Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

# Efficient video downscaling transcoder from MPEG-2 to H.264*

Xiang-wen WANG, Jun SUN, Rong XIE, Song-yu YU

(*Shanghai Key Lab. of Digital Media Processing & Transmissions, Institute of Image Communication & Information Processing, Shanghai Jiao Tong University, Shanghai 200240, China*)

E-mail: {wxw21st, junSun, Xierong}@sjtu.edu.cn; Syyu@cdtv.org.cn

Received Nov. 8, 2007; revision accepted Jan. 15, 2008

**Abstract:** The new H.264 video coding standard achieves significantly higher compression performance than MPEG-2. As the MPEG-2 is popular in digital TV, DVD, etc., bandwidth or memory space can be saved by transcoding those streams into H.264 in these applications. Unfortunately, the huge complexity keeps transcoding from being widely used in practical applications. This paper proposes an efficient transcoding architecture with a smart downscaling decoder and a fast mode decision algorithm. Using the proposed architecture, huge buffering memory space is saved and the transcoding complexity is reduced. Performance of the proposed fast mode decision algorithm is validated by experiments.

**Key words:** Video transcoding, Mode decision, Edge direction analysis
**doi:**10.1631/jzus.A071585 **Document code:** A **CLC number:** TN919.8

## INTRODUCTION

The new H.264 standard achieves significantly higher compression performance than previous standards. The old MPEG-2 standard (MPEG, 1993) is the most common video standard adopted in the multimedia industry that finds wide applications in digital TV, DVD, etc. As bandwidth or memory space can be saved by transcoding from previous standards to H.264, the video transcoding plays an important role in multimedia applications such as video gateways, media proxies, and digital video recorders (DVRs). In these applications, video transcoding with spatial resolution reduction should adapt for various terminal devices. In this paper, we focus on transcoding from MPEG-2 to H.264 with spatial resolution downscaling by two in each dimension.

It is well known that the higher compression performance of the H.264 is at the cost of higher computation complexity. Among the new coding features adopted in H.264, the motion estimation with seven variable block sizes and multiple references is the most complex module and affects the compression performance significantly. For instance, in the reference H.264 encoder JM (http://bs.hhi.de/suehring/tml/download), the coding mode for each inter-MB is decided after seven times of motion estimations with different block sizes ranging from 16×16 to 4×4 and two times of intra predictions with two sizes: 4×4 and 16×16. The huge complexity of the coding mode decision keeps it from practical applications.

Whereas, considering that the motion and the mode information in the MPEG-2 input stream can be re-used for the H.264 re-encoder in the transcoding situations, the burden of the mode decision can be relieved significantly. Several fast mode decision (Chang *et al*., 2003) and motion vector reusing schemes have been proposed to reduce the mode decision complexity for the re-encoder (Xin *et al*., 2002). A motion vector (MV) and coding mode re-using method is proposed in (Zhou *et al*., 2005) which uses the MV in input stream as prediction MV and splits an MB following a "top-down" procedure. The information such as distortion and coding modes are used to de-

* Project (No. CNGI-04-15-2A) supported by the China Next Generation Internet (CNGI)

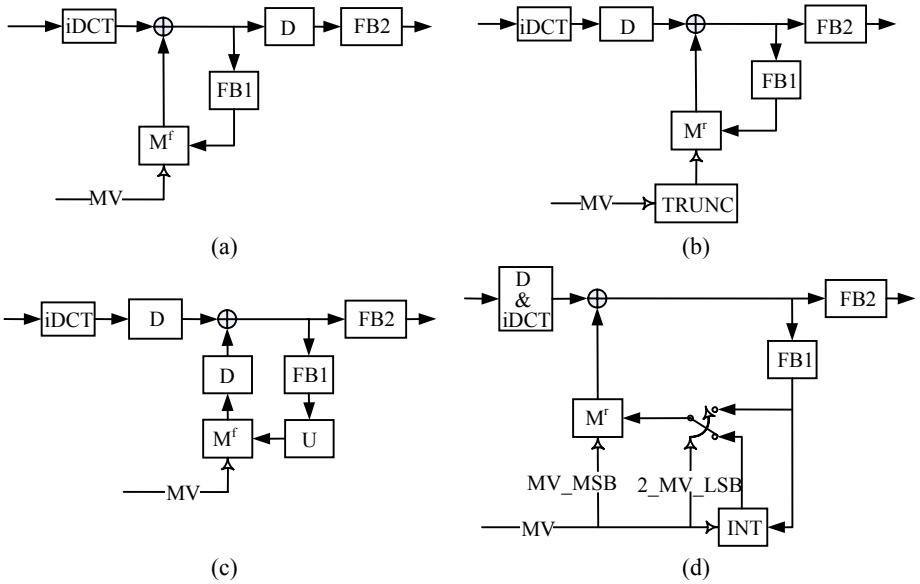
termine which modes can be eliminated and the MVs from MPEG-2 are utilized for an H.264 re-encoder using EPZS (enhanced predictive zonal search) in (Lu *et al.*, 2005). Two "bottom-up" merging schemes combined with early-stop strategies for variable sizes selection algorithms are proposed in (Kucukgoz and Sun, 2004; Bu *et al.*, 2006). In all these algorithms, only the input motion vectors and coding modes are exploited for fast mode decision while the information in the input residual signal is never exploited.

In this paper, we mainly propose a fast mode decision algorithm for the video downscaling transcoder from MPEG-2 to H.264 by exploiting the useful information carried by the MVs as well as the residual signal in the input steam. An efficient transcoding architecture including a smart downscaling decoder and the fast mode decision is also proposed. Much buffering memory space and a great number of operations are saved by the downscaling decoder. Furthermore, with the proposed fast mode decision, huge complexity is reduced by deciding the coding block size for each inter-MB without the motion estimations at the H.264 re-encoding stage. This efficient transcoding architecture is especially suitable for practical applications with limited computation resources.

The rest of the paper is organized as follows. We first present the smart transcoding architecture in Section 2. The fast mode decision is then proposed in Section 3. Some experimental results of the proposed fast mode decision and a discussion are presented in Section 4. Section 5 concludes the paper.

## TRANSCODING ARCHITECTURE

Pixel domain transcoding architecture (Xie *et al*., 2002; Vetro *et al*., 2003; Ahmad *et al*., 2005; Lefol *et al*., 2006; Qian *et al*., 2006) is adopted in this paper. A conventional pixel domain downscaling transcoder is simply composed by cascading an H.264 re-encoder after a whole MPEG-2 decoder and a downscaler. The primary modules of the MPEG-2 decoder and the downscaler, called as "downscaling decoder" hereafter, adopted by the cascaded transcoder, are shown in Fig.1a. The input residual DCT coefficients first undergo the inverse DCT module (iDCT), which produces the MB residual. An MB is reconstructed by adding a referenced MB pointed by the motion vector (MV) in a previous frame to the MB residual. This process is called as motion compensation ($M^f$) where the superscript "f" means motion compensation at full




U: up-sampling; D: down-sampling; TRUNC: motion vector truncation; iDCT: inverse discrete cosine transform
$M^f$: motion compensation with full resolution; $M^r$: motion compensation with reduced resolution; INT: interpolation

**Fig.1 MPEG-2 downscaling decoders. (a) Full motion compensation; (b) Reduced motion compensation; (c) Reduced motion compensation with full MV; (d) Smart motion compensation with full MV**

resolution. Then the reconstructed MB is buffered and downscaled (D). Two buffers, FB1 and FB2 in Fig.1a, are needed to store the decoded frame at full resolution for the next frame reconstruction and the downscaled frame with a quarter size of the full image for re-encoder, respectively. The main complexity of decoding and downscaling processes depends on the operations in iDCT, $M^f$ and D modules and memory accessing. As shown in Fig.1a, both iDCT and $M^f$ work at full resolution and FB1 needs to store the image at full size. In the following segments we focus on reducing the complexity of MPEG-2 decoding and downscaling processes.

To reduce the complexity of the downscaling decoder, the simplest way is to conduct motion compensation in the reduced resolution ($M^r$) shown in Fig.1b. In this scheme, the downscaling stage is accomplished before motion compensation. Three quarters of the additions for motion compensation and memory space are saved. An additional module TRUNC is employed to truncate the input MV proportionally to match the image size downscaling. Although huge complexity is reduced by this scheme, it introduces unacceptable quality degradation. The quality degradation originates from the mismatch error between the predictive and residual components due to the noncommutative property of motion compensation and downscaling (Yin *et al*., 2002; Shen, 2005).

To compensate the quality degradation induced by the reduced motion compensation granularity, another reduced resolution motion compensation (also denoted as $M^r$) method is first proposed by Ng *et al*.(1993). The decoder with this scheme is show in Fig.1c. In this scheme, full motion information is employed for the reduced resolution image reconstruction. As shown in Fig.1c, an up-sampling module (U) and a downscaling module (D) are inserted into the motion compensation loop. This scheme is improved smartly by Sun (1993)'s replacing the D and U modules with an interpolating module (INT). With the INT, one 8×8 block can be interpolated into sixteen 8×8 blocks. The selection of the referenced 8×8 block is decided using the two least significant bits (LSB) of the input MV. The INT module is designed optimally with regard to the D module in (Vetro and Sun, 1998). The decoder with the improved motion compensation scheme is shown in Fig.1d. The D and iDCT modules are emerged into a single module since the computation complexity of the D and iDCT can be reduced significantly by the merging (Merhav and Bhaskaran, 1996). Excluding the additional complexity induced by INT, the memory requirement and addition operations of the decoder in Fig.1d are the same as the one in Fig.1b. As the moderate video quality is preserved with low complexity, the decoder in Fig.1d is adopted in our transcoder.

With the smart downscaling decoder, the transcoding architecture on which our fast mode decision algorithm is based is diagrammed in Fig.2. Two modules of "edge direction detection" and "MV scaling & mode decision" are added in the transcoder. With these two modules, the coding modes for the new MB can be decided without motion estimation. Huge complexity is reduced in the H.264 re-encoder.

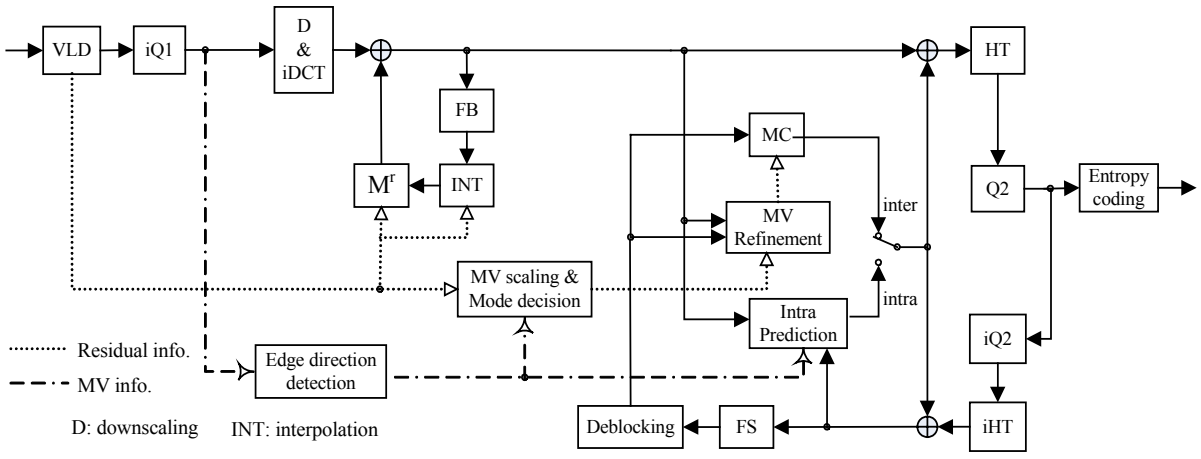



**Fig.2 MPEG-2 to H.264 downscaling transcoding**

## FAST MODE DECISION

Each input MB in the pre-coded MPEG-2 stream is downscaled and then decoded into one 8×8 block. Four such neighboring 8×8 blocks are combined into one new MB in the H.264 re-encoder. As intra coding mode for an inter MB is allowed in MPEG-2, there are three combinations for the four neighboring pre-coded MBs: (1) all intra coded; (2) all inter coded; (3) hybrid by intra and inter. According to the coding types of the four corresponding MBs in the pre-coded stream, the coding mode for the new MB is decided as follows: (1) Coding the new MB as an intra MB if all four corresponding pre-coded MBs are intra coded; (2) Coding the new MB as an inter MB if all four corresponding pre-coded MBs are inter coded; (3) Conducting both intra prediction and inter prediction and selecting the one with minimum rate-distortion (RD) cost. In the H.264 re-encoder, the four scaled 8×8 blocks can be merged into larger blocks: 16×8, 8×16 or 16×16, or split into smaller blocks: 8×4, 4×8 or 4×4. The merging and the splitting procedures for one new MB are illustrated in Fig.3. To match the spatial resolution downscaling, the motion vector of each pre-coded MB is also truncated. As half-pixel resolution motion vector is allowed in MPEG-2 and sub-pixel resolution in H.264, the value of the truncated MV equals to the input MV. As shown in Fig.1d, the two LSBs are employed to decide how to interpolate the referenced block. In our merging scheme, we exploit the four input vectors to decide whether to merge the four 8×8 blocks. If the four MVs satisfy the merging condition, a larger block size is selected for this new MB. Otherwise, the edge direction of each 8×8 block residual is calculated and the block is split according to the edge pattern.

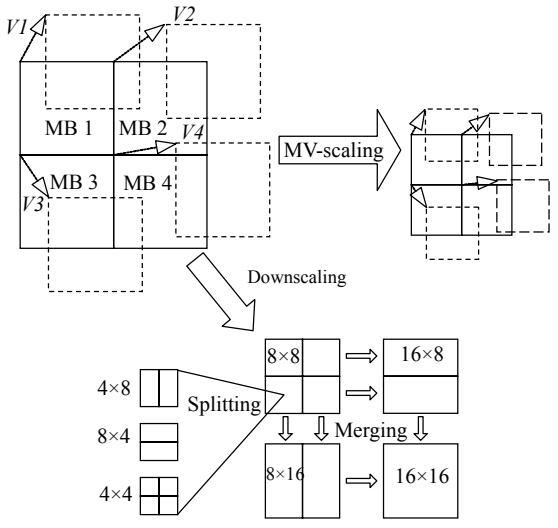



**Fig.3 MB and MV downscaling and block size selection**

### Merging scheme

The following constraint is used to decide whether two 8×8 blocks can be merged into a larger one.

$$Dist(V1, V2) = |V1\_x - V2\_x| + |V1\_y - V2\_y| \le TH, \quad (1)$$

where $Dist(V1, V2)$ is the distance between $V1$ and $V2$. $V1\_x$ and $V1\_y$ are the $x$ and $y$ direction values of $V1$, respectively. $TH$ is the threshold which is set to 3 in the experiences. When the distance between the MVs of two neighboring 8×8 blocks is not larger than 3, these two blocks are merged into a larger block. The merging process is expressed as follows:

(i) If $Dist(V1,V2)<=TH$ && $Dist(V3,V4)<=TH$, 16×8 block size is selected for the MB;

(ii) If $Dist(V1,V3)<=TH$ && $Dist(V2,V4)<=TH$, 8×16 block size is selected for the MB;

(iii) If both (i) and (ii) are true, 16×16 block size is selected for the MB; Else if both (i) and (ii) are not true, splitting detection is conducted for each 8×8 block.

If the merging condition is satisfied, a new MV for the larger block is simply generated by averaging the included MV (Youn *et al*., 1999). This MV as well as the median predicted MV defined in H.264 (JVT, 2003) is used as the prediction MVs. The point which has less RD cost is selected as the centre for the motion re-estimation. Then the mode decision ends. If the merging condition is not satisfied, a splitting detection is conducted.

### Residual edge pattern analysis

To decide whether to split an 8×8 block into smaller blocks, the relative motion information in the residual block is analyzed. Employing the principle that if the motion is uniform, the direction of the object movement is perpendicular to the dominant texture direction in the residual image (Seferidis and Ghanbari, 1994; Wang *et al*., 2007), we can extract the relative motion direction in the block by analyzing the edge direction in this 8×8 block residual. As each

MB is downscaled into an 8×8 block, we use the edge direction in the decoded MB residual to replace the edge direction in the 8×8 block residual.

Deducing from a simple pixel-domain edge detection algorithm (Shen and Sethi, 1996), we extract a dominant edge direction in the MB residual from several DCT coefficients. The MB residual is first divided into four 8×8 blocks. Then, each 8×8 block is further divided into four 4×4 blocks as Fig.4.

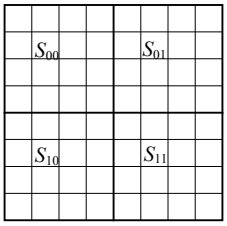



**Fig.4 Partition of an 8×8 block**

Let $S_{00}$, $S_{01}$, $S_{10}$ and $S_{11}$ denote the average intensity of each 4×4 block.

$$S_{uv}=\frac{1}{16}\sum_{i=0}^{3}\sum_{j=0}^{3}p(4u+i,4v+j),\quad u,v=0,1, \tag{2}$$

where $p(k,l)$ is the intensity value of each pixel. $k$, $l$= 0,1,…,7 are the vertical and horizontal indexes. Two edge feature parameters, vertical edge parameter $V$ and horizontal edge parameter $H$, are introduced.

$$V=\left\lfloor\frac{(S_{00}+S_{10})-(S_{01}+S_{11})}{S}\right\rfloor, \tag{3}$$

$$H=\left\lfloor\frac{(S_{00}+S_{01})-(S_{10}+S_{11})}{S}\right\rfloor, \tag{4}$$

where $\lfloor\rfloor$ represents the floor function. The scaling factor $S$ equals 4 times of the quantization step, *QPstep*, in the H.264 re-encoder. Employing the property of DCT, $V$ and $H$ can be calculated in DCT domain (Chang and Kang, 2005).

$$V=\left\lfloor\frac{0.45X(1,0)-0.16X(3,0)+0.1X(5,0)-0.09X(7,0)}{2\cdot QPstep}\right\rfloor, \tag{5}$$

$$H=\left\lfloor\frac{0.45X(0,1)-0.16X(0,3)+0.1X(0,5)-0.09X(0,7)}{2\cdot QPstep}\right\rfloor, \tag{6}$$

where $X(k, l)$ is the DCT coefficient after inverse quantization in the MPEG-2 decoder.

According to $H$ and $V$, each 8×8 block can be classified into one of the four categories as tabulated in Table 1. Through synthesizing the edge directions of the four 8×8 blocks, we classify each 16×16 block as follows:

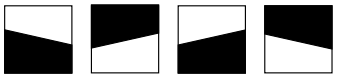
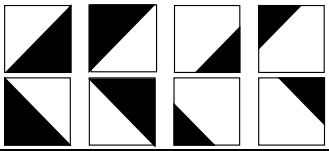
(1) No obvious edge. If all the 8×8 blocks belong to the No Obvious Edge;

(2) Vertical dominant direction edge. If the left two 8×8 blocks or the right two 8×8 blocks or only one 8×8 block have the Vertical Dominant Edge;

(3) Horizontal dominant direction edge. If the above two 8×8 blocks or the below two 8×8 blocks or only one 8×8 block have the Horizontal Dominant Edge;

(4) Diagonal direction edge for all other combinations.

**Table 1 Edge direction categories of one 8×8 block**

| Measure | Edge direction | |
|---|---|---|
| $\|H\|=\|V\|=0$ | No obvious edge |  |
| $0<\|V\|<\|H\|$ | Vertical dominant edge |  |
| $0<\|H\|<\|V\|$ | Horizontal dominant edge |  |
| $\|H\|=\|V\|>0$ | Diagonal edge |  |

## 8×8 block splitting scheme

The dominant edge direction in the 8×8 block residual is represented by the corresponding MB residual direction. We segment each 8×8 block along the edge direction in the 8×8 block residual as follows:

(1) 8×8 size for no obvious edge;

(2) 8×4 for horizontal dominant direction edge;

(3) 4×8 for vertical dominant direction edge;

(4) 4×4 for diagonal direction edge.

## EXPERIMENTAL RESULTS AND DISCUSSION

A transcoder is constructed simply by cascading a JM8.6 encoder after an MPEG-2 decoder and a

downscaler. The proposed transcoder and the reference transcoder are simulated for RD performance comparison. Our proposed fast mode decision is employed in the proposed transcoder. The reference transcoder decides the coding mode by conducting seven variable block size motion estimations and selecting the block size with the smallest RD cost. The first 100 frames of Akiyo, Paris, and Foreman sequences at CIF (352×288) resolution are first upsamled to 4CIF size (704×576) by a bilinear interpolation filter and encoded by an MPEG-2 encoder at 6 Mbps with GOP size of 15 and "IPPP" structure. The Stefan sequene is coded at 8 Mbps. And then they are transcoded to H.264 format with "IPPP…" structure with different *QP*. The re-encoder uses the CAVLC entropy coder and one reference frame. The motion refinement search range is ±3 around the centre point. The RDO is off because we focus on low complexity applications. The RD curves of the proposed transcoder and the reference transcoder are shown in Fig.5. The peak signal-to-noise ratio (PSNR) is between the downscaled output image and the re-encoder reconstructed image.

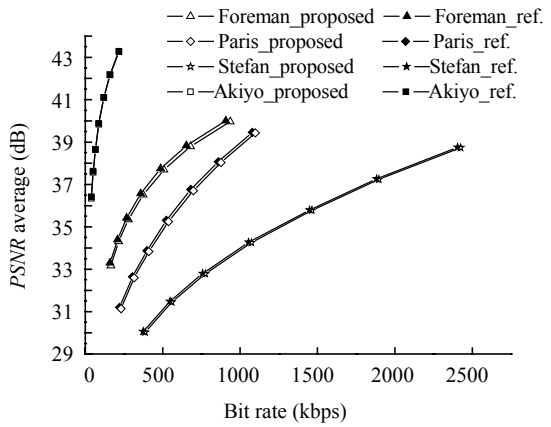



**Fig.5 Rate-Distortion curves of the proposed transcoder and the reference transcoder**

From Fig.5, we can see that the RD curves by the proposed transcoder are close to those of the reference transcoder. Series of other sequences at different bitrates are also tested. There is about 1.9% for average bitrate increase and no more than 3.8% for the worst case with the same *PSNR* by the proposed transcoder comparing with the reference one. Table 2 tabulates the performance comparison between the proposed fast mode decision and the reference mode decision at *QP*=28. In the table, "Δ*PSNR* (dB)" and "ΔBits (%)" represent the *PSNR* change and the bit rate change in percentage, respectively, of the proposed mode decision comparing with the reference one. "speedup (times)" represents the speedup times calculated by dividing the re-encoding time consumed by the reference transcoder with the re-encoding time consumed by the proposed one. It can be seen from Fig.5 and Table 2 that with little degradation of RD performance, the proposed algorithm speeds up the re-encoder dramatically.

**Table 2 Experimental results of the proposed transcoder vs. the reference one at *QP*=28**

| Sequence | Δ*PSNR* (dB) | ΔRate (%) | Speedup (times) |
|---|---|---|---|
| Stefan | −0.023 | +1.8 | 4.4 |
| Foreman | −0.063 | +3.8 | 4.8 |
| Akiyo | −0.009 | +0.0 | 5.6 |
| Paris | −0.056 | +2.1 | 5.1 |

As we know, the mode decision in the reference encoder is designed to select the mode with the maximum quality and minimum bit rate. While our proposed algorithm is designed to partition the MB fast according to the movement characteristic of the MB. Without calculating the seven RD costs by seven times motion estimations, the coding mode selected by our algorithm may not have the least RD cost. So the RD performance is a little lower than the reference transcoder. But with little loss of RD performance, the proposed algorithm significantly reduces the computation complexity. There is more than 4.4 times speedup. It is worthwhile to sacrifice little RD performance to reduce huge transcoding complexity. This is necessary for computation constraint situations.

## CONCLUSION

In this paper, we propose a fast mode decision algorithm for a video downscaling transcoder from MPEG-2 to H.264. The efficient transcoding architecture includes a smart downscaling decoder and the fast mode decision re-encoder. Much memory space and a great number of operations are saved by the downscaling decoder. By exploiting the motion information carried in motion vectors as well as the residual signal, the proposed fast mode decision al-

gorithm decides the coding block size for one new MB without motion estimation. Huge complexity is reduced with little compression performance degradation. With the proposed architecture, the transcoding task even can be realized on a single embedded processor such as DAVINCI for real-time applications.

## References


Ahmad, I., Wei, X.h., Sun, Y., Zhang, Y.Q., 2005. Video transcoding: an overview of various techniques and research issues. *IEEE Trans. on Multimedia*, **7**(5):793-804. [doi:10.1109/TMM.2005.854472]

Bu, J.J., Mo, L.J., Chen, C., Yang, Z., 2006. Fast mode decision algorithm for spatial resolutions down-scaling transcoding to H.264. *J. Zhejiang University Sci. A*, **7**(Suppl. 1):70-75. [doi:10.1631/jzus.2006.AS0070]

Chang, A., Au, O.C., Yeung Y.M., 2003. A Novel Approach to Fast Multi-block Motion Estimation for H.264 Video Coding. Proc. Int. Conf. Multimedia and Expo, **1**:105-108.

Chang, H.S., Kang, K., 2005. A compressed domain scheme for classifying block edge pattern. *IEEE Trans. on Image Processing*, **14**(2):145-151. [doi:10.1109/TIP.2004.840706]

JVT, 2003. Joint Video Team of ITU-T and ISO/IEC JTC 1. Advanced Video Coding: ITU-T Rec. H.264 and ISO/IEC14496-10, Version 1.

Kucukgoz, M., Sun, M.T., 2004. Early-Stop and Motion Vector Reuse for MPEG-2 to H.264 Transcoding. Proc. SPIE, **5308**:932-936. [doi:10.1117/12.522040]

Lefol, D., Bull, D., Canagarajah, N., 2006. Performance evaluation of transcoding algorithms for H.264. *IEEE Trans. on Consum. Electr.*, **52**(1):215-222.

Lu, X., Tourapis, A., Yin, P., Boyce, J., 2005. Fast Mode Decision and Motion Estimation for H.264 with a Focus on MPEG-2/H.264 Transcoding. Proc. IEEE Int. Symp. Circuits Syst. Kobe, Japan.

Merhav, N., Bhaskaran, V., 1996. A Fast Algorithm of DCT-Domain Image Downscaling. Proc. ICASSP. Atlanta, GA, **2**:2307-2310.

MPEG, 1993. MPEG-2 Test Modal 5. ISO/IEC JTC1/SC29/SG11, N0400.

Ng, S.B., Thomson Consumer Electronics, 1993. Low Resolution HDTV Receivers. US Patent 5 262 854.

Qian, T.J., Sun, J., Li, D., Yang, X.K., Wang, J., 2006. Transform domain transcoding from MPEG-2 to H.264 with interpolation drift-error compensation. *IEEE Trans. on Circuits Syst. Video Technol.*, **16**(4):523-534. [doi:10.1109/TCSVT.2006.871392]

Seferidis, V.E., Ghanbari, M., 1994. Adaptive motion estimation based on texture analysis. *IEEE Trans. on Commun.*, **42**:1277-1287. [doi:10.1109/TCOMM.1994.580237]

Shen, B., 2005. Submacroblock motion compensation for fast down-scale transcoding of compressed video. *IEEE Trans. on Circuits Syst. Video Technol.*, **15**(10):1291-1302. [doi:10.1109/TCSVT.2005.854216]

Shen, B., Sethi, L.K., 1996. Direct Feature Extraction from Compressed Images. Proc. SPIE, **2670**:404-414. [doi:10.1117/12.234779]

Sun, H.F., 1993. Hierarchical decoder for MPEG compressed video data. *IEEE Trans. on Consum. Electr.*, **39**(3):559-564. [doi:10.1109/30.234635]

Vetro, A., Sun, H.F., 1998. On the motion compensation within a down-conversion decoder. *J. Electr. Imaging*, **7**:616-617. [doi:10.1117/1.482615]

Vetro, A., Christopoulos, C., Sun, H.F., 2003. Video transcoding architectures and techniques: an overview. *IEEE Signal Processing Magazine*, **20**:18-29. [doi:10.1109/MSP.2003.1184336]

Wang, X.W., Sun, J., Liu, Y.Q., Li, R.J., 2007. Fast Mode Decision for H.264 Video Encoder Based on MB Motion Characteristic. IEEE Int. Conf. Multimedia and Expo, p.372-375.

Xie, R., Liu, J.L., Wang, X.G., 2002. Efficient MPEG-2 to MPEG-4 Compressed Video Transcoding. Proc. SPIE, **4671**:192-201. [doi:10.1117/12.453058]

Xin, J., Sun, M.T., Choi, B.S., Chun, K.W., 2002. An HDTV-to-SDTV spatial transcoder. *IEEE Trans. on Circuits Syst. Video Technol.*, **12**(11):998-1008. [doi:10.1109/TCSVT.2002.805508]

Yin, P., Vetro, A., Liu, B., Sun, H.F., 2002. Drift compensation for reduced spatial resolution transcoding. *IEEE Trans. on Circuits Syst. Video Technol.*, **12**(11):1009-1020. [doi:10.1109/TCSVT.2002.805509]

Youn, J., Sun, M.T., Lin, C.W., 1999. Motion vector refinement for high-performance transcoding. *IEEE Trans. on Multimedia*, **1**(1):30-40. [doi:10.1109/6046.748169]

Zhou, Z., Sun, S., Lei, S., Sun, M.T., 2005. Motion Information and Coding Mode Reuse for MPEG-2 to H.264 Transcoding. IEEE Int Symp. on Circuits Syst., p.1230-1233. [doi:10.1109/ISCAS.2005.1464816]