Using a Region-Based Blurring Method and Bits Reallocation to Enhance Quality on Face Region in Very Low Bitrate Video

Chang-Hong Chen, Liang-Gee Chen, Hao-Chieh Chang
Department of Electrical Engineering, National Taiwan University,
No.1 Roosevelt Road, Sec.4, Taipei, Taiwan

Abstract—A region-based blurring algorithm to reduce bitrate in very low bitrate video coding is proposed. This algorithm reduces the bitrate by passing the original background image through a filter before motion estimation. Each frame in original sequence will be segmented into foreground and background at first. Background part, which is of less importance for human vision, will be blurred through a blurring filter, while the foreground is kept unchanged. The bitrate for head-and-shoulder type sequences can achieve 5%~20% bitrate saving by this blurring algorithm. According to the human vision model, PSNR degradation in the background is not evident. Using this blurring algorithm, but kept coded bitstream the same bitrate with the original sequence, about 0.5~1dB gain in foreground can be achieved due to bits reallocation between foreground and background.

1. INTRODUCTION

In current trend, H.263[1] is the most prevailing very low bitrate video coding standard. H.263 is based on motion-compensated block DCT, which can provide good quality with lower complexity, however, the performance is still limited above some range. It is almost true that all researches are trying to improve performance to code the video data. Object-based motion-compensation approach seem to provide a way. Object-based scheme introduces the concept of "object". An object is a subset of pixels in the picture. Pixels belonging to the same object have some similar property thus can be coded in more efficient way than pixel based method. To identify an object in a picture, segmentation in advance is needed, which will be the overhead of object-based coding.

Conventional coding standards, including H.261, MPEG1, MPEG2 and H.263, are all pixel-based codec(we treat block-based as a special case of pixel-based). To provide higher compression ratio, new generation of video coding, such as MPEG4, thus includes the ongoing concept of object-based coding. As a result, object-based coding becomes the focus of new coding trend.

Based on the concept of object-based coding, we present a region-based blurring algorithm to reduce the bitrate in very low bitrate coding. The algorithm will be described in section II. In section III, a segmentation method needed in the blurring algorithm will be presented. In section IV, considerations in blurring will be explained.

2. REGION-BASED BLURRING

The basic concept of the region-based blurring is: if we blur the original image to remove some high frequency component, bitrate saving can be achieved. However, blurring process will certainly degrade image quality, so the image will be segmented into foreground object and background at first. Only background image is blurred in order to keep the quality of the foreground object. The flowchart of the algorithm is in Fig. 1. Each original image is segmented into foreground and background in advance. The foreground is motion-estimated as before, while the background will be blurred at first and motion estimated by some fast search algorithm. Motion estimated result in both foreground and background is then transmitted to DCT and quantization stage that is the same as before. There are two issues, segmentation and blurring, to discuss in the algorithm. They are explained in the next two sections.

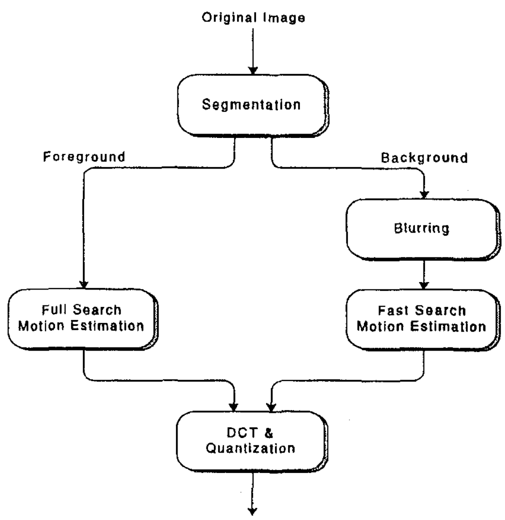


Fig. 1. The region-based blurring algorithm.

3. SEGMENTATION METHOD

Segmentation for very low bitrate video coding is ever presented[3]. For segmentation of successive images, the information we can use is color and motion. To speed up,

0-7803-4455-3/98/$10.00 © 1998 IEEE

we use only luminance and FD(frame difference). Luminance contains partial information of color, while FD contains partial information of motion. In our algorithm, FD is used only once to get the first reference rectangle. After the first reference rectangle is got, succeeding reference rectangles will be found through luminance.

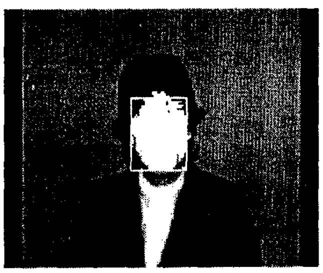
The way to find the first reference rectangle by FD is as follows. In step 1, current frame is subtracted from previous frame to get FD and get the integral projections on FD in horizontal and vertical directions. In the projection histogram, a segment with most non-zero values in horizontal projection is viewed as the horizontal range of the reference rectangle, and the same in vertical projection, which is step 2.

In vertical projection, however, the bottom position of the reference rectangle may be disturbed due to motion of shoulder. To avoid this problem, a scheme similar to [5] is adopted, the bottom position of the reference rectangle is determined according to the left, right and top positions of the reference rectangle:

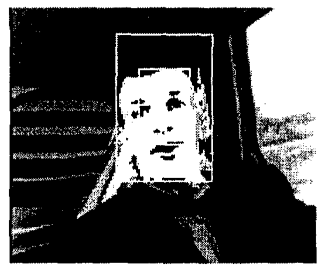
$$BottomPos = 1.5 \times (RightPos - Left\_Pos) + TopPos$$

After step 1 and 2, the face region is roughly known. To find the exact face region, region-growing method is adopted. First, finding a "seed point" as the start point to grow in advance. All points neighboring the start point with difference less than some threshold will be taken as region candidates. Each candidate point will be treated as a start point in next iteration and the process will stop after no candidate points exist.

Here we explain how to decide the seed points. At first, a rectangle smaller than the face rectangle shall be found, and construct a histogram on this small rectangle. Extract the pixel value which appears the most times in the histogram. All pixels with values within some predetermined range from the most frequent pixel are treated as seed points. The reason we construct a histogram on a smaller rectangle is that the face rectangle may contain hairs and some background. We want a rectangle containing less noise so we shrink it. In our trial, the small rectangle is shrunk to 1/2 both in width and height.



(a)Claire



(b)Carphone

Fig. 2. Result of region grow for Claire and Carphone.

$$\frac{1}{25}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

(a) 5 × 5 full

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

(b) 3 × 3 full

$$\frac{1}{5}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

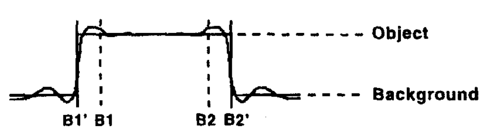
(c) 3 × 3 cross

$$\frac{1}{8}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

(d) 3 × 3 weighted cross

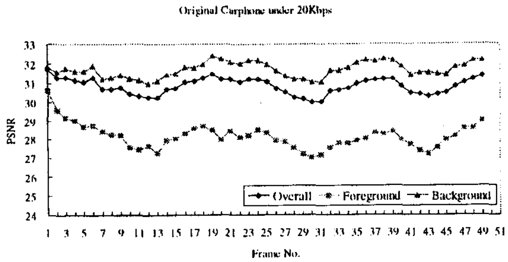
Fig. 3. Different blurring filters. (a) 5x5 full blurring filter. (b) 3x3 full filter. (c) 3x3 cross filter. (d) 3x3 weighted cross filter.

(a)Original blurring boundary.

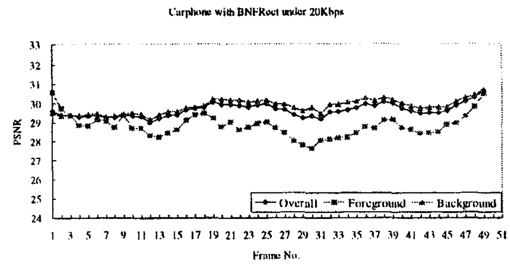


(b)Extended blurring boundary.

Fig. 4. Distortion on boundaries introduced by quantization.



(a)Original sequence.



(b)Blurred non-face rectangle sequence.

Fig. 5. PSNR of foreground, background and overall frame for Carphone under 20Kbps.

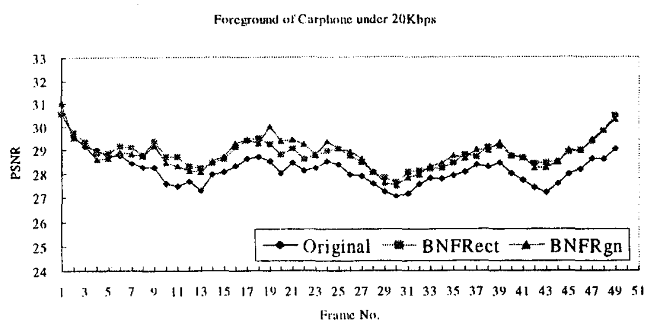


Fig. 6. PSNR of foreground for Carphone under 20Kbps with original sequence, blurred non-face rectangle sequence and blurred non-face region sequence.

(a)Original frame. Bitrate is 24.95Kbps

(b)With 3x3 weighted cross filter. Bitrate is 22.61Kbps

(c)With 3x3 full filter. Bitrate is 21.09Kbps.

(d)With 5x5 full filter. Bitrate is 20.46Kbps.

Fig. 7. Frame 87 of Claire with different blurring filters.

After these steps, a face region has been found. In addition, a face region outside frame (face rectangle) was extracted. The face rectangle will serve as the reference rectangle for the next face rectangle, and we will not rely on FD to get reference rectangle any more, because the person will not move in the same degree all the time.

Some segmentation result is shown in Fig. 2. We find for sequence as complex as "carphone", in which camera is shocking slightly and some of the background is moving, the segmentation result is still good.

4. Considerations in Blurring

Two issues are presented: the decision of filter to use and the decision of blurring boundary.

A. Filter to Use

We use four blurring filters for simulation, as shown in Fig. 3. The degree of blurring effect, from high to low, is (a), (b), (c) and (d). The larger the filter is, the more the blurring will be. For an $n \times n$ filter, computation needed for a single pixel is $(n^2 - 1)$ additions and 1 division, and thus (a) takes six times of additions over (c).

B. Decision of Blurring Boundary

Blurring boundary refers to the boundaries which distinguish regions to be blurred and regions not to be blurred. The direct thought is making blurring boundary just the same as the object boundary. However, due to quantization, high frequency components of the image will be forsaken, leading to some low-pass effect. The low-pass effect, which will make the boundary not so sharp as before[4], will be more evident on the boundaries. We see an example in Fig. 4(a). Assume the horizontal axis represents distance and the vertical axis represents gray level. An object with higher gray level is between boundary B1 and boundary B2. In original image, B1 and B2 is a step boundary. After reconstruction, we can find there will be some distortion near B1 and B2. The gray level at the boundary is smoothed out due to low-pass effect.

The scheme in Fig. 4(b) is utilized to avoid the problem introduced by blurring. Assume B1 and B2 are the real object boundaries. We extend B1 and B2 to B1' and B2', and blur pixels outside B1' and B2' rather B1 and B2. After such modification, the disconnection will still exist near B1' and B2', however, the distortion near B1 and B2 is prevented.

5. SIMULATION RESULT

Fig. 5 shows the PSNR of foreground, background and overall frame of Carphone. They are simulated at 20Kbps. We find the PSNR in foreground is always lower than the PSNR in background. This corresponds to the feature of the sequence, for that the foreground is a man who is moving widely and certainly carries more information, causing the PSNR to lower down. Comparing Fig. 5(b) to Fig. 5(a), we find PSNR of background in (b) is lowered and PSNR of foreground in (b) is raised. This is because some bits in background is re-allocated to foreground, causing the PSNR to re-distribute.

To compare the rank of PSNR of foreground for three sequences: original sequence, blurred non-face rectangle(BNFRect) sequence and blurred non-face region(BNFRgn) sequence, we can see Fig. 6. BNFRect and BNFRgn are always better than original sequence by 0.5~1dB. However, the cost to pay is the degradation of PSNR in background. We can find the degradation of dB in background(2~3dB) is higher than the enhancement of dB in foreground(0.5~1dB). This is reasonable, because it costs more bits to code complex foreground.

6. CONCLUSION

We propose a region-based blurring scheme to reduce bitrate in very low bitrate video coding. The scheme may reduce the bitrate up to 20% by blurring the background image. Quality degradation is not evident because the important region is kept unchanged. Moreover, if we apply full search to the foreground but apply fast search such as 3-step search to the background, the computation time can be saved further. Since the background has already been blurred, the local minimums are smoothed out and thus the fast search algorithm will not degrade too much quality any more.

REFERENCE

[1] ITU-T Study Group 15, Q2/15, "Draft ITU-T Recommendation H.263–Video coding for low bit-rate multimedia communication", Nov. 1995.

[2] Touradj Ebrahimi, Emmanuel Reusens and Wei Li, "New Trends in Very Low Bitrate Video Coding", *Proc. IEEE*, Vol.83, No.7, June 1995

[3] C. A. Christopoulos, A. N. Skodras, W. Philips, J. Cornelis, A. G. Constantinides, "Progressive Very Low Bit-Rate Image Coding", *Proc. Of the Int. Conference on Digital and Signal Processing*, Limassol, Cyprus, pp.443-438, June 26-28, 1995.

[4] Tsuhan Chen, C. T. Swain, and B. G. Haskell, "Coding of Subregions for Content-Based Scalable Video", *IEEE Trans. on Circuit and System for Video Tech.*, Vol. 7, No. 1, Feb. 1997.

[5] Y. Chen, L. Chen and M. Chen, "A Very Low Bit Rate Video Coding System Using Adaptive Region-Classified Vector Quantization", *IEEE ICASSP*, Vol. 2, 1996.