required data rate for image transmission can be decreased by about 10% while maintaining the same image quality in the area of MF-Objects.

Future work will concentrate on improving the mapping function and the update strategy.

ACKNOWLEDGMENT

The author wishes to thank Prof. Musmann for encouraging his work and many helpful discussions. Further, the author wishes to thank Dr. Hötter for many discussions about OBASC and prototype prediction and Mr. Schiller for his software support and his suggestions concerning the subject of prediction error coding. This work has been supported by the Deutsche Forschungsgemeinschaft (DFG).

REFERENCES

[1] M. Bierling, "Displacement estimation by hierarchical blockmatching," *3rd SPIE Symposium on Visual Communications and Image Processing,* Cambridge, MA, USA, pp. 942–951, Nov. 1988.
[2] British Telecom Research Lab (BTRL), "test sequence *Miss America,* CIF, 10 Hz, 50 frames," Martlesham, Great Britain.
[3] CCITT, "Draft revision of recommendation H.261: video codec for audio visual services at $p \times 64$ kbit/s," *Study Group XV, WP /1/ Q4,* Specialist group on coding for visual telephony, Doc. no. 584, Nov. 1989.
[4] CCITT SG XV, Doc. #525, *Description of Ref. Model 8 (RM8),* 1989.
[5] Centre National d'Etudes des Telecommunication (CNET), "test sequence *Claire,* CIF, 10 Hz, 156 frames," Paris, France.
[6] H. Harashima, K. Aizawa and T. Saito, "Model-based analysis synthesis coding of videotelephone images—conception and basic study of intelligent image coding—," *The Transactions of the IEICE,* vol. E 72, no. 5, pp 452–459, May 1989.
[7] M. Hötter and R. Thoma, "A selfadaptive model based segmentation algorithm," *PCS'88,* Torino, Italy, 2.2, Sept. 1988.
[8] M. Hötter and R. Thoma, "Image segmentation based on object oriented mapping parameter estimation," *Signal Processing,* vol. 15, no. 3, pp. 315–334, Oct. 1988.
[9] M. Hötter, "Predictive contour coding for an object-oriented analysis-synthesis coder," *IEEE International Symposium on Information Theory,* San Diego, California, U.S.A., p. 75, Jan. 1990.
[10] M. Hötter, "Optimization of an object-oriented analysis-synthesis coder based on the model of flexible 2D-objects," *Picture Coding Symposium (PCS '91),* Tokyo, Japan, no. 10.4, Sept. 1991.
[11] M. Hötter, "Optimization and efficiency of an object-oriented analysis-synthesis coder", *IEEE Trans. on Circuits. and Sys. for Video Technol.,,* vol. 4, no. 2, pp. 181–194, Apr. 1994.
[12] M. Hötter, *Objektorientierte Analyse-Synthese-Codierung basierend auf dem Modell bewegter, zweidimensionaler Objekte,* Fortschr.-Ber. VDI Reihe 10 Nr. 217, Düsseldorf, VDI-Verlag, 1992.
[13] M. Hötter, "Object-oriented analysis-synthesis coding based on moving two-dimensional objects," *Signal Processing: Image Communication,* vol. 2, no. 4, pp. 409–428, Dec. 1990.
[14] Commitee Draft of Standard ISO 11172, Coding of moving pictures and associated audio, ISO/MPEG 90/176, Dec. 1990.
[15] Y. Kato, "Transformation compensation for videophone image coding," *PCS '88,* Torino, Italy, 12.5, 12–14, Sept. 1988.
[16] H. G. Musmann, M. Hötter and J. Ostermann, "Object-oriented analysis-synthesis coding of moving images," *Signal Processing: Image Communication,* vol. 1, no. 2, pp. 117–138, Oct. 1989.
[17] 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, Feb. 1990.
[18] J. Ostermann, "An analysis-synthesis coder based on moving flexible 3D-objects," *PCS 93,* Lausanne, Switzerland, 17–19 Mar. 1993, 2.8.
[19] J. Ostermann, "Object-oriented analysis-synthesis Coding based on the source model of moving rigid 3D objects," submitted for publication to *Signal Processing: Image Communication.*
[20] R. Plomben, Y. Hatori, W. Geuen, J. Guichard, M. Guglielmo and H. Brusewitz, "Motion Video Coding in CCITT SG XV—The Video Source Coding," Proc. *IEEE GLOBECOM,* vol. II, pp. 31.2.1—31.2.8, Dec. 1988.
[21] H. Schiller and M. Hötter, "Investigations on colour coding in an object-oriented analysis-synthesis coder," *Signal Processing: Image Communication,* vol. 5, no. 4, pp. 319–326, Oct. 1993.
[22] D. Thalman, *Image Synthesis: Theory and Practice,* Springer, Tokyo, 1987.
[23] R. Thoma and M. Bierling, "Motion compensating interpolation considering covered and uncovered background," *Signal Processing: Image Communication,* vol. 1, no. 2, pp. 191–212, Oct. 1989.
[24] B. Welsh, Model-based coding of images, *Ph.D.-thesis,* Essex University, Great Britain, Jan. 1991.
[25] M. Wollborn, "Object-Oriented Analysis-Synthesis Coding Using Prototype Prediction for Colour Update," *PCS '93,* Lausanne, Switzerland, 17.6, March 17–19, 1993.

Michael Wollborn received the Dipl.-Ing. degree in electrical engineering from the University of Hannover, Hannover, Germany, in 1991.

Since 1991 he is with the "Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung" at the University of Hannover (Germany), where he works in the areas of image analysis and image coding. During his work he is involved in several internal projects and a research project supported by the Deutsche Forschungsgemeinschaft (DFG) dealing with the coding of image sequences at low bit rates. His present research interests and activities cover motion estimation and compensation, image analysis and the application of object-based image coding for very low bit rates.