Real Time Software-Only Bit Rate Reducing of MPEG Sequences

S.G. Rodríguez, M. F. López, V. G. Ruiz and I. García

Abstract— **The idea of MPEG bit rate reducing relates to altering or scaling the amount of data in a previously compressed MPEG bit stream. The new scaled bit stream conforms to constraints that are not known nor considered when the original precoded bit stream was constructed. Numerous applications for video transmission and storage have been developed based on the MPEG video coding standard. In this paper, we present and evaluate a real time software implementation of a bit rate reducer system called Real Time Bit Rate Reducer (RT-BR).**

Keywords— **MPEG video encoding, real time, bit rate reducing, video file-server.**

I. Introduction

MPEG (Moving Picture Experts Group) is an international standard for compressed audio and video that is widely used in multimedia applications and interactive television. The standard specifies the compression of an audio and video stream by defining the syntax of a compressed bit stream. Audio and video information is interleaved in the bit stream along with system information required to decode it. There are two main very extended standards: MPEG-1 is designed to deliver VHS-quality video at 1.5 megabits/second (Mbps) [1], while MPEG-2 is designed to deliver studio-quality video at 4-8 Mbps [2].

A video sequence, which is essentially a sequence of still images, is a source of a large amount of data. Compression is necessary to allow conventional video material to be used in computer applications. MPEG uses an algorithm similar to the JPEG (Joint Photographic Experts Group) baseline compression algorithm to reduce the spatial redundancy into a picture and a Motion Estimation technique to exploit the temporal redundancy between pictures.

The main reason that has motivated our work, is the existence, at the present time, of Video On-Demand (VOD) services. Consider a VOD scenario wherein a video file-server includes a storage device containing a library of precoded MPEG bit streams. These bit streams in the library are originally coded at high quality. A number of clients may request retrieval of these videos at a particular time. The number of users and the quality of video delivered to the users is constrained by the outgoing channel capacity. Different users may require different levels of video quality, and the quality of a video will be based on the fraction of the total channel capacity allocated to each user. The requirements that must be satisfied in order to offer video sequences at different quality levels can be obtained with a system that reduces its bit rate.

Department of Computer Architecture and Electronics, University of Almería. E-mail: `segarci,mflopez,vruiz,inma@ace.ual.es`.

The basic function of a bit rate reducer may be thought of as a black box which passively accepts a MPEG bit stream at the input and produces a scaled bit stream which meets new constraints that are not known beforehand during the creation of the original bit stream.

While the idea of bit rate reducing shares many similar concepts as those provided by the various MPEG-2 scalability profiles, the intended applications and goals differ. The MPEG-2 scalability methods (data partitioning, SNR scalability, spatial scalability and temporal scalability) are aimed at providing the encoding of source video into multiple service grades, that are predefined at the time of encoding. The multiple bit stream created by MPEG-2 scalability are hierarchically dependent in such a way that by decoding an increasing number of bit streams, higher service grades are reconstructed. Bit rate reducing methods, in contrast, are primarily computationally low-cost techniques for converting an existing precoded bit stream to another one that meets new bit rate constraints.

In this paper we focus on scaling a MPEG constant bit rate (CBR) encoded bit stream down to a lower bit rate. We assume that for a given rate, the original source is encoded in an optimal way. The basic methods for bit rate reducing are described in Section 2. Our particular implementation is in Section 3. The results and concluding remarks are in Section 4 and 5 respectively.

II. Methods for bit rate reducing

An MPEG video stream is organized as a hierarchy of layers. The first layer, called the Sequence Layer, defines the overall video sequence. Context information for the stream is contained here, such as the image size and frame rate. Below the Sequence layer is the Group of Pictures (GOP). A GOP is a series of one or more pictures intended to assist random access into the sequence.

A picture consists of three rectangular matrices of eight-bit numbers: a luminance matrix (Y), and two chrominance matrices (Cb and Cr). There are three types of pictures that use different coding methods: a Intra-coded (I) picture is coded using information only from itself, a Predictive-coded (P) picture is a picture which is coded using motion compensated prediction from a past I picture or P picture, and a bidirectionally predictive coded (B) picture is a picture which is coded using motion compensated prediction from a past and/or future I picture or P picture. These dependencies are shown in Figure 1.

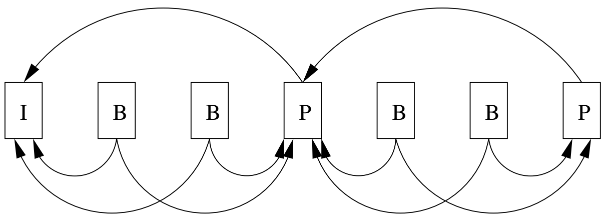

Fig. 1. Dependencies between I, P and B pictures.

A picture is divided in several slices. A slice is a series of an arbitrary number of macroblocks, from left to right and top to bottom in the picture. A macroblock is a portion of image that consists of 16x16 pixels. Each macroblock contains several blocks of luminance and chrominance. The block is the smallest coding unit in the MPEG algorithm.

Figure 2 shows a simplified structure of an MPEG encoder. In this structure, each block of image data is first transformed to a set of coefficients c. The coefficients are then quantized $\widetilde{c}$ with a quantizer step which is decided according to the given bit rate.

When encoding P or B picture the encoder searches each macroblock for the most similar block of pixels from the previously encoded pictures, using Motion Estimation. As a result, the encoder transmits motion vectors (that represent the relative coordinates of macroblocks) and the difference between current and preceding macroblocks. Motion Estimation is usually the most time consuming part of MPEG encoding. The principal way by which the encoder controls the bit rate is to vary the quantizer scale.

The direct form to obtain a video bit stream at smaller bit rate than the original one is to decode and encode again with the new requirements. We call it Brute Force Bit Rate Reducer (BF-BR), and is a slow bit rate reducer. The main way by which an MPEG video encoder controls the bit rate is to vary the quantizer scale. This parameter is set in each slice and may be set at the beginning of any macroblock. Therefore, there are three main methods for faster bit rate reducing. From lower to higher level complexity they are:

1. Elimination of the high frequency coefficients from $\widetilde{c}$.
2. Requantization of $\widetilde{c}$ at a coarser quantizer scale.
3. Re-encoding of the reconstructed pictures with motion vectors and coding decision modes extracted from the original high quality bit stream.

Each of them has its own particular benefits that are suitable for particular applications. Methods 1 and 2 are candidates for software based implementation. We have chosen to implement the second one because its performance is better [3].

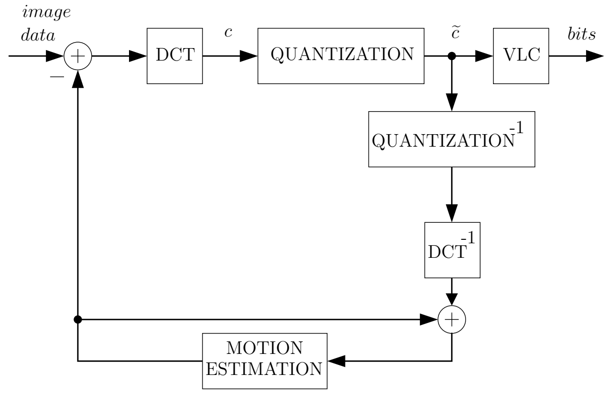

Fig. 2. Simplified MPEG-1 & MPEG-2 encoder.

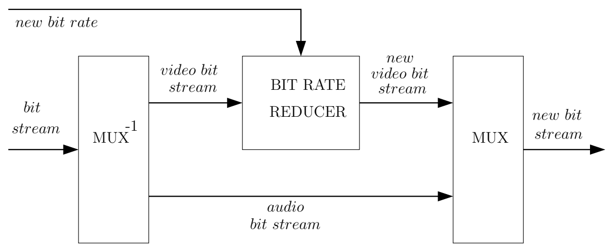

Fig. 3. General scheme for a bit rate reducing system for MPEG.

III. Description of Real Time Bit Rate Reducing System

Figure 3 shows a first level of complexity of the Real Time Bit Rate Reducing (RT-BR) system that we have implemented for the task of reducing the bit rate of a MPEG bit stream. The inputs are the MPEG bit stream, which is demultiplexed to separate the video and audio bit streams, and the new bit rate at which it is desired to obtain the new video bit stream, which is multiplexed together the original audio bit stream to obtain the final output. The task of bit rate reducing of the original video bit stream is performed using method 2 introduced before.

This method is based on the requantization of the DCT coefficients after increasing the quantization step. Figure 4 shows the operation of method 2 to reduce the bit rate of a video bit stream. First of all, the video bit stream is parsed and, if necessary, some information is variable length decoded. Before requantization the DCT coefficients it must be obtained a new quantizer scale. This is performed by the process RATE CONTROL in a simple and efficient way. Given the size of the bit stream of the original video it is estimated the size of the output video bit stream according to the ratio $\frac{bit\_rate}{new\_bit\_rate}$.

The DCT coefficients $\widetilde{c}$ are variable length decoded, dequantized and quantized with a new, coarser, quantizer scale, obtaining $\widetilde{\widetilde{c}}$. The process UPDATE INFO modifies the information necessary from $info$ to produce a bit stream conformed to the standard after the requantization of the blocks. It sets the new quantizer scale at the slice and macroblock layers; and the new coded block pattern, macroblock address increment and macroblock type

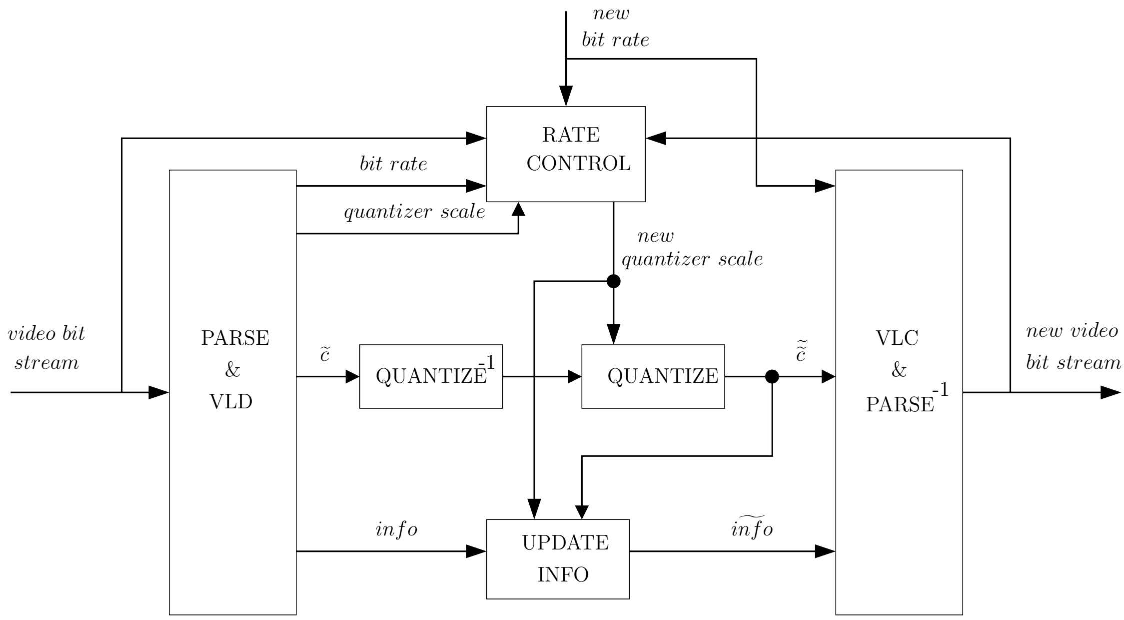

Fig. 4. Our bit rate reducer.

at macroblock layer. This is necessary because, after the requantization, it can vary the number of blocks that are variable length coded in each macroblock. At last, the task VLC & PARSE^{-1} produces the new video bit stream coding the data and making the inverse parsed.

This bit rate reducer introduces in the new bit stream drift errors, caused by a drift between the decoder's prediction image and the prediction image used. Drift errors occur in P and B pictures and can accumulate in P pictures until the next I picture is processed. Therefore, the temporal distance of I pictures has an important impact on the visibility of drift related artifacts.

IV. Results

Next we present the results of the comparison of our codec RT-BR as opposed to the alternative to decode, and subsequent encoding (BF-BR). We have used a series of standard test sequences. These sequences are: *Bus* (150 frames), *Container* (300 frames), *Flower Garden* (250 frames), *Foreman* (300 frames), *Mobile* (300 frames) and *Tempete* (260 frames). All of them have a resolution of 352×288 pixels/frame, 24 bpp of depth and 30 Hz of frame rate.

We have compared RT-BR and BF-BR using an objective measurement, concretely the PSNR (Peak Signal-to-Noise Ratio); and a subjective measurement, the visual quality of the reconstructed sequences. PSNR is defined as

$$\mathrm{PSNR} = 10\log_{10}\frac{255^2}{\frac{(\mathrm{MSE(Y)+MSE(Cb)+MSE(Cr)})}{3}}\mathrm{dB} \tag{1}$$

where, for example, MSE (Mean Squared Error) of the luminance is calculated as

$$\mathrm{MSE(Y)} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i=1}}^{\mathrm{N}}(\mathrm{Y[i]} - \mathrm{Y'[i]})^2 \tag{2}$$

being $Y[i]$ the value of luminance for the point $i = (x, y, t)$ of the reference picture, $Y'[i]$ the corresponding value in the decoded picture and N the number of pixels in each sequence. In our experiments the reference sequences were encoded at 4 Mbps using Test Model 5 [4] with software downloaded from www.mpeg.org. The GOP pattern is, in all cases, IPBBPBBPBB.

TABLE I
PSNR of RT-BR y BF-BR for the test sequences.

	0.5 Mbps		1.0 Mbps		2.0 Mbps	
Sequence	RT-BR	BF-BR	RT-BR	BF-BR	RT-BR	BF-BR
Bus	28.0	29.7	29.0	30.9	31.7	33.2
Container	30.2	35.0	33.0	37.0	35.8	38.3
Flower G.	24.6	26.3	25.6	28.0	27.8	29.7
Foreman	32.4	34.3	34.5	36.9	37.1	38.8
Mobile	24.6	26.1	26.2	28.4	28.8	30.8
Tempete	28.5	30.3	29.9	32.6	32.7	33.9

We have verified the quality of the reconstructions using RT-BR and BF-BR at different bit rates: 2 Mbps, 1 Mbps y 0.5 Mbps. The results are shown in the Table I. We can conclude that:

1. RT-BR generates, on average, inferior rates of compression than BF-BR. This is mainly due to correction drift errors.
2. RT-BR shows a loss in the quality of the reconstructions, that varies between 1.2 and 4.8 dB.

We have measured the gain in time that takes place when we used RT-BR. The measures were taken in a PC with an intermediate power of calculation:

TABLE II

Times (in seconds) of RT-BR y BF-BR for the test sequences.

		0.5 Mbps		1.0 Mbps		2.0 Mbps	
Sequence	t_v	RT-BR	BF-BR	RT-BR	BF-BR	RT-BR	BF-BR
Bus	5.0	3.6	50	3.7	51	3.9	51
Container	10.0	7.3	85	7.6	86	8.0	87
Flower G.	8.3	4.9	69	5.2	69	5.7	70
Foreman	10.0	7.5	89	7.8	90	8.5	90
Mobile	10.0	6.9	84	7.0	84	7.6	85
Tempete	8.6	6.2	70	6.2	71	6.8	71

1. Operating system GNU/Debian Linux.
2. Processor AMD K7 1 GHz (1998.84 BogoMIPS).
3. 512 MB RAM.
4. gcc compiler version 3.2.2.

In order to measure the times we used the utility *time* from the operating system.

The results are shown in the Table II. We deduce that:

1. RT-BR is executed in real time. Run times never exceed the visualization times (t_v) of the sequences.
2. RT-BR is between 10 and 14 times faster than BF-BR. RT-BR avoids the heavy computational task of Motion Estimation.

Finally, we have chosen some reconstructed pictures for a subjective comparison. The images appear in the Figures 5 and 6. Reconstructions generated with RT-BR are on the left side and, on the right side, the images obtained with BF-BR. It can be observed that most of the artifacts appear in the zones where there is more movement and more detail. This type of distortion is due to the prediction drift errors introduced by RT-BR.

In spite of this deficiency, the quality of the images reconstructed by the decoder is acceptable and RT-BR allows MPEG-1, and MPEG-2, video streaming in real time.

V. Conclusions

In this work we have presented a software implementation of a bit rate reducer for MPEG-1 and MPEG-2 bit streams. RT-BR is performed in real time with medium speed computers.

The implementation of RT-BR is based on modules, therefore it is easy to replace the algorithms that have been used and test the performance of others. In addition, it can be mapped on systems with several processors, this way, heavier algorithms can be used and higher resolution sequences can be processed.

In order to measure the quality of the videos generated by RT-BR. We have compared its performance to the alternative of decoding and encoding again. The results demonstrate that the loss of quality caused by prediction drift errors is not significant and RT-BR can be executed to make streaming in real time.

References

[1] ISO/IEC JTC1 SC29 WG11, *International Standard ISO/IEC 11172*, 1993.

[2] D. LeGall, "MPEG: A Video Compression Standard for Multimedia Applications," *Communications of the ACM*, vol. 34, no. 4, April 1991.

[3] H. Huifang Sun, W. Kwok, and J.W. Zdepski, "Architectures for MPEG Compressed Bitstream Scaling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 2, pp. 191–199, April 1996.

[4] International Organisation for Standardisation (ISO/IEC JTC1 SC29 WG11), *Test Model 5*, 1993.

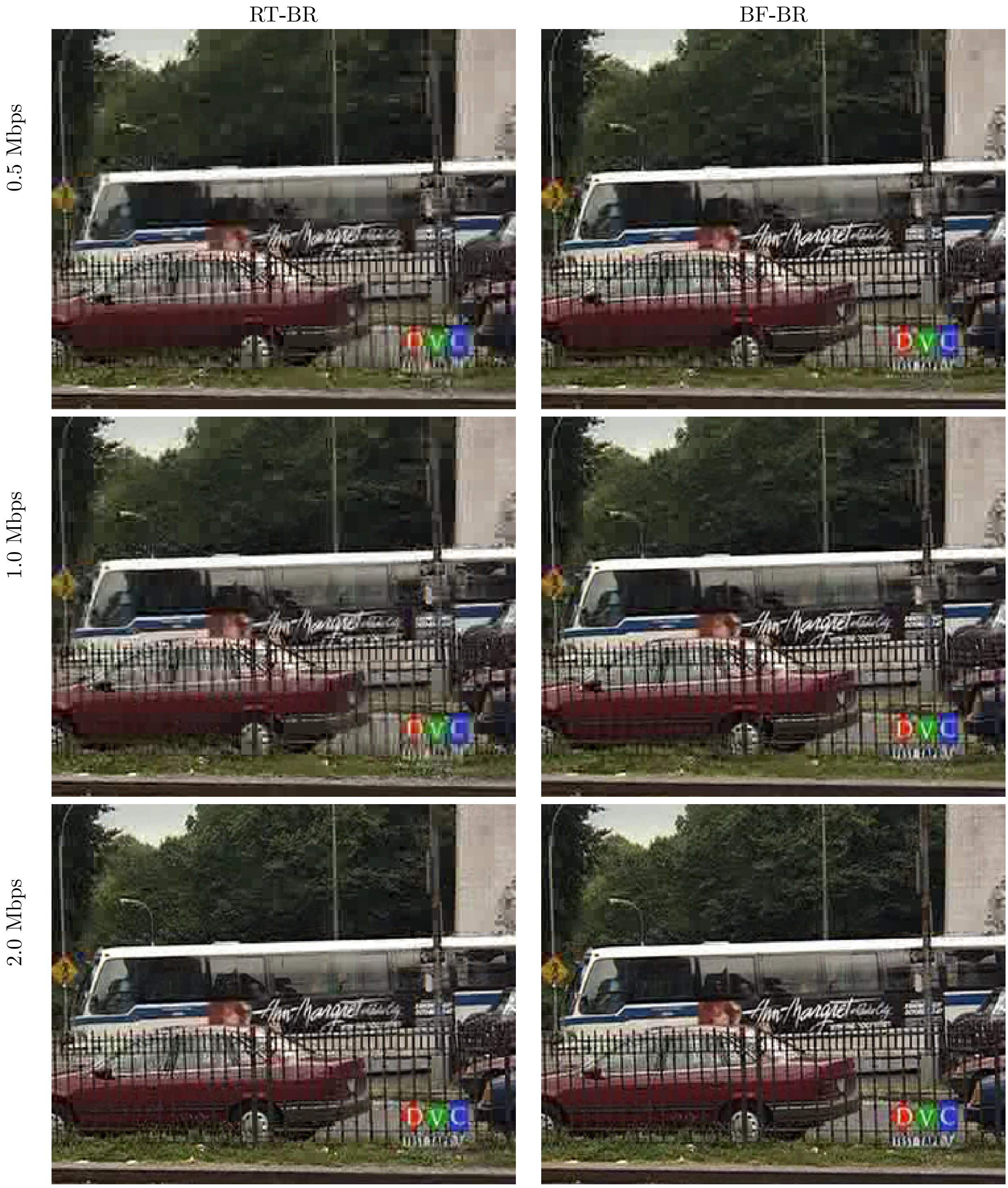

Fig. 5. Visual comparison for the *Bus* sequence (picture 99).

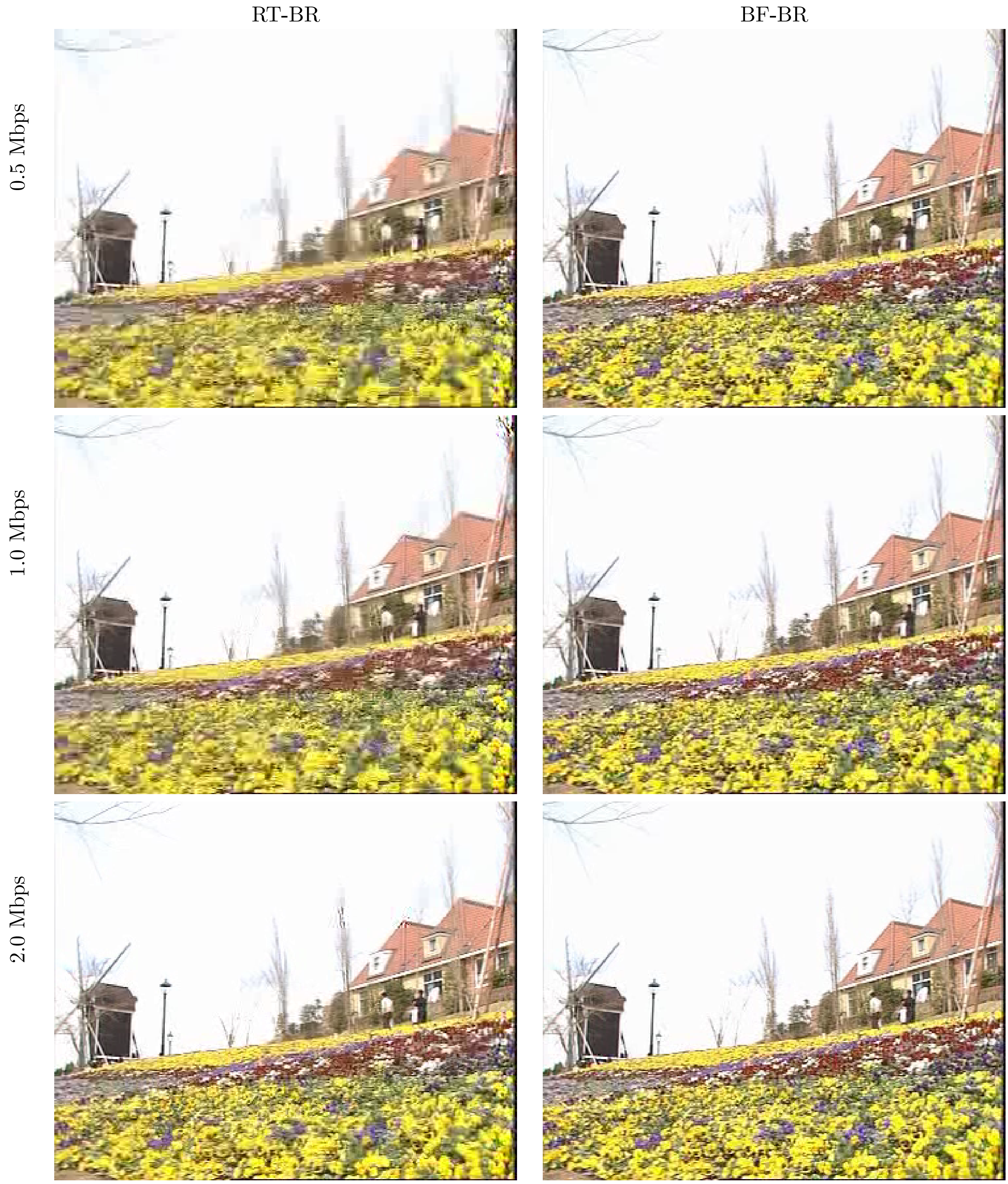

Fig. 6. Visual comparison for the *Flower Garden* sequence (picture 223).