Chapter 2
Conventional Transcoder

Abstract The H.264/AVC video compression standard offers over two times higher compression rate than what the MPEG-2 can offer, while maintaining the same visual quality. Additionally, H.264/AVC offers a substantial network adaptation capability, which is essential for many network dependent video applications, e.g. delivering Digital TV (DTV) programs over the Internet and video conferencing. For that reason, the relativity new H.264/AVC standard has been pushed to the top of the candidate stack for coding Digital TV programs at a bit-rate lower than the MPEG-2's bit-rate. However, for over a decade now, most of the video resources have been encoded in MPEG-2 format. As a result, the heterogeneous transcoding solution, MPEG-2 to H.264/AVC, becomes necessary for distributing video resources specially over the Internet. In the following section, we will discuss the current employed MPEG-2 to H.264/AVC transcoding solution in the video industry.

2.1 The MPEG-2 to H.264/AVC Conventional Cascaded Transcoding

The simplest transcoding method is the one that converts the MPEG-2 video formats to the H.264/AVC bit-stream by means of cascaded transcoding. Also, it is well known in the video industry with its commercial name, i.e. "Conventional Cascaded Transcoder (CCT)". Simply stated, the idea behind such transcoder is based on sequential decoding/ encoding processes. At first, the MPEG-2 video file is completely decoded using a traditional MPEG-2 decoder. After that, the uncompressed video frames, which also entitled either "Raw Video", uncompressed video or Common Intermediate Format (CIF), are encoded by an independent H.264/AVC encoder to produce the H.264/AVC bit-stream.

As illustrated in Fig. 2.1, the MPEG-2 bit-stream is decoded by the MPEG-2 decoder to produce the reconstructed raw video data. Then, the uncompressed video frames are compressed by the H.264/AVC encoder to generate the H.264/AVC compressed video bit-stream.

T. Elarabi et al., *Real-Time Heterogeneous Video Transcoding for Low-Power Applications*, DOI: 10.1007/978-3-319-06071-2_2,

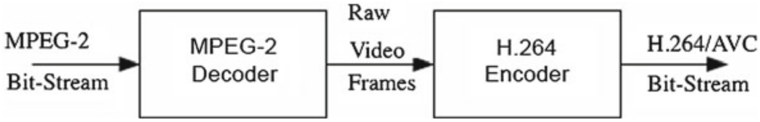

Fig. 2.1 The conventional cascaded MPEG-2 to H.264 transcoder

In fact, in terms of hardware and software implementation, the cascaded transcoding architecture is implementation friendly. Also, it can reduce the lab-to-market time gap, besides, the engineering cost [8]. Furthermore, it has been widely used in the market because of its conceptually straight forward design structure and relativity easily implementation. However, the cumulated computational complexity is causing a serious problem for the cascaded transcoding solution.

Indeed, the computational complexity of the H.264/AVC decoder is ten times or more than the computational complexity of the MPEG-2 encoder. Accordingly, most of the computational complexity is localized in the H.264/AVC encoder side of the conventional transcoder. In contrast to the MPEG-2 codec, the H.264/AVC standard codec permits not only Inter frame coding to remove the frames temporal redundancy, but also Intra frame coding to eliminate the frames' spatial redundancy. In particular, both the Intra macro-block mode decision and the Intra macro-bock direction prediction processes are the main computational demanding stages in the Intra frame coding process [9].

Furthermore, the H.264/AVC has two prediction mode candidates for the macro-block's Luminance (Luma) component. On the other hand, it has only one mode for each of the macro-block's Chrominance (Chroma) components. Accordingly, the mode selection process is done only on the macro-block's Luma component. Briefly, the two Luma Intra prediction modes are the Intra 16 $\times$ 16 mode or the Intra 4 $\times$ 4 mode in the H.264/AVC baseline profile [7].

On one hand, the Intra Luma 16 $\times$ 16 mode executes the direction prediction process on the whole 16 $\times$ 16 macro-block as one unit. In contrast, the Intra Luma 4 $\times$ 4 mode performs the direction prediction process on each of the sixteen 4 $\times$ 4 sub macro-blocks within the current 16 $\times$ 16 Luma macro-block. Furthermore, there are four reconstruction direction candidates for the Intra Luma 16 $\times$ 16 mode, as shown in Fig. 2.2. Whereas, there are nine direction prediction candidates for the Intra Luma 4 $\times$ 4 mode, as shown in Fig. 2.3.

On the other hand, there is only one Intra 8 $\times$ 8 prediction mode for the Chroma macro-block components [10]. It has four different direction prediction candidates, which are similar to Intra 16 $\times$ 16 Luma direction prediction candidates except that the numbers of the modes are using different order as shown in Table 2.1 [7].

Furthermore, in the conventional cascaded pixel domain transcoder, the Rate Distortion (RD) is used to evaluate and further decide the reconstruction direction for all of the macro-block's components. In details, the RD for all the four Intra Luma 16 $\times$ 16 direction prediction candidates, all the nine Intra Luma 4 $\times$ 4 direction prediction candidates and the four Chroma direction prediction candidates for

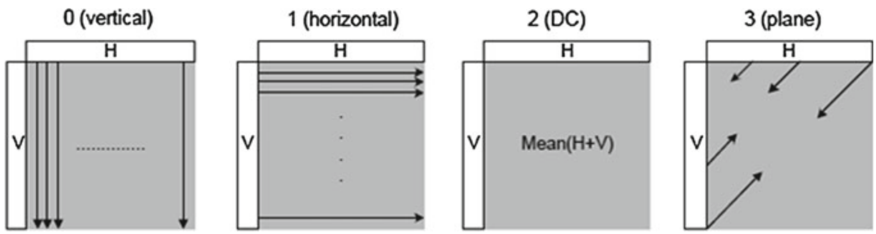

Fig. 2.2 The standard direction candidates of the 16 × 16 mode

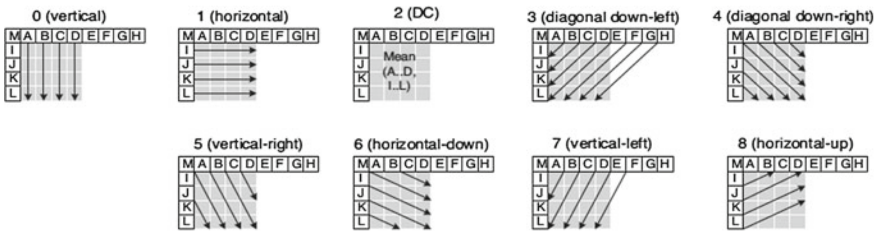

Fig. 2.3 The standard direction candidates of the 4 × 4 mode

Table 2.1 The Luma versus the Chroma direction candidates' number

Prediction direction	Luma number	Chroma number
DC	2	0
Horizontal (H)	1	1
Vertical (V)	0	2
Plane (P)	3	3

each macro-block must be computed. In other words, in order to make a mode and reconstruction direction decision for each of the macro-block's components, the cost is evaluated for all the macro-block direction prediction candidates based on the RD theory [11] as in Eqs. (2.1) and (2.2).

$$RD_{cost} = D + LM * BR \tag{2.1}$$

$$LM = 0.85 * 2^{(QP-12)/3} \tag{2.2}$$

where D and BR represent the distortion and bit-rate for a given prediction direction, respectively. QP and LM are the quantization parameter and Lagrangian multiplier, respectively. Also, in order to compute BR and D, the actual coding is performed for each 4 × 4 sub macro-block, which is the basic processing unit for transformation and Variable Length Coding (VLC) in the H.264/AVC codecs.

All in all, as shown in Fig. 2.4, the total number of needed Rate Distortion Operations (RDO) for deciding the best direction for each macro-block is 592 RDOs

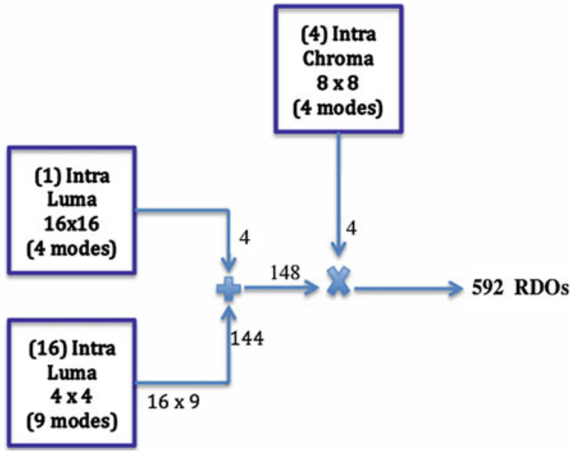

Fig. 2.4 The required RDOs for each standard Intra macro-block

[9]. Such huge computational complexity is the current bottleneck in the MPEG-2 to H.264/AVC conventional cascaded transcoder processes.

In conclusion, the computational complexity of the H.264 encoder part of the conventional cascaded transcoder is proved to be impractically huge. Besides, it requires multiple Digital Signal Processing (DSP) cores, which demand sophisticated parallel processing algorithms [8]. Moreover, this implementation issues are very serious for real-time video communication, e.g. real-time TV broadcasting to H.264/AVC enabled smart phones and video telephony [8].

Additionally, the H.264/AVC video decoders in the mobile devices periodically use the Intra frame prediction to quickly recover from channel error propagation at rate of one Intra frame per every five frames [9]. Whereas, the Intra prediction built in processes is the cause for most of the H.264/AVC computational complications. Therefore, the main scope of the thesis is focusing on enhancing the H.264/AVC Intra prediction side of the transcoder. However, there are many research efforts that have been introduced to reduce the computational complexity of the conventional transcoders. Also, many researchers have handled the Intra prediction complexity issue at the H.264/AVC encoder part of the CCT transcoder. As a result of such research efforts, several enhanced transcoding algorithms have been recently introduced.

For example, a very interesting work was introduced in [8]. In this research work, the authors proposed a mode skipping rule for the H.264/AVC encoder's Intra prediction process in the MPEG-2 to H.264/AVC transcoder. Their idea is based on an experimental analysis results, which show that the DCT energy trend in the MPEG-2 bit-stream has a strong correlation with the Intra mode selection in the H.264/AVC encoder. In Chap. 3, we discuss their Intra prediction technique, which is entitled "Kim's algorithm" in some details. Additionally, in Chap. 5, we compared

Table 2.2 Brief comparison between MPEG-2 and H.264/AVC

	MPEG-2	H.264/AVC
Transformation	DCT	DCT
Inter prediction	Yes	Yes
Intra prediction	No	Yes
MB partition	No	Yes
Block sizes	16 × 16, 8	16 × 16, 8; 8 × 16, 8, 4; 4 × 8, 4
Reference frames	1 or 2	Up to 16

Kim's Intra prediction algorithm with our proposed Full-Search Free Intra prediction algorithm, which we introduced in Sect. 5.2 of the same chapter.

In addition, the authors of [9] proposed an Intra mode decision method for the MPEG-2 to H.264/AVC transcoder, which is based on a spatial activity analysis for the DCT coefficients in the MPEG-2 decoder part of the transcoder. Chapter 3, discusses Yoo's Intra prediction algorithm in details. Also, Chap. 5 includes a comparison between Yoo's Intra algorithm and our proposed Full-Search Free Intra prediction algorithm. Furthermore, analyzing the similarities and the differences between both the H.264/AVC and MPEG-2 video compression standards is a good start in order to design an efficient transcoder.

2.2 The MPEG-2 versus H.264/AVC Video Standards

In fact, many transcoding challenges have been arisen due to the differences between the MPEG-2 and the H.264/AVC standard formates. These differences can be summarized, but not limited to the fact that the MPEG-2 standard does not use any Intra compression technique to remove the Intra frames' spatial redundancy. However, the H.264/AVC standard encoder uses the Intra prediction technique quite frequently for compressing the reference frames (I-frames). In brief, Table 2.2 summarizes the main differences between the MPEG-2 and the H.264/AVC standards.

In fact, the H.264/AVC's Inter prediction uses different block sizes than the MPEG-2's motion estimation's block sizes. As the MPEG-2 standard is using 16 × 16 and 8 × 8 macro-block sizes, the H.264/AVC has a larger set of block sizes: 16 × 16, 16 × 8, 8 × 16, 8 × 8, 8 × 4, 4 × 8, and 4 × 4. Also, MPEG-2 uses only 1/2-pixel accuracy for motion estimation and compensation. In contrast, H.264/AVC is using up to 1/4-pixel. Similarly, the motion vectors used in MPEG-2 are not constrained to the current frame boundaries. But, it uses only one reference frame for P-type frames that is a previously encoded frame and two reference frames for B-type frames, which are future frames [12]. On the other hand, the H.264/AVC may reference up to eight P-Frames and/or eight B-Frames.

All in all, there are many critical differences between the MPEG-2 and the H.264/AVC video compression standards. As a result, many challenges are facing the industrial demand for an efficient real-time MPEG-2 to H.264/AVC transcoders. In the scope of this book, the Intra prediction and the block sizes are the most concerning differences because of their dramatically high run-time and computational resources consumption. Also, we are focusing on the DCT transformation in both standards as the most concerning similarity between standards from our research point of view.

http://www.springer.com/978-3-319-06070-5

Real-Time Heterogeneous Video Transcoding for
Low-Power Applications
Elarabi, T.; Abdelgawad, A.; Bayoumi, M.
2014, XII, 84 p. 73 illus., 44 illus. in color., Hardcover
ISBN: 978-3-319-06070-5