Object-based Coding for Future Broadcasting

Yoshiaki Shishikui, Yutaka Kaneko, Shinichi Sakaida,
Wantao Zheng and Yuji Nojiri

NHK Science and Technical Research Laboratories
1-10-11, Kinuta, Setagaya-ku, Tokyo 157 Japan
phone +81-3-5494-2231 fax +81-3-5494-2237
{sisikui, kaneko, sakaida, zheng, nojiri}@strl.nhk.or.jp

Abstract: This paper describes the concept of object-based coding for future broadcasting environments. Digital broadcasting uses the MPEG2 coding scheme which is regarded as a picture-based coding. An object-based coding scheme is a potential candidate for future broadcasting both for studio and distribution uses, and it offers a higher compression and more flexible content handling. This paper also describes key technologies that we have been developing for the object-based coding, e.g., image analysis, object extraction and coding of objects.

1. Introduction

Video coding has become popular in broadcasting environments, although once it was regarded as irrelevant for high quality applications such as broadcasting. Video coding was introduced into transmissions for contribution purposes, digital VCRs and non-linear editors. Furthermore, digital broadcasting using video coding has now come to the reality.

Several kinds of coding techniques are used for these applications. Among them, the MPEG-2 coding scheme is the most important, supported by a world-wide consensus and rapid development of the conformed LSIs. The performance of the MPEG-2 coding scheme could be improved further by the optimum selection of coding parameters, such as motion vectors, bit allocations and GOP structures. However, there are limitations in functionality and coding efficiency as far as the current framework of coding scheme is used.

Conventionally, image compression coding techniques have been applied to a whole image or sequence in the same manner, assuming static image characteristics for the whole image or sequence. Deviations from this assumption lead to deterioration of coding efficiency. With respect to functionality, it is basically impossible to edit or process the picture content in the coded domain.

Recently, a new type of coding scheme called object-based coding is attracting much interest. Unlike the current coding scheme which is regarded as a picture-based coding, in object-based coding, image processing is applied to each object, such as a person, or the background within an image. Object-based coding is a key candidate for next-generation coding for broadcasting environments to overcome the limitations of the current coding.

In this paper, our concept of object-based coding is described. The required technologies for object-based coding are summarized, and the present status of development of these techniques is outlined.

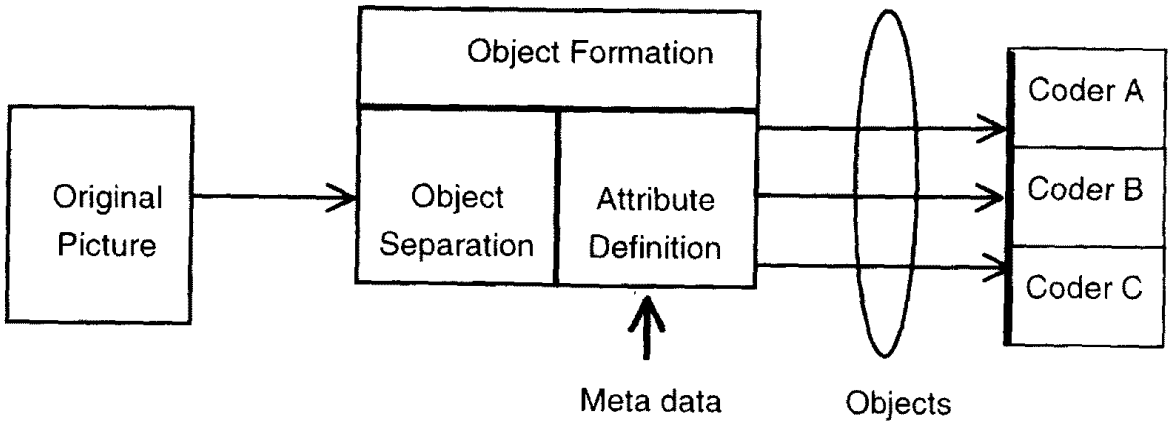


Figure 1 Object-based Coding.

2. Object-based coding

2.1 Object-based coding in broadcasting application

The object-based coding method produces data for each object and sends them to users for free editing of pictures. The users can enlarge or replace any objects they choose. This functionality is useful for flexible and efficient program production. This functionality might be provided to viewers to some extent, thus expanding the scope of program enjoyment at home.

In picture coding, statistical characteristics are used to compress the data of the picture. However, the characteristics are different for objects. Current coding technology produces compressed data for each scene which is composed of multiple objects, whereas each object is compressed by this object-based coding, thus allowing more efficient picture compression by utilizing the proper compression method for the objects.

As mentioned above, object-based coding offers higher compression and flexible content handling, which are required for both distribution and studio uses.

2.2 Object-based processing in broadcasting environment

An "object-based" approach is not necessarily new in broadcasting. Currently, some television programs are produced in an object-based way, meaning that objects are combined to create a certain scene. In this case, objects are people extracted by chromakey technology, and images are produced by computer. Object-based coding is regarded as an expansion or evolution of the idea for studio application.

For object-based processing, more information is required than just that of the image. In the studio, it is not so difficult to obtain image-related information as meta-data.

Hence, object-based coding is not only necessary but also suitable for the broadcasting environment.

2.3 Technologies of object-based coding

A flowchart of object-based coding is shown in Figure 1.

If the picture is not produced in an object-based way, we must first separate it into component objects. A primitive level of the object separation is region segmentation.

At the same time, we have to define the attributes of the objects, which include shape, texture, motion, position, etc. Some of them are given, but we need to extract the others from the

input image sequence and available meta data using image analysis techniques. The extracted, or estimated, attributes can be used for the object separation also.

The objects are then coded and transmitted. With object-based coding, objects take different shapes depending on the contents of the scene. We need image compression methods suitable for these objects.

3. Development of key techniques

As mentioned in Section 2.2, object-based coding requires various techniques. In this section, we outline the current status of our research on object-based coding.

3.1 Extraction of objects

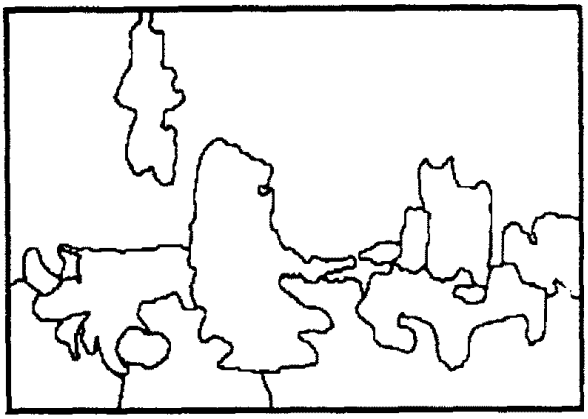
We have developed an image segmentation technique based on the K-means algorithm[1][2]. Clustering based on the K-means algorithm is widely used, but this tends to segment the image into too many small regions. Furthermore, since the shape of initial clusters has a lasting influence, the segmentation sometimes takes place in areas other than the original true contour of an object in the image .

To overcome these problems, we propose a 3-step region integration method. In our method, we first conduct the K-means algorithm using the characteristic values, three color signal levels, two position parameter coordinates and two motion information parameters. Then we apply the following integration methods:

1. Majority filtering
2. Integration of small isolated regions
3. Integration of several region segmentation results.

Figures 2(a) and (b) show the original picture and the result of region segmentation by our method. We also applied our method to several pictures and the results showed that we can extract the true contours of regions from many kinds of images even with the same parameters. In comparison with other methods, our method provides better results with less computational complexity.

(a) The "girl" image



(b) Segmentation result

Figure 2 Segmentation with region integration.

3.2 Motion detection

Among attributes of the object, the motion parameters are very important. They are used for segmentation of objects, as used our method in Section 3.1, and also for efficient coding of the object.

We have studied two motion estimation techniques.

3.2.1 Gradient method with eigen-value algorithm

One is based on the gradient method[3][4], and the other is based on the block-matching method[5]. Gradient-based motion estimation methods have a feature that there is no restriction on the estimation precision. Hitherto, a least-squares algorithm was widely used to derive the motion vectors from the gradients. However, the gradient method with the least-squares algorithm tends to increase estimation errors when there are large movements. We found that another method, an eigenvalue algorithm, has the capacity to estimate larger movements as shown in Figure 3.

Along with the motion, we developed a way to derive reliability indexes of the estimated motion by analyzing the estimation procedure, which are used for the optimal selection of the block size. We used the indexes to improve the accuracy of motion estimation. The pixels with low accuracy estimation are classified by reliability indexes, and then the size of the estimation block is changed. The size modification is repeated until the reliability indexes indicate high accuracy. We confirmed the effectiveness of this scheme through simulations. Figures 4 (a) and (b) show an original picture and final block size for each pixel.

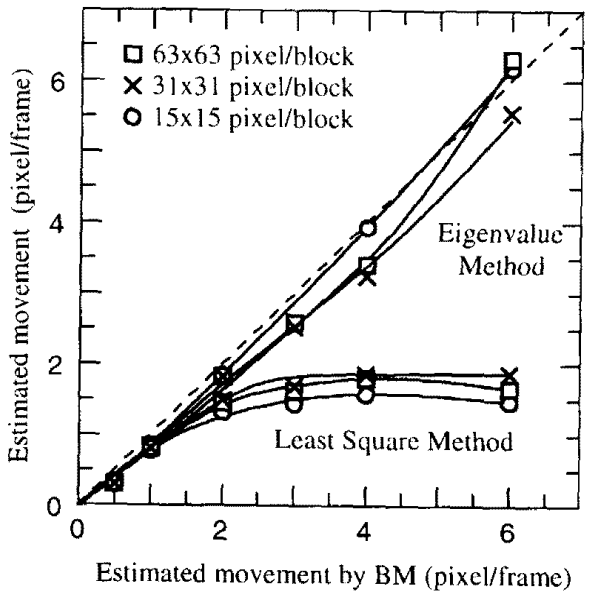


Figure 3 Comaprison of least square method and eigenvalue method.

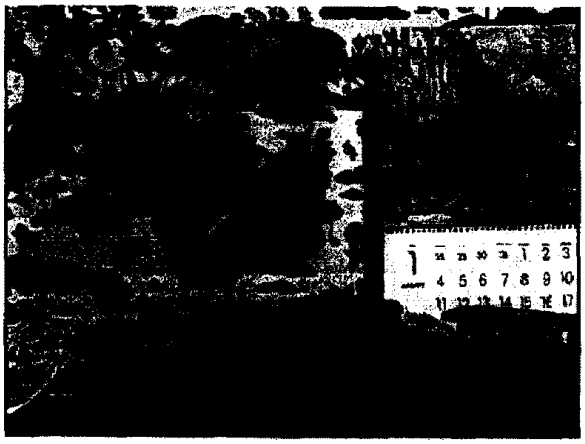
(a) The "Mobile & Calendar" sequence

(b) Block size used for estimation

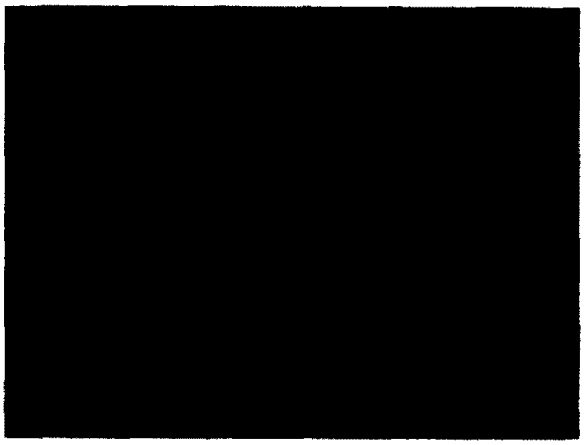
Figure 4 Eignvalue method with variable block size.

3.2.2 Block matching method with spatio-temporal block

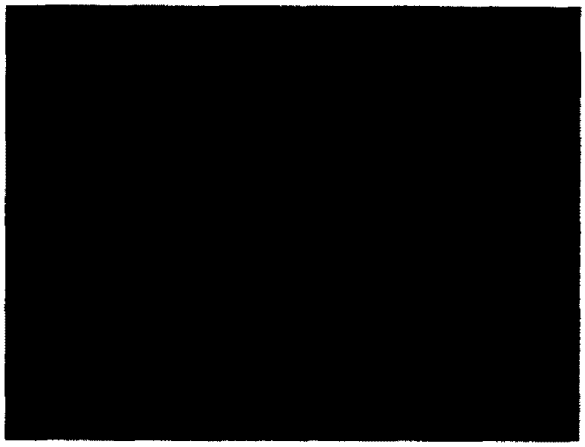
For the block matching method, we tried a new approach, that is, a spatio-temporal block matching algorithm. In principle, the relation between the reliability of estimated motion and the resolution of the motion field is antinomy, with respect to the block size of estimation. To avoid the antinomy, we introduced a spatio-temporal three-dimensional block. Figures 5 (a), (b) and (c) show an original picture, estimated motion using



(a) One frame of "Mobile & calendar"



(b) Results with conventional spatial blocks



(c) Results with spatio-temporal blocks

Figure 5 Spatio-temporal block Matching.

one frame blocks (conventional), and estimated motion using nine frame blocks, respectively. These indicate that the proposed method provides accurate estimation, keeping a high resolution motion field.

3.3 Object Coding

Objects take different shapes, so a texture coding for arbitrarily shaped objects is required. For this purpose, we have developed "Region Support DCT (RS-DCT)"[6][7]. RS-DCT is based on the conventional two-dimensional DCT. The object is divided into rectangular blocks, and blocks inside the object are coded with the 2D-DCT, while boundary blocks are coded with RS-DCT. The RS-DCT tool is used to derive the best 2D-DCT compatible coefficient set which represents pixel values of the target object within the block. To derive the coefficient set, the support region of the analysis vector bases is modified to fit the shape of the target object. The ortho-nomality of the bases is lost by the

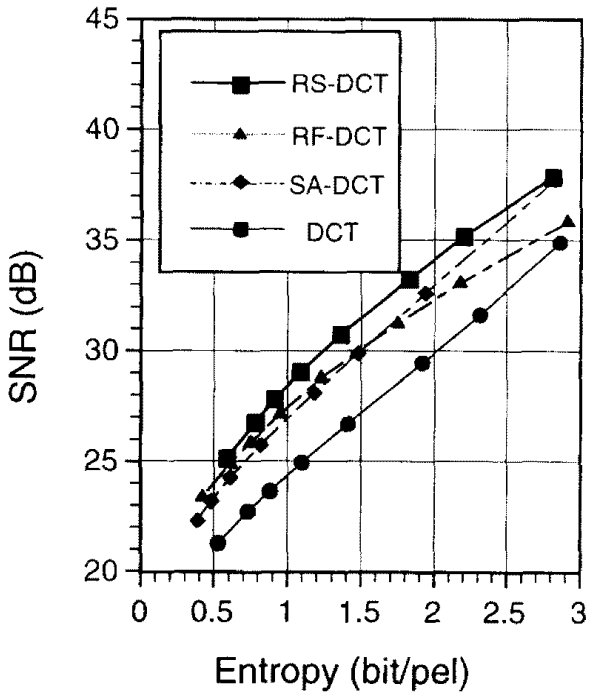


Figure 6 Comparison of coding performance of RS-DCT with other methods.

modification, hence we apply norm compensation on the bases, and derive coefficients based on a successive approximation procedure using the norm compensated bases.

Figure 6 shows a comparison of coding performances of coding methods for arbitrarily shaped textures[8][9]. RS-DCT provides the best performance among them.

4. Conclusion

Object-based coding in the broadcasting environment was described, and the key techniques of the object-based coding and examples of developed techniques were introduced. We are still in the early stages of development, and many challenges remain with respect to clarification of the requirements for applications and establishment of techniques.

On the other hand, MPEG-4 covers a similar field, in which a total system for object-based coding is being developed. This provides a good framework for our purpose. However, some of the key techniques such as object separation and attribute extraction are outside of the scope of the standard. The detailed design for specific applications is also outside of the scope.

We shall develop a system for broadcasting use utilizing the MPEG-4 and our own technology.

[References]

[1] S. Sakaida, Y. Kaneko, Y. Shishikui, Y. Tanaka and I. Yuyama, " Image Segmentation by Integration Approach Using Initial Dependence of K-Means Algorithm", Proc. PCS'97, O6.1, pp.265-270, Sept. 10-12, 1997.

[2] S. Sakaida, Y. Shishikui, Y. Tanaka and I. Yuyama, " An Image Segmentation Method by the Region Integration Using the Initial Dependence of K-Means Algorithm", IEICE Trans. vol.J81-D-II, no. 2, Feb. 1998(to be published).

[3] Y. Kaneko, Y. Shishikui,Y. Tanaka and F. Okano, " Improvement of Motion Estimation Accuracy Using Gradient Vector Distribution in Advanced Picture Coding", IEEE Trans. Broadcasting, vol. 42, no.3, pp.237-244, Sept. 1996.

[4] Y. Kaneko, Y. Shishikui and Y. Tanaka, " A Study of the Suitable Block Size in Motion Estimation using Gradient Vector Distribution", Proc. PCSJ'96, P-5.23, pp.115-116,Oct. 7-9, 1996.

[5] W. Zheng, Y. Shishikui, Y. Tanaka and I. Yuyama, "Pixel-wise Motion Estimation with Spatiotemporal Block Matching", IEICE Tech. Report, IE97-17, pp.61-68, May 23, 1997.

[6] Y. Shishikui, " Arbitrary Shape Texture Coding Using Region Support DCT", IEICE Trans. vol.J80-D-II, no.7, 1810-1817, July 1997.

[7] Y. Shishikui, Y. Nojiri and Y. Tanaka, " Improvement of Region Support DCT Coding Scheme", Proc. PCSJ'97, P-2.8, pp.55-56,Oct. 6-8, 1997.

[8] T. Sikora and B. Makai, "Shape-adaptive DCT for Generic Encoding of Video," IEEE Trans. CSVT, vol. 5, no.1, 1995

[9] N. Itoh, H. Katata and H. Kusao, "A method of DCT for Arbitrarily Shape", Proc. PCSJ'95, 5-2, pp.77-78, 1995