# Object-based Video Coding for Arbitrary Shape by Visual Saliency and Temporal Correlation

Kazuya Ogasawara, Tomo Miyazaki, Yoshihiro Sugaya, and Shinichiro Omachi
Graduate School of Engineering, Tohoku University
Sendai, Japan
email:{oga203, tomo, sugaya}@iic.ecei.tohoku.ac.jp, machi@ecei.tohoku.ac.jp

***Abstract*—This paper addresses a problem of object-based video coding. We propose a video coding method for arbitrary shapes of objects. The proposed method extracts objects on the basis of visual saliency and temporal correlation between frames. Subsequently, we compress the video by changing coding quality for the extracted objects and background regions. The experimental results show that the proposed method can reduce bit rate while preserving target object quality.**

***Keywords-arbitrary shape; visual saliency; object-based coding.***

## I. INTRODUCTION

Object-based video coding is effective in many applications, such as video conference and surveillance where we only need regions of interest and it is unnecessary to transmit the entire video. For that reason, object-based coding schemes have been investigated actively. MPEG-4 is a video coding standard for arbitrarily shaped objects. However, this was standardized in 1999, so there is possibility not to preserve important information because compression efficiency of MPEG-4 is lower than current mainstream coding standards. Lan et al. proposed an object-based coding scheme that determines foreground using a depth map and incorporates technologies of MPEG-4 into HEVC [1]. However, it requires a special video camera to capture the depth map. For videos captured by a stationary camera, there are some researches for video surveillance [2] and video conference [3]. In [4], Ng et al. proposed an object-based coding system using multiple video cameras for dynamic image-based representations. However, all of the above methods focus on videos obtained under certain conditions. Many coding schemes have been researched for the purpose of preservation of quality in the area where human tends to perceive. In [5], the foreground is determined on the basis of perception characteristics of human who pay attention on moving objects. In [6], the facial region is set as the foreground, and face parts (e.g., eyes, nose and mouth) are compressed in high quality. However, in these coding systems, the target object is restricted. There are researches which use saliency to determine important regions in videos. The coding method which changes the quality parameter by the macroblock based on saliency is investigated in [7]. However, the method [7] directly uses responses of saliency to compress videos, it does not separate foreground and background. Hence the method doesn’t take account of the shape and the contour of the objects.

The contribution of this paper is as follows. We focus on saliency and temporal correlation to extract objects in video. In addition, we propose an arbitrary shape object-based coding scheme which varies the coding quality depending on the foreground and the background. Then, we demonstrate the effectiveness of the proposed method by experimental results.

The outline of this paper is given as follows. In Section II, overview of the proposed method is explained. In Section III, we describe the proposed object extraction method in detail. Afterwards, Section IV proposes an object-based coding system. Finally, Section V shows some experimental results and Section VI concludes this paper.

## II. OVERVIEW OF THE PROPOSED CODING SYSTEM

The block diagram of the proposed coding system is shown in Figure 1. The encoder extracts visually attractive objects automatically from the input video. Then, we create the foreground video in which background pixel values are equal to 0 and the mask video which consists of mask images. By using a standard video coding method, we encodes these two videos at high quality, and also encode the input video at low quality in order to use it as the background when decoding.

In the decoder, three videos are decoded. Then, the region of foreground is extracted from the high quality object video using the mask. Similarly, the region of background is extracted from the low quality entire video using the mask and the synthesis video is reproduced by combining the foreground and the background.

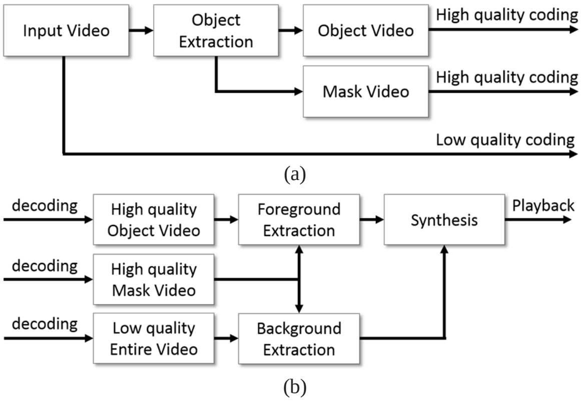


Figure 1. Block diagram of the proposed coding system: (a)Encoder (b) Decoder

## III. Visually Attractive Object Extraction

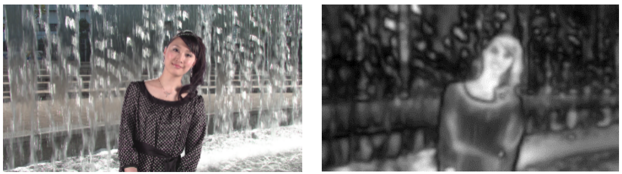
This process extracts visually attractive objects automatically. From the input video frames, key frames are chosen at a constant interval. In each key frame, we create a mask which shows the target object area using a saliency map [8] and GrabCut [9]. In other frames, we create it by moving the mask from the key frame to subsequent frames using optical flow. It reduces processing time and keeps the shape of the object among frames not to create masks for each frame from scratch. Furthermore, in order to improve the accuracy of object extraction for key frames, we compare the created mask with a predicted mask based on the mask of the preceding key frame, and create a mask again in a different setting if the difference is large.

### A. Key Frame Mask Creation

*1) Pre-Processing:* Pre-processing is performed to improve the accuracy of mask creation. First, the size of input frames is reduced to make the subsequent process easier. Second, smoothing with Gaussian filter is performed to reduce the high-frequency components which have bad influence on segmentation.

*2) Mask Creation:* This process estimates the target object area and creates a mask of the objects. We use the saliency map proposed in [8] to estimate it. The saliency map shows the degree of visual attention. In the saliency map, large values represent locations where human will pay attention. Examples of an input frame and its saliency map are shown in Figure 2.

To create a highly precise mask, we use the GrabCut algorithm. The GrabCut is a graph-based two-class segmentation method. In the algorithm, the input image is expressed as a weighted graph based on similarity to samples of foreground and background given by the user and color difference of adjacent pixels. Then, segmentation is conducted by finding the minimum cut to divide the graph into two subgraphs. The saliency map does not always have uniform saliency in the same object, and there are large saliency pixels in background. Therefore, it is difficult to create a mask using the saliency map only. The proposed method realizes to create a highly precise mask by combining the saliency map and the GrabCut. In order to assign sample pixels, a label mask is created. The label mask consists of four values: BGD (to be a background sample pixel), FGD (to be a foreground sample pixel), PR_BGD (to be probably background pixel), and PR_FGD (to be probably foreground pixel). The PR_BGD and PR_FGD pixels are estimated by calculation based on the FGD and BGD pixels. We create the label mask automatically based on the saliency map. First, we hypothesize that humans tend to pay attention to the center of the screen rather than the edge of the screen. Based on this hypothesis, 15 pixels from the border of the image are set as BGD pixels regardless of its saliency. The other pixels in the inside of the screen are set as some label based on the saliency map. Specifically, the high saliency pixel becomes FGD pixel, the middle saliency pixel becomes PR_FGD pixel, and low saliency pixel becomes PR_BGD pixel. Figure 3 shows the label mask. Then, the GrabCut using the label mask creates a mask which shows the area of the visually attractive object in the frame.



(a) (b)

Figure 2. A saliency map: (a) Input frame “Fountain(Chromakey)” (b) Saliency map

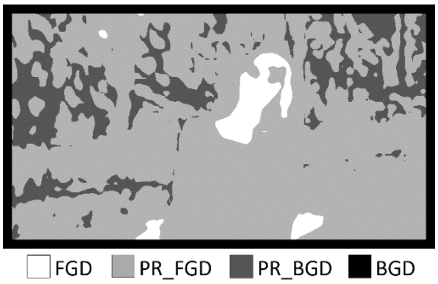


Figure 3. The label mask created based on the saliency map

*3) Post-Processing:* Two types of post-processings correct misjudged pixels. In the mask created by using the GrabCut, some background pixels may be judged as foreground by mistake. Therefore, we do labeling process to the mask and judge small foreground labels as background, the size of which is smaller than 0.5% of the frame size. Likewise, some foreground pixels may be judged as background by mistake. Therefore, we change the background pixels surrounded by foreground pixels to foreground. This process fills holes of the foreground and improves the mask.

*4) Mask evaluation considering temporal correlation:* For more accurate mask creation, we do comparative evaluation of the mask that takes advantage of temporal correlation, which is a characteristic of video. Video is agregation of continuously captured images, so it is rare that the contents vary greatly between frames. Therefore, it is not desirable that the shape of the mask varies greatly between frames. However, there is a possibility that the shape of the mask varies greatly when doing object extraction using information about the target frame only.

We compare the created mask and the predicted mask, which is predicted from the mask created in the previous key frame. We use the optical flow to create the predicted mask. First, we calculate the optical flow between continuous key frames based on the method of Gunnar Farneback [10]. Second, we do the labeling process to the mask, and calculate the average vector of the optical flow of each label in the foreground area. Then, we create the predicted mask by moving the foreground area for the average vector of each label. The number of pixels where the predicted mask does not overlap with the mask created using the GrabCut is found by taking exclusive OR of them. If the number becomes more than 10% of the number of all pixels in the frame, we conduct the following process for the mismatched pixels. When the pixel of the predicted mask is the background pixel, we set the pixel of the label mask as PR_BGD. Similarly, when the pixel of the predicted mask is the foreground pixel, the pixel of the label mask is set as PR_FGD. Then, we conduct the GrabCut algorithm again using the renewed label mask.

### B. *Mask Creation in the Non-Key Frame*

For the non-key frame, the mask is created by moving the previous mask created already using optical flow. This process is generally the same as creating the predicted mask for comparative evaluation, but uses the optical flow between continuous frames. It reduces the shape change of the foreground to move the mask based on the optical flow.

### C. *Object Extraction*

In this process, we enlarge the masks to the size of the input frame and extract the foreground from the input frame based on the mask. Then, the foreground video, of which pixel values in the background are 0, is created. Figure 4 shows the foreground frame. We create the mask video by aggregating the masks. The mask video is utilized to determine the background at the time of decoding.

## IV. Codec System

### A. *Encoder*

In the proposed method, the encoder realizes preserving important information and reducing unnecessary information by changing the coding quality in H.264 according to the type of video. The foreground video is coded in high quality, so that information of the target object is preserved. Because the value of all pixels of the background area is zero, it is possible to reduce the data size with high quality. In addition, the mask video used in decoding is coded in high quality, so that boundary between the foreground and the background is preserved. Furthermore, the input video which is the subject of object extraction is coded in low quality, and the data size is reduced significantly. This entire coded video is used as the background at the time of decoding.

### B. *Decoder*

This process decodes three kinds of coded videos based on H.264 at first. Next, we extract the foreground area from the object video coded in high quality based on the mask. Similarly, we extract the background area from the entire video coded in low quality based on the mask. Then, we synthesize the foreground and the background. Eventually, we can obtain the synthesis video which preserve the quality of the foreground and reduce the quality of the background because the coding quality is different in the foreground and the background. Since it is not the block based quality control, we can handle arbitrary shape objects. Figure 5 shows the synthesis frame.

## V. Experiments

In order to confirm the effectiveness of the proposed method, we conducted a comparative experiment with H.264. We used three video sequences at FHD resolution (1920×1080): “Fountain(chromakey)”, “Fountain(dolly)” and “Truck Train” included in the ITE/ARIB Hi-Vision Test Sequence 2nd Edition [11], shown in Figure 1 (a) and Figure 6. In the proposed method, we set the key frames at intervals of three frames, and the same low-quality entire video is used for background regardless of the bit rate.

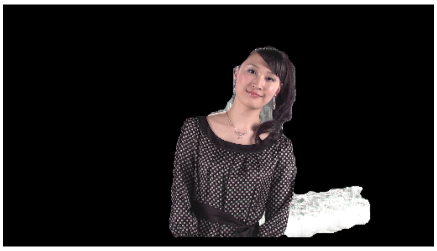


Figure 4. The foreground frame

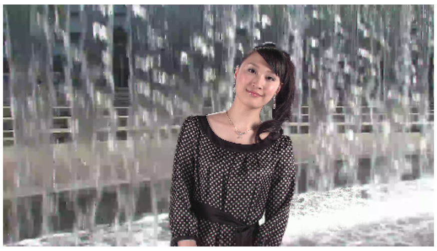


Figure 5. The synthesis frame

### A. *Rate-Distortion Performance Evaluation*

In this subsection, we evaluated the coding performance by rate-distortion curves, which indicates a relationship between the bit rate and PSNR. Figure 7 shows the rate-distortion curve in whole and foreground regions.

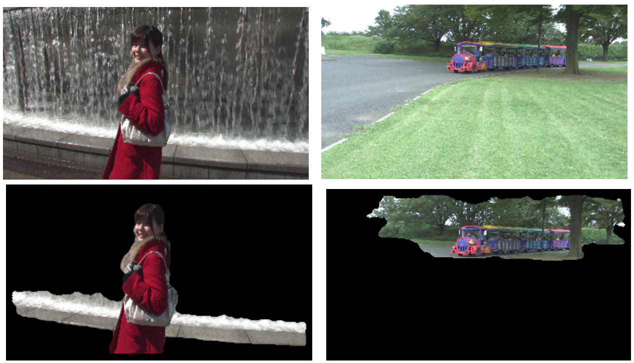
In the proposed method, the image quality of the foreground was good, but that of the background was significantly degraded. Therefore, PSNR on the entire area became low values. In contrast, the video coding standards, such as H.264 utilize rate-distortion optimization to choose the partition manner and the coding mode, so PSNR is optimized. For that reason, PSNR of H.264 became larger than that of the proposed method.

On the other hand, we confirmed that PSNR on the foreground of the proposed method became large than that of the H.264 at various bit rates. This is because that the proposed method reduces the bits about the background significantly. This result shows that the proposed method can reduce more bit rate than H.264 when the image quality of the foreground is compressed to the same degree between the proposed method and H.264. However, as for “Truck Train”, the PSNR on the foreground of the proposed method became lower than that of H.264 at a low bit rate. The background of “Truck Train” is not complicated. Therefore, the proposed method could not have enough effects. In addition, inaccuracy of object extraction affected the results.

### B. *Subjective Quality Evaluation*

In this subsection, we evaluate subjective quality of the foreground by measuring mean opinion scores (MOS). In this experiment, all the video sequences compressed by the proposed method and H.264 at different bit rates were displayed in a random order, MOS is ranging 1 to 10. We had 13 participants involved in this experiment.

Figure 8 shows the MOS values. As for “Fountain



(a) (b)

Figure 6. Top row: Original video sequences Bottom row: Object extraction results (a)”Fountain(dolly)” (b)”Truck Train”

(chromakey)” and “Fountain(dolly)”, the MOS values of the proposed method are larger than H.264 at any bit rate. However, as for “Truck Train”, the MOS value of the proposed method is smaller especially at a low bit rate. This result could be due to boundary flicker caused by inaccuracy of synthesis of foreground and background.

## VI. Conclusion

In this paper, we proposed an object-based coding system that extracts visually attractive object with arbitrary shape and preserves the information of the object. The region of interest is estimated with the saliency map, and the objects are extracted using the GrabCut.

The experimental results show that the proposed method realizes both preserving the image quality of the target objects and reducing the bit rate. However, there is a possibility of reducing important information greatly if objects are not extracted correctly. Therefore, it is necessary to investigate a more accurate object extraction technique. In addition, it is very complicated to hold the foreground video, the mask video and the entire video for background. This reason is that we use the video coding standard, which is not object-based. Therefore, it needs to investigate an object-based encoder.

## Acknowledgment

This work was partially supported by JSPS KAKENHI Grand Number 26540077.

## References

[1] C. Lan, J. Xu, and F. Wu, “Object-based coding for Kinect depth and color videos,” Proceedings of the IEEE Conference on Visual Communications and Image Processing, San Diego, 2012, pp. 1-6.

[2] R. V. Babu and A. Makur, ”Object-based Surveillance Video Compression using Foreground Motion Compensation,” Proc. ICARCV, Singapore, 2006, pp. 1-6.

[3] Y. Li, X. Tao, and J. Lu, ”Hybrid model-and-object-based real-time conversational video coding,” Signal Processing: Image Communication, vol.35, 2015, pp. 9-19.

[4] K. T. Ng, Q. Wu, S. C. Chan, and H. Y. Shum, ”Object Based Coding for Plenoptic Videos,” IEEE Trans. Circuits and Syst. Video Technol, vol.20, no.4, 2010, pp. 548-562.

[5] M. Bosch, F. Zhu, and E. J. Delp, ”Video coding using motion classification,” Proceedings of the IEEE International Conference on Image Processing, San Diego, CA, 2008, pp. 1588-1591.

[6] M. Xu, X. Deng, S. Li, and Z. Wang, ”Region-of-Interest Based Conversational HEVC Coding with Hierarchical Perception Model of Face,” IEEE Journal of Selected Topics in Signal Processing, vol.8, 2014, pp. 475-489.

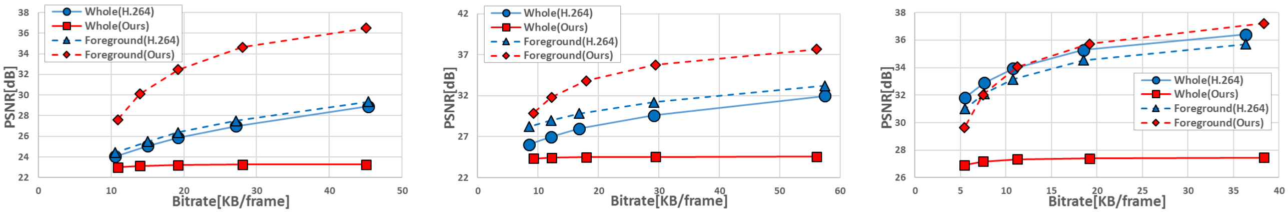
[7] H. Hadizadeh and I. V. Bajic, ”Saliency-Aware Video Compression,” IEEE Trans. on Image Processing, vol.23, Issue.1, 2014, pp. 19-33.

[8] L. Itti, C. Koch, and E. Niebur, “A Model of Saliency-based Visual Attention for Ra-pid Scene Analysis,” IEEE Trans. on PAMI, vol. 20, no.11, 1998, pp. 1254-1259.

[9] C. Rother, V. Kolmogorov, and A. Blake, “GrabCut: Interactive Foreground Extra-ction using Iterated Graph Cuts,” Proc. on ACM SIGGRAPH, 2004, pp. 309-314.

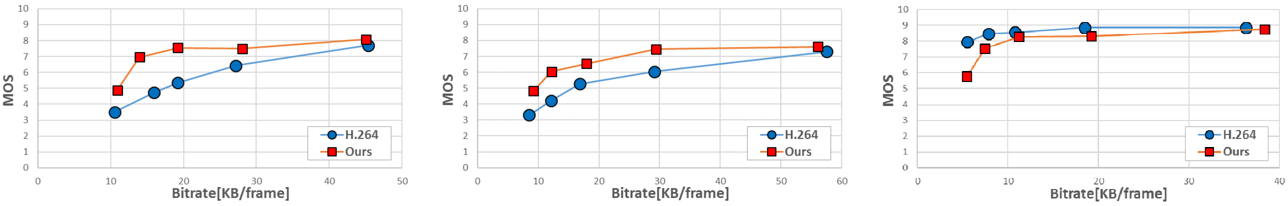
[10] G. Farneback, ”Two-Frame Motion Estimation Based on Polynomial Expansion,” Proc. of 13th Scandinavian Conference on Image Analysis, SCIA, 2003, pp.363-370.

[11] ITE/ARIB Hi-Vision Test Sequence 2nd Edition Reference Manual, 2009.



(a) (b) (c)

Figure 7. Rate-distortion curve: (a) “Fountain(chromakey)” (b) “Fountain(dolly)” (c) “Truck Train”



(a) (b) (c)

Figure 8. MOS value: (a) “Fountain(chromakey)” (b) “Fountain(dolly)” (c) “Truck Train”