透明度情報を領域表現に用いた動画像符号化

角野 真也[1)]　津田 賢治郎[1)]　ブン チュンセン[1)]　栄藤 稔[2)]
{skadono,boon}@drl.mei.co.jp　etoh@crl.mei.co.jp
松下電器産業株式会社
映像音響情報研究所[1)] 中央研究所[2)]
〒571 大阪府門真市大字門真[1)]　〒619-02 京都府相楽郡精華町[2)]

我々は、既に画像の階層化表現と多重テンプレートを用いた画像符号化方式を提案している。この提案方式では、画像を視線方向の前後関係、領域の占有／透明度情報により重なり合う画像の合成として表現する。階層表現された画像を符号化する際には、通常の輝度・色差の画素値情報以外に領域の占有／透明度情報も符号化することが必要となる。本報告では、チェイン符号、4分木、DCTなど各種の領域情報符号化方式の比較および領域適応型変換符号化の比較を行った。実験により、マクロブロック単位の動き補償DCT符号化では、4分木による領域符号化と物体境界の無効画素に画素値補充を行うDCTが適していることを示す。

An Image Coding Scheme Using Opacity Information

Shinya Kadono[1)]　Kenjirou Tsuda[1)]　C.S. Boon[1)]　Minoru Etoh[2)]

Audio Video Information Technology Laboratory[1)]　Central Research Laboratory[2)]
Matsushita Electric Industrial Co., Ltd.
[1)]Kadoma, Osaka 571　[2)]Seika, Kyoto 619-02, Japan

We have already proposed an image coding scheme using layered representation and multiple templates. An image sequence is decomposed into layers of image sequences with depth ordering. In addition to pixel values of luminance and chrominance, opacity information is coded in this scheme. In this paper, we report the comparison result of region (opacity) coding methods such as chain coding, quadtree coding and DCT coding; we also evaluate shape adaptive DCT coding. We conclude a quadtree based opacity representation and DCT with shape adaptive value padding are suitable for current DCT based coding.

1. Introduction

Recently, we encounter synthetic image sequences more often than before in several occasions such as movies, advertisement films, animation, and games. These sequences are synthesized from natural and artificial images. Although the synthetic images are of 2D and/or 3D structures, the conventional coding standards (MPEG1/2 [LG91,JTC93]) abandon the structure and encode the image just as projected imagery. By preserving the structure in encoding, we can provide new functionality to a future image coding standard (e.g.,MPEG4 [AOE94]). It is anticipated that the new functionality is content-based which enable data access is based on the audio-visual content by using various accessing tools such as indexing, hyper-linking, querying, browsing, uploading, downloading, and deleting.

With this observation, we have proposed a layered image coding scheme that preserves 2.1D structure[EBK95], where 2.1D means 2D regions with depth ordering[NM90]. The layer decomposition, in which 2.1D description is automatically obtained from an image, is beyond the scope of this paper. We primarily start with synthetic images.

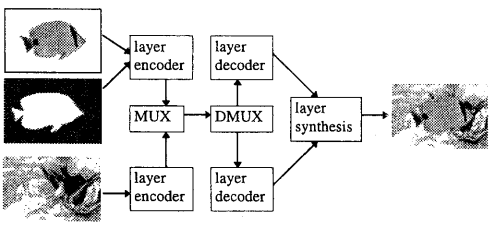
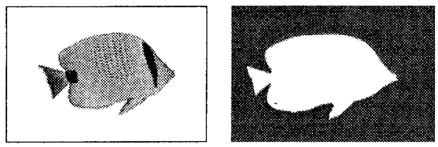


Fig. 1 A layered image coding

Fig. 1 depicts a layered image coding scheme. Although the concept of this scheme was originated by Adelson[Adel94], we cannot straightforwardly apply the scheme to image coding due to irregular deformation caused by deformable objects and/or perspective projection. To cope with this real problem, we have proposed two extensions:

- efficient waveform coding of color and alpha plane with global and local deformation
- use of multiple templates



color plane alpha plane

Fig. 2 Color and opacity planes

Fig. 2 depicts a structure of color and alpha planes in which the occupancy of objects is indicated by alpha values varying from 0 (transparent) to 1 (opaque) at each pixel position. In computer graphics, the image structure is often expressed as "RGBa". For a layered image coding in real situation, we should consider not only color coding however also opacity (transparency) coding over frames. In general, chroma-keying process is frequently used to generate alpha values in video production. In this case, almost all alpha values are 0 or 1 while other values seldom occur. Encoding binary alpha values using gray scale coding scheme such as DCT might be inefficient. On the other hand, binary scale coding schemes used in facsimile are frame based, and cannot be used with usual block based motion compensation. When an alpha value is 0, the corresponding pixel is meaningless. Encoding pixels only with non-zero values, therefore, improves coding efficiency[Gea89,SM94].

In this paper, we report the comparison result of region (opacity) coding methods such as chain coding, quadtree coding and DCT coding. Furthermore, we evaluate shape adaptive transform coding.

In the next section, we introduce our layered image coding scheme. In the following sections, we describe the opacity coding and shape adaptive color coding of our current implementation. In section 5, we discuss experimental results and justify our coding scheme.

2. Overview of Coding Scheme

2.1. Prediction Structure

In our coding scheme, an image consists of color and alpha planes that represent luminance intensity and opacity/transparency of objects. To obtain much higher coding efficiency, the deformation of objects must be described by a few global deformation parameters. We adopt a second order function of image coordinates that covers the following motion models[Bea78]:

1. translational motion
2. affine motion
3. planar motion

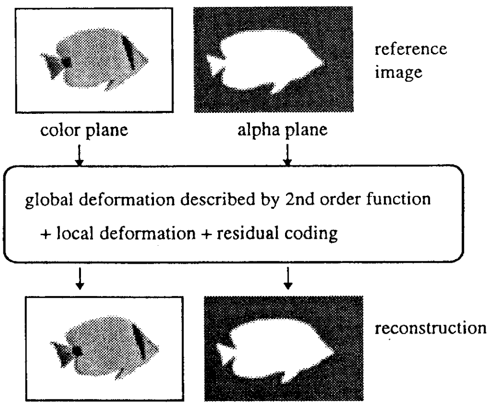


Fig. 3 Color and alpha plane coding

We cannot believe that the above deformations are valid for all deformable objects in general. Therefore, as illustrated in Fig. 3, we incorporate local deformation and a waveform residual coding with the affine or second order global deformation.

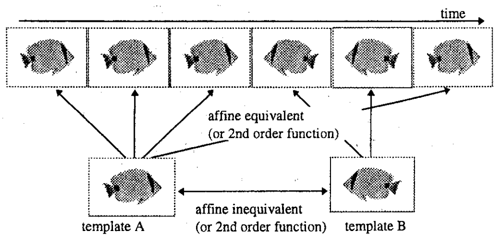


Fig. 4 Color and alpha plane coding

Fig. 4 depicts the idea of multiple templates for each layer image. The multiple templates are a generalization of "IBBPBBP" structure of MPEG1/2. An image sequence is analyzed surch that pictures, to be approximately transformed from a same template by affine parameters (or the second order function if necessary), make a cluster. We can expect high coding efficiency when a long image sequence is reconstructed from a sufficiently small number of templates using only deformation parameters..

2.2. Template Deformation

Each of the input images is coded by the following four steps: 1. template selection, 2. global deformation from the selected template, 3. local deformation from the globally-deformed template, and 4. residual coding of the globally and locally deformed template. Step 3 and 4 are optional, according to the deformation result and given bit-rate.

2.2.1. Global Deformation

For the global deformation is in good accordance to an affine motion, the affine coefficients and its fitting error can be obtained as described in [EBK95]. In real situations, however, the above affine fitting might cause significant fitting error, because there exist outliers. Therefore, we adopt a robust regression technique.

2.2.2. Local Deformation

Regarding the local deformation, 16×16 window is used for SAD matching. The SAD matching is performed between the input image and the best-matched template after global deformation. The local motion vectors are calculated at every 16×16 window positions. We also use the local energy function for a regularization process as used in [Eto94] to obtain a smooth motion field. The local deformation is performed in a manner of overlapped block motion compensation [AKOK92] using the local motion vectors.

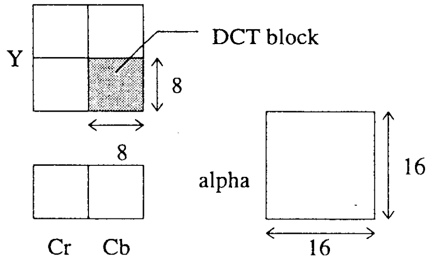


Fig. 5 Macroblock structure

3. Opacity Coding

Macroblock structure of MPEG1/2 is extended to describe the opacity of object as illustrated in Fig. 5. 16×16 size block of alpha plane is added into a macroblock. DCT and quadtree structure were candidates for alpha plane coding. In generall, there are two types of opacity information. One is generated by computer graphics and it has continuous values varying from 0 to 1. The other is made by chroma-keying process and almost all the values are 0 or 1. In our implementation, we apply DCT for the former and quadtree structure for the latter.

For DCT-based coding, 16×16 size block is subsampled to quarter size and 8×8 size DCT is adopted. The subsequent process is the same as a chrominance block. For quadtree based coding, 16×16 size block is coded directly as follows.

3.1.Quadtree structure

Quadtree coding process has the following four steps.

1. truncate gray alpha value to binary
2. make index of lowest level and convert to label of lowest level
3. make index of each layer from lower level label and convert to label
4. encode index from highest level

At first, the gray scale alpha plane is truncated to binary data. Then the alpha block is subdivided into 64 sub-blocks that consists of 2×2 pixels. This forms the lowest level of the quadtree structure. In each sub-block, an index is assigned to each alpha value scanned in the order shown in Fig. 6 Order of concatenation For example, the indices of Fig. 6(b) is "TTTT", and label of Fig. 6(c) is "TOOO" when "T" indicates transparent and "O" indicates opaque. Then the indices are converted to a trinary label as follows.

Label is "T" when the indices are "TTTT".

Label is "O" when the indices are "OOOO".

Label is "M" otherwise.

In l, the next levels, four sub-blocks in the lower leve are grouped to form a sub-block. Fig. 7 depicts the quadtree structure of an alpha block. Each index is by concatenating lower level labels in order of Fig. 6(a) and then the indices are converted to trinary labels. In such way, we build a total of 85 (=1+4+16+64) labels for each macroblock.

The index is coded with VLC from higher level to lower level when label is "M". When label is "T" or "O", it means that all sub-blocks in lower level and that sub-block have same label ("T" or "O"), therefore they can be decided from higher index. Only indexes of lowest level are different from other level, we use another VLC table for lowest level.

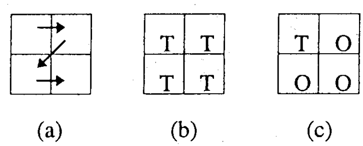


Fig. 6 Order of concatenation

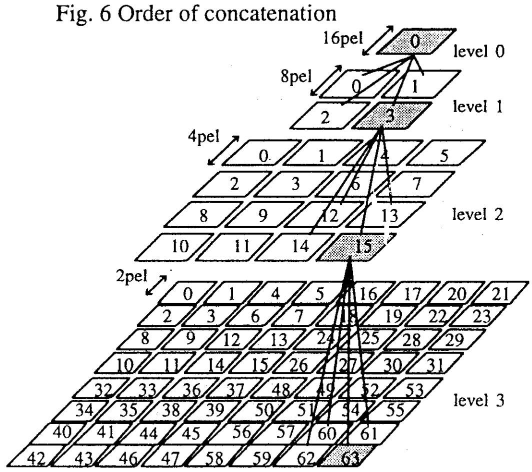


Fig. 7 Quadtree structure

An alpha block is intra/inter frame coded using the same motion vector of luminance block. When the motion estimation error of alpha block is negligible, the block is skipped.

3.2.Inter frame coding

Generally speaking, nter frame coding with macroblock based motion compensation is very efficient for luminance and chrominance signals. Therefore, we have implemented macroblock based inter frame coding for alpha plane.

When many alpha pixels have many gray values and therefore DCT based alpha coding scheme is used, we use conventional inter frame coding same as chrominance block. When alpha pixels have two typical values, frame differential pixel has three typical values. It sometimes increases the entropy of alpha plane than intra frame coding. On the other hand, coding efficiency of quadtree can be improved by skipping to encode sub-blocks of lower levels, depending on whether that sub-block value is uniform or not.

Therefore, when the compensation error of alpha block is negligible, we use motion compensation with no residual coding and skipped all index coding of quadtree. The motion vector is the same as that of luminance block. When the compensation error is unnegligible, the alpha block is intra frame coded with quadtree regardless of the coding mode of the luminance block (intra or inter).

4. Shape Adaptive Coding

Since both encoder and decoder share the boundary information from a locally decoded alpha plane, we can use shape adaptive scheme for coding Y, Cr and Cb blocks on a boundary. Some methods for shape adaptive coding were already proposed. Gilge[Gea89] proposed to rebuild orthogonal transformation for each block with its shape. That is, at first, orthogonal function (polynomial or DCT) set is orthogonalized invalid pixels, then the block is transformed with the orthogonalized set. Sikora[SM94] proposed to transform valid pixel's area with spatial and frequency shift. That is, at first, valid pixels are orthogonally transformed in vertical direction and reordered as vertical frequency, then the *i*-th vertical frequency components are orthogonally transformed in horizontal direction. These schemes need high computational complexity.

Considering the computational burden, we adopt value padding at invalid pixels (alpha = 0) and then we apply the conventional DCT transform to all blocks with non-zero alpha pixels. Since invalid pixels are discareded at decoder, we may replace the pixel with any value. To improve coding efficiency, we tried to pad pixel with mean values of Y, Cr and Cb, or with values that supress the high frequency components of DCT. Furthermore, we may adaptively choose an optimum padding scheme by using *a posteriori* bits estimator. That is, as shown in Fig. 8, more than one scheme is used for padding the invalid pixels of each block which is then orthogonally transformed. The bits estimator then estimates the coded bits of each block. The padding scheme which yields the least number of bits is selected. Although the encoder hardware become complex, it is benefitable for creating package media that does not need real time encoder. Decoder is simple and same as any padding methods.

When the padded pixels's values are calculated from a formula such as mean, quantized distortion per pixel can be reduced using correlation between pixel values at decoder. It contributes to improving the coding efficiency of Y, Cr and Cb blocks.

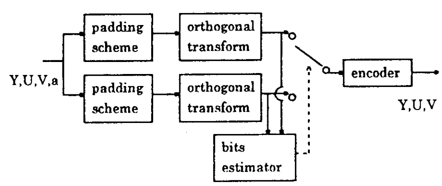


Fig. 8 Adaptive padding

5. Experimental Comparison

We evaluated several opacity coding schemes and shape adaptive transform schemes. We tested, 240-frame sequences of SIF (352×240 resolution) format called "bream", "children" and "bicycle". The opacity information of "bream" and "children" are extracted by chroma-keying process, while that of "bicycle" is made with computer graphics. Fig. 2 shows "bream" and Fig. 9 shows "children" and "bicycle" sequences.

5.1.Opacity Coding

We evaluated MH, MR, MMR[1], chain differential coding, DCT and quadtree scheme. When we evaluated loss-less binary data coding scheme, alpha plane was truncated to binary data. DCT and quadtree method are macroblock based coding, while the other schemes are frame based coding. Since DCT needs more bits than other coding scheme, we tested with subsampled alpha plane. Quantization step was set to produce nearly same picture quality as other binary coding scheme. Average bits to encode one alpha frame in intra frame coding is shown in Table 1.

Table 1 Result of opacity coding

	bream	children	bicycle
MH	11502bits	13226bits	9457bits
MR	8557bits	9521bits	7123bits
MMR	5747bits	3598bits	2946bits
chain	1515bits	1947bits	10813bits
quadtree	2159bits	2825bits	1506bits
DCT	9217bits	11208bits	3986bits
subsample DCT	3951bits	4999bits	1377bits

Quadtree scheme yields the least bits. Moreover, as it is a macroblock-based scheme, macroblock-based motion compensation can be easily applied. Subjective quality of binary based scheme is better than that of subsampled DCT. Subjective quality is summarized in Table 2.

[1] MH, MR, MMR are used in G4 facsimile.

Table 2 Subjective quality

	bream	children	bicycle
binary based	good	good	good
DCT	good	good	fair
subsample DCT	fair	fair	bad

For inter frame coding, we evaluated quadtree based scheme and DCT based scheme with subsampled alpha plane. Chain coding scheme is not examined because it cannot cope with macroblock based motion compensation. For inter frame coding, we used the proposed method described in section 2 and section 3. The results are shown in Table 3. The sequence is "bream" and we controled the bit rate to about 300Kbps.

Table 3 SNR comparison

	DCT	quadtree
SNR of Y	28.5dB	33.2dB
SNR of Cr	36.9dB	43.6dB
SNR of Cb	37.3dB	42.3dB
SNR of alpha	25.8dB	34.2dB

Both SNR and subjective quality are excelent in the quadtree scheme. Y, Cr and Cb blocks are coded with DCT while the alpha plane is coded with DCT and quadtree, respectively. Comparing to DCT-based method, quadtree yields less bits for alpha data, and therefore more bits are assigned to Y, Cr and Cb. The same effect was observed at the sequence "children".

5.2.Shape Adaptive Coding

We tested the adaptive-shape orthogonal transform methods proposed by Gilge for coding Y, Cr and Cb blocks. In Gilge's scheme, we tested polynomial and DCT-based orthogonal function. The average bits for encoding color frame in intra frame coding are summarized in Table 4.

Table 4 Result of shape adaptive coding

		bream	children	bicycle
Gilge (polynomial)	Y	102200bits	76932bits	24679bits
	Cr	9328bits	9189bits	689bits
	Cb	4693bits	13402bits	725bits
Gilge (DCT)	Y	102235bits	76920bits	24829bits
	Cr	9300bits	9007bits	670bits
	Cb	4642bits	13382bits	692bits
Sikora	Y	104569bits	79258bits	25755bits
	Cr	10635bits	10868bits	2427bits
	Cb	6926bits	15090bits	2472bits
padding	Y	102357bits	77999bits	25302bits
	Cr	9484bits	8923bits	588bits
	Cb	4537bits	13513bits	607bits

Since quantization steps of all schemes is were set as same SNR and subjective quality are nearly the same. In spite of high computational complexity of the Gilge's shape adaptive transform, we found that the coding efficiency gain is not notable in comparison with value padding. Sikora's scheme needs more bits for "bicycle". Therefore, we conclude that the padding scheme is more useful than shape adaptive coding.

5.3.Total Performance

The original sequence consists of 240 frames, three layers:CG rendered background, a real fish, and CG rendered captions. Fig. 10 shows each layer of the image sequence. The total bit rate attains about 250Kbps at 30 frames/sec for 240 frames. The details are reported as:

background CG layer:

1 template(480×240 pel), 66Kbit+affine global deformation,

intermediate real fish layer:

14 templates(1 I-picture + 13 P-pictures), total 544Kbit + general frames (reconstructed by only global and local deformation without residual coding), 1296Kbit (i.e., 5.4Kbit/frame), and

foreground caption layer:

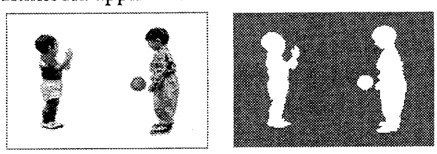
1 template 57Kbit + 2nd order global deformation.

To compare with the result of an MPEG1 coding result, we applied the residual coding to the intermediate fish layer. Although the total bit rate increases to 500 Kbps, its subjective quality is comparable to the result of MPEG1 at 1Mbps.

6.Conclusion

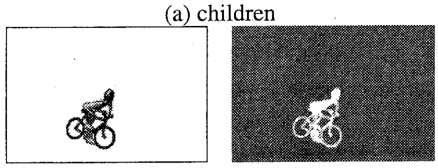
The advantages of the layered image coding can be summarized as follows: By treating regions for each objects separately, we may more often apply global motion compensation. And there is no necessity to encode covered and uncovered background regions. Those are the keys for higher coding efficiency. We prove that quadtree scheme is efficient for opacity coding produced by chroma keying. Furthermore, we proved that padding of invalid pixel contributes to high coding efficiency with less implementation complexity comparing shape adaptive transform methods. Therefore we conclude that quadtree based opacity representation and DCT with shape adaptive value padding is suitable for current DCT-based coding.

The challenge is to obtain 2.1D description automatically from a natural scene.[2] Nevertheless, we believe that starting with synthesized images is a good initiative for contribution to the multimedia applications.



color plane alpha plane

(a) children



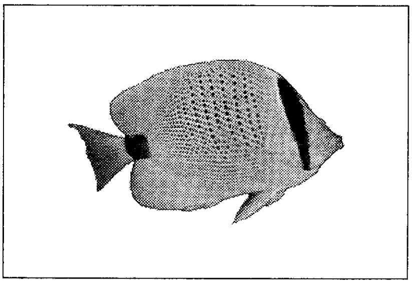
color plane alpha plane

(b) bicycle

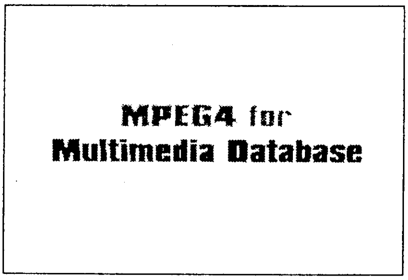
Fig. 9 Layer image sample

[2] Automatic error free segmentation is an extremely tough problem. You can develop the semi-automatic segmentation tools interactively manipulated by AV authors.

background layer



middle layer



top layer

Fig. 10 Layer components of sequence

7.Reference

[Adel94] J. Y. A. Wang and E. H. Adelson. "Representing Moving Image with Layers." IEEE trans. on Image Processing, Vol.3, No.5, pp.625-638, Sept. 1994.

[AKOK92] C. Auyeung, J. Kosmach, M. Orchard and T.Kalafatis, "Overlapped block motion compensation." In Proc. SPIE VSIP'92, pp.561-572, Nov. 1992.

[AOE94] AOE subgroup, "Proposal package description(PPD) revision 1.0." ISO/IEC JTC1/SC29/WG11 Document N821, Nov. 1994.

[Bea78] Paul R. Beaudet, "Rotationally invariant image operators." In Proc. 4th ICPR, pp. 579-583. IAPR, 1987.

[EBK95] M. Etoh, C. S. Boon and S. Kadono, "An image coding scheme using layered representation and multiple template." Technical report of IEICE, IE94-159,pp.99-106, March 1995.

[Eto94] M. Etoh, "Low bit rate image coding using wavelet transform and control grid interpolation." In Proc. '94 Spring IEICE Convention Vol. 7, pp.49, 1994. D-316.

[Gea89] Michael Gilge and et. al., "Coding of arbitrarily shaped image segments based on a generalized orthogonal transform." Signal Processing: Image Communication, Vol. 1, pp.153-180, 1989.

[JTC93] ISO/IEC JTC1/SC29, "Information technology-generic coding of moving pictures and associated audio." ISO/IEC JTC1/SC29 WG11/602, Nov. 1993. Committee Draft ISO/IEC 13818-2.

[LG91] D. Le Gall, "MPEG:a video compression standard for multimedia applications." Trans. ACM, April 1991.

[NM90] M. Nitzberg and D. Mumford. "The 2.1-D sketch." In Proc. ICCV'90, pp.138-144, Osaka, Nov. 1990. IEEE.

[SM94] T. Sikora and B. Makai, "Shape-Adaptive DCT for Generic Coding of Video" ISO/IEC MPEG, Julay 1994.