A Guided Search Method for Real Time Transcoding a MPEG2 P Frame into H.263 P Frame in a Compressed Domain

Euisun Kang[1], Maria Hong[2], Younghwan Lim[1], Youngsong Mun[3], and Seongjin Ahn[4]

[1] Department of Media, Soongsil University, Seoul, Korea
{kanges, yhlim}@media.ssu.ac.kr
[2] Digital media Engineering, Anyang University, Kyonggi-do, Korea
maria@anyang.ac.kr
[3] Department of computer Science, Soongsil University, Seoul, Korea
mnu@computing.ssu.ac.kr
[4] Department of Computer Education, Sungkyunkwan University, Seoul, Korea
sjahn@comedu.skku.ac.kr

Abstract. Our objective is to enable a format transcoding between a heterogeneous compression format in real time mode and to enhance the compression ratio using characteristics of the compression frame. In this paper, we tried to transcode MPEG 2 digital contents having a low compression ratio into H.263 contents with a high compression ratio. After analyzing MPEG2 bit stream and H.263 bit stream of the same original video, we found that the number of intra coded macro blocks in MPEG2 data is much higher that the number of the intra coded blocks in H.263 data. In the process of P frame generation, an intra coded block is generated when a motion estimation value representing the similarity between the previous frame and the current frame does not meet a threshold. Especially the intra coded macro block has a great impact on the compression ratio. Hence we tried to minimize the number of intra coded macro blocks in transcoding the INTRA coded block into INTER coded block using the information about motion vectors surrounding the intra macro block in order to minimize the complexity of the motion estimation process. The experimental results show that the transcoding of MPEG2 into H.263 can be done in real time successfully.

1 Introduction

In ubiquitous communication environments, the devices for accessing a digital item have many different (even unpredictable) characteristics. Therefore the contents should be adopted according to the system device characteristics, network bandwidth and user preferences.

Digital item adaptation (DIA) is one of main parts in MPEG21. The goal of the DIA is to achieve interoperable transparent access to multimedia contents by shielding users from network and terminal installation, management and implementation issues. As shown in Fig.1, the combination of resource adaptation and descriptor adaptation produces newly adapted Digital Item.

O. Gervasi et al. (Eds.): ICCSA 2005, LNCS 3480, pp. 242 – 251, 2005.

DIA tools store all the necessary data of descriptor information [1]. There could be a variety of Digital Item that need to be managed in the DIA framework. Depending on the characteristics of digital item, the architecture or implementation method of Resource Adaptation Engine (RAE) may be different. Current research works for RAE are focused on the transcoding of single medium transcoder such as format transcoder, size transcoder, frame rate transcoders, MPEG2 into H.263 transcoder, or 3D stereoscope video into 2D video transcoder[2] etc. Also they provide a fixed single transcoder for each case of transcoding need.

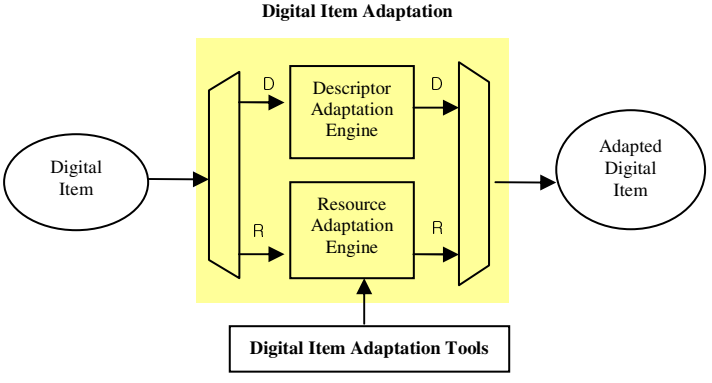

Fig. 1. Illustration of DIA

As mentioned above, researches for RAE are focused on the transcoding of single medium transcoder. And there are many research works about the adaptation with multimedia contents, especially MPEG series. Their concerns are on metadata for mobility characteristics[3], description tools[4] and metadata driven adaptation[5]. They provide a fixed single transcoder for each case of transcoding need. With these methods, if the QoS of source and the QoS of destination get changed, the same transcoder that was used for the adaptation at the last time cannot be used again. And in the previous studies, they suggested the frame rate, color depth and resolution in linking order of transcoders[6]. They considered network bandwidth and requiring bandwidth from a mobile host about multimedia content in mobile environment. In another study, they suggested the bit rate based transcoding with frame rate and resolution transcoding concurrently[7]. They considered network environment and capacity of server and client for multimedia service in real-time. But both of them did not consider the user preference about content occurring in real-time. And there was not clear explanation about a criterion to decide linking sequence between transcoders

One of critical problems of these works is that the transcoding method suggested can not be used to an application requiring a real time adaptation. In order to meet the real time, the MPEG2 contents should be transcoded in a compressed domain.

Comparing the MPEG 2 data and H.263 data for the same video, we found H.263 data has fewer I frames and more P frames than MPEG 2 data. More important difference we pay our attention is that the P frames of H.263 have much more INTER

coded macro blocks than MPEG 2 P frame. The our main efforts for transcoding MPEG 2 data into H.263 data are spent to find a method of transcoding the INTRA coded blocks in P frame of MPEG 2 data into the corresponding INTER coded macro blocks in P frame of H.263. For generating corresponding INTER macro blocks in P frame of H.263, we must find motion vectors by any means of search method. Our key idea is that we can have the guiding information from the corresponding INTRA macro blocks in P frame of MPEG2. The guide information may accelerate search procedure.

In the following chapter, we describe the concept of transcoding MPEG2 into H.263 in compressed domain with problems. Then a guided search algorithm and experimental results are explained.

2 Heterogeneous Video Transcoding

2.1 Format Transcoding

There are three kinds of methods that convert MPEG2 into H.263 with characteristic of frame, I frame to I frame, I frame to P frame, P frame to P frame [10][11].

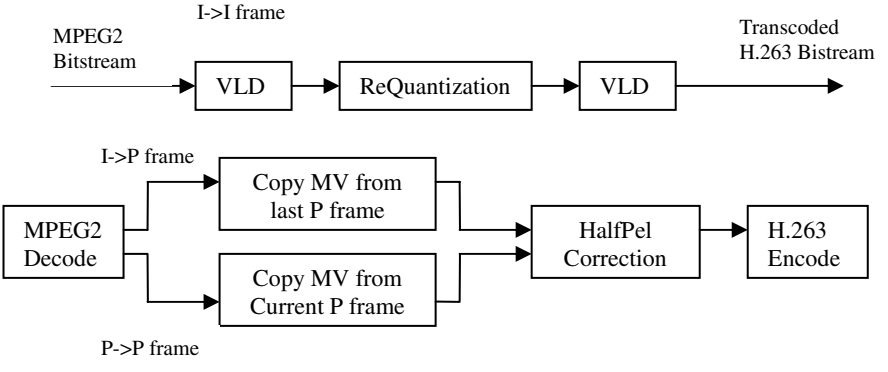

Fig. 2. The methods using the characteristics of frames

As depicted in Fig 2, it is fast to convert I frame to I frame due to avoid IDCT and DCT computations. The rest of conversion between frames requests a little complexity though it can look forward to being higher compressibility for re-sampling new motion vectors with Half-pel Search.

2.2 The Difference Between MPEG2 P Frame and H.263 P Frame

Each macroblock estimated by temporal compression for encoding a series video stream, is encoded in intra or inter mode. Inter mode makes compress ratio higher because of a motion vector that provides an offset from the coordinate position in the current picture to the coordinates in a reference picture. Otherwise, intra mode encoded similarly like spatial compress takes lots of bits than inter mode. Accordingly,

we attempt to apply to real-time and enhance compression ratio using property of macroblock by temporal predictions. The following Figure shows the number of intra macroblock within MPEG2 and H.263 at temporal compression.

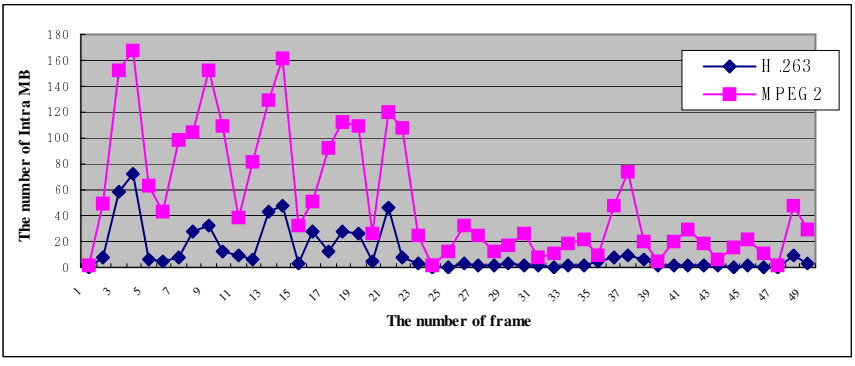

Fig. 3. The number of intra macroblocks in each frame of H.263 and MPEG2

In Fig 3, we can observe that MPEG2 has more intra coded blocks than H.263 has. The reason is that the value of threshold, used to decide whether the macroblock should be intra or inter coded, is different in temporal compression of MPEG2 and H.263. In this paper, in order to solve this problem we tried to convert intra macroblocks within P frame of MPEG2 into inter macroblocks of H.263 appropriately in real time, to enhance the compressibility by decreasing the number of intra macroblocks of MPEG2 and thus to prevent the damage of screen quality as much as possible after conversion.

2.3 Conversion of Macroblock in the Compressed Domain and the Problem

The simplest method of converting the intra macroblock within P frame of MPEG2 into the inter macroblock of H.263 is to carry out the process of motion estimation with the threshold of H.263 in the pixel domain. In this method, it is, however, computationally intensive for real time applications because all the processes of motion estimation are carried out. To alleviate this problem, we tried to apply information of motion vectors and modes of macroblocks within MPEG2 in compressed domain to analogize a new vector. This can reduce the computation all over due to perform without fully decoding and then re-encoding the video.

Figure 4 shows the result of PSNR in case of converting the intra macroblock into the inter macroblock. Intra 0 to Intra 8 indicate the number of adjacent macroblocks surrounding an intra macroblock. For example, “Intra 4” means that there are fewer than 4 Intra macroblocks.

If the motion vector points out wrong position, the video quality may be damaged because the motion vector refers to the most similar block from the previous frames. It can be seen through PSNR that the video quality decreases, as the number of adjacent intra macroblocks is larger in the above figure 3. The reason is that a newly calculated motion vector has a value of another specific point differently from the direc-

tion or the movement level of adjacent motion vectors. Additionally, when the block of the initially restored frame is broken in restoration, the current frames that refer to the previous frames for their restoration become broken more clearly, so that we cannot perceive to the naked eye.

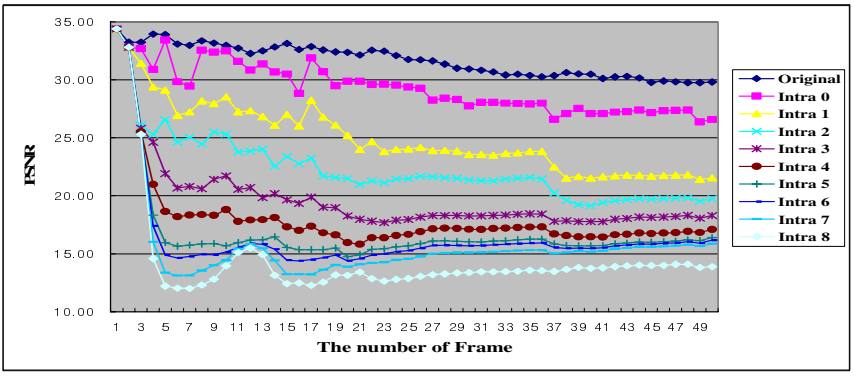

Fig. 4. PSNR in case of employing the adjacent Motion vectors

Therefore, we propose a guided search method for efficiently transcoding the intra coded block within MPEG2 into inter coded block within H.263. The guided search method reuse the information about motion vectors surrounding intra blocks and estimate motion about only intra macroblock in order to consider the problem of quality and minimize the cost of computation.

3 The Guided Search Algorithm Using the Adjacent Motion Vectors

As mentioned above, we can confirm that the problem of quality is generated by motion vectors pointing out wrong positions. Hence, we propose two efficient motion estimation algorithms, making use of neighboring motion vector for image quality, though it is a little more complicated temporally in converting macroblocks in the compressed domain. It can reduce the entire computation of conversion in comparison with the motion estimation process because it performs the motion estimation process not for all the macroblocks in frames but only for the intra macroblock within MPEG2 P frame. So far, many search algorithms [4] have been introduced to minimize a large amount of computation. This paper attempts to reduce the complexity of converting MPEG2 P frame into H.263 P frame by omitting step 1 in three-step search algorithm that has center-biased characteristics among many search algorithms of [4].

3.1 Two-Step Search Algorithm(TSSA)

Three-step search algorithm determines the direction of motion in step 1. Therefore, if the best direction has been decided, the step 1 can be avoided. TSSA starts with 8

motion vectors around an intra macroblock in order to determine the direction of motion and then the other points for comparison are selected based on the following algorithm.

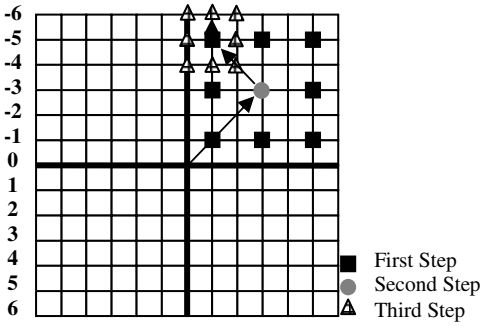

Fig. 5. Route of Two Step Search

[Algorithm 1] Two Step Search Algorithms

Step 1. Estimate the direction of motion using the adjacent 8 motion vectors. In order world, after setting the center of the whole search range, decide one of adjacent 8 locations included in 1/2 of the whole search area from the center. Among 9 points around search window center, find a point which the cost function is the smallest and proceed with step 2. If the direction of motion is 0 (in order words, the direction indicate the center), starts with step 1 of three-step search algorithm.

Step 2. Set adjacent 8 locations included in $1/2^2$ of the whole search range around the estimated direction. Among the 9 points including the estimated direction, the cost function is applied to new 8 points and decides an optimal point that has the minimum value.

Step 3. Find a point whose value of cost function is the fewest among the adjacent 8 macroblocks around the position found in step 2. In this step, if the smallest value of cost function is larger than the threshold of H.263, the current macroblock is decided by intra macroblock.

As seen in this process, because the direction is decided in first step of three-step search algorithm using the adjacent motion vectors, the algorithm can avoid 8 comparisons in the first step for finding the most similar position.

3.2 Axis Division Search Algorithm (ADSA)

TSSA can reduce all the frequencies of comparison considerably than three-step search algorithm because it reduces 9 times of comparisons that are performed in step 1 to 1 time. However, it cannot improve the compression ratio greatly than three-step

search algorithm because its accuracy of finding the most similar macroblock is lower. ADSA can make the TSSA more accurate and increases the number of comparisons performed in step 1.

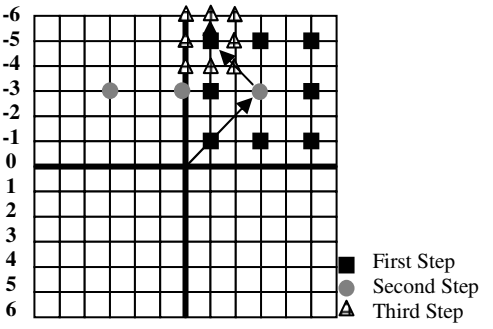

Fig. 6. Route of Axis Division Search Algorithm

[Algorithm 2] Axis Division Search Algorithm

Step 1. Estimates the direction of motion using the adjacent motion vector and set 3 positions horizontally or vertically in 1/2 of the whole search range. Then, decide a point whose cost function is the minimum and proceed with the next step. If the estimated direction is 0, starts with step 1 of three-step search algorithm.

Step 2. Set adjacent 8 positions included in $1/2^2$ of the search range around the point decided in step 1. Then, calculate cost function about 8 positions and select the best position whose weight of the cost function is the fewest.

Step 3. Find a point whose value of cost function is the fewest among the adjacent 8 macroblocks around the position found in step 2. In this step, if the smallest value of cost function is larger than the threshold of H.263, the current macroblock is decided by intra macroblock.

ADSA depends on 3 positions vertically or horizontally for solving the about deciding the direction of motion in step 1. Though the frequencies of comparison performed in step 1 are difficulty much more than TSSA, it can enhance accuracy and compression ratio.

4 Results of Experiment

In order to evaluate the performance, we captured 50 frames with CIF(352*288) resolution, the sequence were pre-coded as MPEG2 bit stream, and IPPP…. structure (i.e., the first frame was intra coded and the rest of the frames were inter coded.). These

convert into H.263 with a lower bitrate by supported algorithms. We compare the performance of TSSA and ADSA with three-step search algorithm.

4.1 Frequency of Comparison

The frequency of comparison indicates the frequency of comparison for finding the most appropriate cost function for each step. The following figure shows the frequency of comparison for all frames using three-step, TSSA, and ADSA in order to convert the intra macroblocks within P frames of MPEG2 into the inter macroblocks of H.263. In this figure, the number of adjacent intra macroblocks of the x-axis indicates the number of cases that the type of macroblock adjacent to the converted macroblock is intra. For example, "4" means that there are 4 intra macroblocks among adjacent macroblocks.

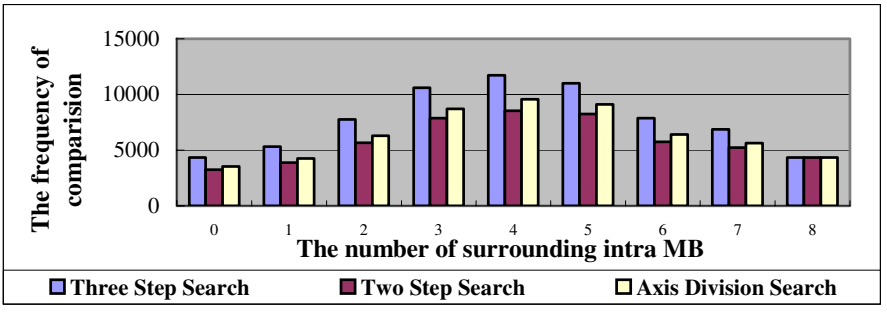

Fig. 7. The frequency of comparison by the number of adjacent Macroblocks according to the search algorithm

As a whole, the frequency of comparison is the most in three-step search algorithm, 9 times of comparison is performed in step 1, the second most in the ADSA, 3 times of comparison are performed in step 1, and the least in TSSA, step 1 is omitted. In case that neighboring intra macroblocks are less 8, the proposed algorithm can decrease the frequency of comparison up to 27% (TSSA) and 25% (ADSA).

4.2 Compressibility

The following figure shows compressibility for all frames in using the suggested guided search algorithm and three-step search algorithm. The search algorithm suggested in the above figure did not increase the compressibility than three-step search algorithm. The reason is that the suggested guided search algorithm was applied only to intra macroblocks. The number of intra macroblocks of TSSA is more than that of three-step search algorithm in comparing the threshold of H.263 because the direction of search is decided in step 1. And, the ADSA compares 3 times more than TSSA, so its number of intra macroblocks is less than that of TSSA, but it is more than the intra macroblocks of three-step search algorithm.

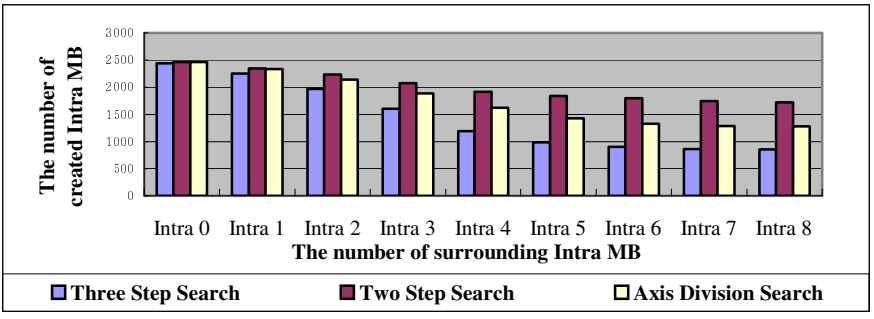

Fig. 8. The number of intra macroblocks generated after applying the guided search algorithm

4.3 Valuation of Image Quality

We also compare PSNR (Peak Signal to Noise Ratio) of images, transcoded by three-step search algorithm and proposed search algorithms. The following table indicates PSNR according to the number of adjacent intra macroblocks in each search algorithm. We confirm that TSSA of three search algorithms shows the best quality. The reason is that if all the neighboring macroblocks is intra coded, macroblocks which desire converting are not used to estimate the new motion vectors.

Table 1. PSNR by each search algorithm

	Three Step Search	TSSA	ADSA
Intra 0	31.60	31.61	31.61
Intra 1	31.53	31.55	31.56
Intra 2	31.43	31.53	31.51
Intra 3	31.30	31.48	31.44
Intra 4	31.19	31.44	31.37
Intra 5	31.10	31.42	31.31
Intra 6	31.07	31.41	31.29
Intra 7	31.06	31.40	31.27
Intra 8	31.06	31.40	31.27

5 Conclusion

Today, it is possible to acquire information anywhere, anytime by development of the wireless technology. However, the information cannot be found easily if not considering the environments of the sender (e.g Server) and the receiver (e.g device with small display, memory, power or etc.). To consider the environment of the receiver in this paper, we employed the type of video compression from the information of QoS(Quality Of Service) of the receiver and converted it appropriately, and to consider the environment of the network, we minimized the number of intra macroblocks within P frames of MPEG2 for increasing the compressibility. We proposed the guided search algorithm (TSSA and ADSA) using the motion vectors of 8 macrob-

lock information surrounding an intra macroblock in order to reduce the complication of motion estimation. Besides we obtained the experiment result that the guided search algorithm has the less frequency of comparison than the existing algorithms and that the whole processing time can be reduced in converting P frames of MPEG2 into P frames of H.263.

Acknowledgments. This work was supported by grant No. R01-2004-000-10618-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

References

1. Youn, J., Xun, J., and Ming, S., "Fast Video Transcoding Architectures for Networked Multimedia Applications," IEEE International Symposium on Circuits and Systems, pp.25-26, 2000.
2. Kang, E., S., "A Study on the Guided Search for Transcoding of MPEG2 P frame to H.263 P frame in a Compressed Domain," Department of Computing Graduate School Soongsil University., pp. 1-15, 2001.
3. Zhijun, L., and Georganas, N., D., "H.263 Video Transcoding for Spatial Resolution Downscaling," Proceedings of the International Conference on Information Technology: Coding and Computing, pp. 425-430, 2002.
4. Fung, K., T., and Siu, W., C., "Low Complexity and High Quality Frame-Skipping Transcoder," IEEE International Symposium on Circuits and Systems, Vol. 5, pp. 29-32, 2001.
5. Yang, J., and Edward J.. "A MPEG4 simple profile transcoder for low data rate wireless applications, "Proceedings of SPIE Visual Communications and Image Processing Part 1, pp. 112-123, 2002.
6. Kim, K., H., Park, D., S., and Whang, J., S., "Compiler," Korean Broadcasting University Press, pp. 96-112, 1998.
7. Huang, C., Liu, P., and Chang R., L., "QoS Streaming Based on a Media Filtering System," IEEE International Conference on Parallel and Distributed Systems, pp. 661-666, 2001.
8. Soam Acharya, Brain C. Smith. : Compressed Domain Transcoding of MPEG. ICMCS (1998) 295-304
9. G. Keesman, R. Hellinghuizen, F. Hoeksema, and G. Heideman. : Transcoding MPEG bitstream," Signal Processing: Image Communication, vol. 8 September (1996)
10. T. Shanableh and M. Ghanbari. : Heterogeneous video transcoding MPEG: 1,2 to H.263. in International Packet Video Workshop, New York City, NY, April (1999)
11. N. Feamster and S.J. Wee. : An MPEG-2 to H.263 Transcoder. SPIE Voice, Video, and Data Communications Conference, Boston, MA, September (1999)
12. Oliver Werner. : Requantiazation for Transcoding of MPEG-2 Intraframes. IEEE transaction on image processing, Vol, 8, No. 2, February (1999)
13. N. Bjork and C. Christopoulos. : Transcoder architecture for video coding. IEEE Trans. Consumer Electron., vol. 44, Feb. (1998) 88-98
14. B. Shen, I. K. Sethi and V. Bhaskaran. : Adaptive motion vector resampling for compressed video down-scaling. Proc. Of IEEE Int. Conf. On Image Processing, Oct. (1997)