Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Hot topic:

Review of the current and future technologies for video compression*

Lu YU†[1,2], Jian-peng WANG[1,2]

(*[1]Institute of Information and Communication Engineering, Zhejiang University, Hangzhou 310027, China*)
(*[2]Key Laboratory of Integrated Information Network Technology of Zhejiang Province, Hangzhou 310027, China*)
†E-mail: yul@zju.edu.cn

Received Nov. 9, 2009; Revision accepted Nov. 16, 2009; Crosschecked Nov. 9, 2009

Abstract: Many important developments in video compression technologies have occurred during the past two decades. The block-based discrete cosine transform with motion compensation hybrid coding scheme has been widely employed by most available video coding standards, notably the ITU-T H.26*x* and ISO/IEC MPEG-*x* families and video part of China audio video coding standard (AVS). The objective of this paper is to provide a review of the developments of the four basic building blocks of hybrid coding scheme, namely predictive coding, transform coding, quantization and entropy coding, and give theoretical analyses and summaries of the technological advancements. We further analyze the development trends and perspectives of video compression, highlighting problems and research directions.

Key words: Video compression, Predictive coding, Transform coding, Quantization, Entropy coding, Theoretical analysis
doi:10.1631/jzus.C0910684 **Document code:** A **CLC number:** TN919.8

1 Introduction

With the advent of the information era, the past two decades have seen many developments in multimedia representation and communications. Digital video service, which lies at the core of information society, has become an integral part of a wide range of industries—from telecommunications to broadcasting and entertainment to consumer electronics. The amount of uncompressed video data, however, is too large for the limited transmission bandwidth or storage capacities. A digital versatile disk (DVD), for example can store only a few seconds of raw video at the resolution and frame rate of television-quality. Therefore, video compression is an essential component of digital video services.

Generally speaking, video compression refers to reducing the quantity of data used to represent digital video images while preserving an acceptable video quality, and is a combination of spatial image compression and temporal motion compensation. The

Introducing editorial board member: Professor Lu Yu, the first author of this invited review, is an editorial board member of *JZUS-C*. Her research areas include video coding, multimedia communication, and relative ASIC design. She is a principal investigator of a number of national R&D projects and inventor of more than 40 granted and pending patents, and acts as the chair of the video subgroup of the working group of audio video coding standard (AVS) of China. She also served as general chair of the 15th International Workshop on Packet Video in 2006 and Technical Committee member, International Steering Committee member, and session chair of international conferences. Now she is a member of Technical Committee of Visual Signal Processing and Communication of IEEE Circuits and Systems Society, an area editor of *EURASIP Journal Signal Processing: Image Communication*.

Dr. Lu YU

---

* Project (No. 2009CB320903) supported by the National Basic Research Program (973) of China

search for efficient video compression techniques has dominated much of the research on video coding, a focus which gave rise to H.261 (ITU-T, 1993), the first milestone in the development of video compression standard. The research on the basic coding tools involved in video compression actually started as early as in the 1950s and 1960s with spatial differential pulse code modulation (DPCM) coding of images. In the 1970s, transform coding techniques were investigated, and the well-known block-based discrete cosine transform (DCT) was introduced by Ahmed *et al.* (1974). Motion compensated prediction error coding also started in the 1970s and gradually matured into practical technology (Jain and Jain, 1981; Girod, 1987). The combination of temporal block-based motion compensated prediction and block-based DCT coding formed the basic hybrid block-based motion compensation/DCT (MC/DCT) systems.

MC/DCT hybrid coding technology provides the key element of most available video coding standards, notably the ITU-T H.26*x* and ISO/IEC MPEG-*x* families. The hybrid coding scheme generally consists of four basic building blocks, within the dashed rectangle as depicted in Fig. 1, namely predictive coding, transform coding, quantization and entropy coding. With the fascinating development in the video coding standardization work, from H.261 to H.264 and from MPEG-1 to MPEG-4, in the past two decades, the techniques involved in each of these building blocks have become more and more matured and sophisticated, from simple DC prediction in frequency domain to multi-directional intra prediction in pixel domain, from forward only prediction to backward- and bi-prediction for motion compensation, from sole 16×16 MC block size to up to 32×32 and down to 4×4 MC block size, from 1-pel accuracy to 1/4-pel and even to 1/8-pel accuracy for MC, from discrete cosine transform to the integer cosine transform and to mode-dependent Karhunen-Loève transform (KLT), and from 2D variable-length coding (VLC) to 3D VLC and context adaptive VLC/context adaptive binary arithmetic coding (CABAC) for entropy coding. Normally, higher compression gains can be achieved with more complex algorithms.

Active research is being carried out to make video compression algorithms more efficient. The Video Coding Experts Group (VCEG) of ITU-T is trying to seek proposals and information that have promising coding performance to justify the step from H.264 to H.265. To better evaluate the contributions and retain progress, the key technical area (KTA) has been developed as the software platform, using JM11 as the baseline and continuously integrating promising tools, and significant gains have been achieved compared to H.264/AVC.

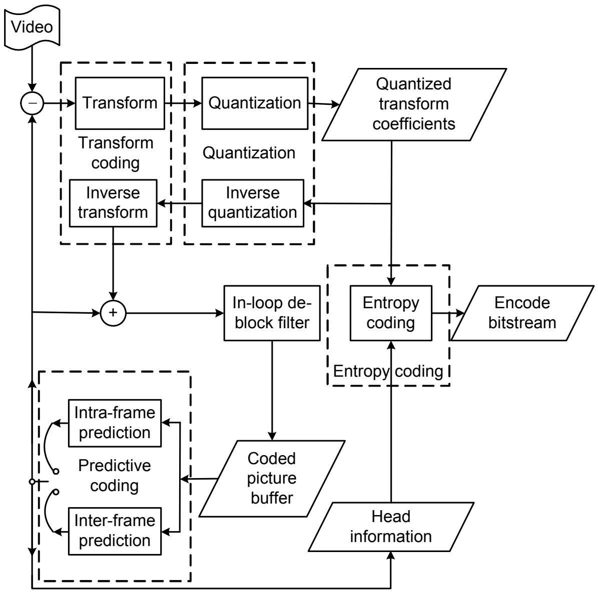

Fig. 1 Diagram of the architecture of today's hybrid video coding

In this paper, we will focus on the block-based MC/DCT hybrid coding schemes, give a review of the technologies involved in predictive coding, transform coding, quantization and entropy coding, and based on this analyze the development trends and perspectives of video compression, highlighting problems and research directions. The rest of the paper is organized as follows. Section 2 gives a brief review of the foundational theories applied to video compression. Sections 3–6 are dedicated to the four basic building blocks, respectively, and theoretical analyses and summaries are made according to their respective development contexts. Section 7 analyzes the future trends of video compression and provides conclusions.

2 Theoretical foundations

Video compression is an example of the concept of source coding in information theory, and the rate distortion theory constitutes the basis for video compression. The basic results of rate distortion theory were presented in the two classic papers (Shannon, 1948; 1959), and were refined and embellished in many works over the years. In this section, we will give a brief introduction of basic concepts, theoretical conclusions, and mathematical models applied to video compression.

2.1 Distortion metrics

In order to specify, evaluate and compare video compression strategies, it is necessary to define metrics of distortion between original and compressed video sequences. Visual quality is inherently subjective and is influenced by many factors that make it difficult to obtain a completely accurate measurement of the distortion. Using objective criteria as a distortion metric gives a quantitative, repeatable result, but there is no objective measurement system that can precisely reproduce the subjective experience of human observation of a video display.

The most widely used objective distortion metric in the video compression and video processing systems is the peak signal-to-noise ratio (PSNR), which is calculated based on the mean squared error (MSE) between original and impaired or reconstructed video frame.

Because of the limitations of crude metrics such as PSNR, there has been a lot of recent work on more sophisticated objective measurements that more closely approximate subjective test results. Many different approaches have been proposed (van den Branden Lambrecht and Verscheure, 1996; Tan and Ghanbari, 2000; Wu *et al.*, 2001) but none of these has emerged as a clear alternative to subjective tests or efficient distortion metric.

2.2 Upper bound on discrete entropy

As stated in Cover and Thomas (2003), the upper bound on discrete entropy is given as follows.

Let X be a discrete random variable on the set $\{a_1, a_2, \ldots\}$, with $\Pr(X=a_i)=p_i$, $i\in\mathbb{N}^*$. Then

$$H(p_1, p_2, \cdots) \le \frac{1}{2}\log(2\pi e)\left(\sum_{i=1}^{\infty} p_i i^2 - \left(\sum_{i=1}^{\infty} i p_i\right)^2 + \frac{1}{12}\right),$$

where the discrete entropy

$$H(p_1, p_2, \cdots) = -\sum_{i=1}^{\infty} p_i \log p_i.$$

Moreover, for every permutation σ,

$$H(p_1, p_2, \cdots) \le \frac{1}{2}\log(2\pi e)\left(\sum_{i=1}^{\infty} p_{\sigma(i)} i^2 - \left(\sum_{i=1}^{\infty} i p_{\sigma(i)}\right)^2 + \frac{1}{12}\right).$$

Therefore, a tighter bound will be achieved when the high probabilities are close together, i.e., by the assignment $\ldots, p_5, p_3, p_1, p_2, p_4, \ldots$ for $p_1 \ge p_2 \ge p_3 \ge p_4 \ge p_5 \ge \ldots$.

Without loss of generality, assume $a_i=i$ for $i=1, 2, \ldots$, so that X is an integer-valued discrete random variable, with the variance

$$\mathrm{Var}(X) = \sum_{i=1}^{\infty} p_i i^2 - \left(\sum_{i=1}^{\infty} i p_i\right)^2.$$

Then we can derive

$$H(p_1, p_2, \cdots) \le \frac{1}{2}\log(2\pi e)\left(\mathrm{Var}(X) + \frac{1}{12}\right).$$

This theorem shows that the entropy of an integer-valued discrete source, such as the prediction residuals, is up-bounded by a function of the variance of the source. This means that smaller variance usually leads to smaller entropy, i.e., fewer bits required to represent that source. This theorem explains why so many current algorithms for video compression are dedicated to reducing the variance (or MSE) of the prediction on a theoretical level.

2.3 AR(1) model for video source

AR(1) process (Jayant and Noll, 1984), where AR stands for ‘autoregressive’, also known as ‘first-order Markov process’, is a good model for first-order prediction analysis. Video compression technologies usually use this to model the correlation between pixels in spatial domain or temporal domain.

Based on the AR(1) model, the correlation between pixels in an image or a video frame can be written as $x_{nm} - \rho_{\mathrm{h}}^{|m-i|}\rho_{\mathrm{v}}^{|n-j|}x_{ji} = z_{nmji}$, where ρ_{h} denotes the correlation coefficient of horizontal neighboring pixels, ρ_{v} denotes the correlation coefficient of vertical neighboring pixels, and z denotes a Gaussian white noise. Then the correlation coefficient between x_{nm} and x_{ji} can be derived as $\mathrm{cor}(x_{nm}, x_{ji}) = \rho_{\mathrm{h}}^{|m-i|}\rho_{\mathrm{v}}^{|n-j|}$. In temporal case, the correlation between two colocated pixels from two adjacent frames can be written as $x_t-\rho_t x_{t-1}=z_t$, and then the correlation coefficient between x_t and x_{t-1} can be derived as $\mathrm{cor}(x_t, x_{t-1})=\rho_t$.

The AR(1) model is very important, and the analysis of transform coding and predictive coding will all be based on this model.

3 Predictive coding

Predictive coding is the primary tool utilized in current video compression technologies, and it is very efficient for removing the correlation between pixels in both spatial domain and temporal domain. Pixel values to be coded are predicted from already coded and reconstructed adjacent pixel values, and only small prediction errors are coded. A brief theoretical analysis of a simple but representative implementation of predictive coding is presented below.

Consider the random variable $D=X-Y$, the prediction error between two spatial adjacent pixels, where Y is the pixel for prediction and X is the pixel to be coded. The variance of D can be obtained as $\mathrm{Var}(D)=\mathrm{Var}(X)+\mathrm{Var}(Y)-2\mathrm{Cov}(X, Y)$. According to the AR(1) model described above, X and Y have the same mean and variance (σ^2), and $\mathrm{Var}(D)=2(1-\rho)\sigma^2$, where ρ denotes the correlation coefficient between X and Y. In the case of $\rho>0.9$, which is almost always true for natural images, the variance of prediction errors is largely decreased compared to the variance of the pixels to be coded. This results in significantly reduced bitrates according to the theorem introduced in Section 2.2. In addition, the prediction between X and Y becomes nonsense in the case of $\rho<0.5$.

Predictive coding techniques involved in video coding can be divided into two classes, intra-prediction coding which aims to remove spatial redundancy, and inter-prediction coding which aims to remove temporal redundancy.

3.1 Intra-prediction coding

The earliest research on intra-prediction coding perhaps can be traced back to the 1950s and 1960s when spatial DPCM coding of image emerged. The spatial DPCM scheme was adopted by JPEG (ITU-T and ISO/IEC, 1992) during the 1980s and early 1990s, as the lossless coding mode with the combination of Huffman coder. However, this scheme is unable to provide a high compression ratio and thus has not been found widespread use.

The widely used intra-prediction algorithms are utilized together with block-based transform coding. In early video coding standards, such as H.261, MPEG-1 (ISO/IEC, 1993), and MPEG-2 (ITU-T and ISO/IEC, 1994), DCT is applied to pixel blocks, which generates one DC coefficient and several AC coefficients. Intra-prediction is performed only on the DC coefficients, i.e., the predictor of the current block DC_i is the DC_{i-1} of the previous block. DC prediction removes the high correlation between DC coefficients of adjacent blocks. Actually, correlations also exist between some AC coefficients of adjacent blocks. For this reason, in the second generation video coding standards, such as MPEG-4 (ISO/IEC, 1999) and H.263 (ITU-T, 2000), intra-prediction is performed not only on DC coefficients, but also on the AC coefficients that represent pure horizontal frequency components or pure vertical frequency components, at the expense of transmitting the extra bits for the prediction mode information. Fig. 2 illustrates the intra-prediction modes involved in H.263. Why do the other AC coefficients not provide any prediction? This is because the correlation coefficients between these AC coefficients are usually small, typically less than 0.5.

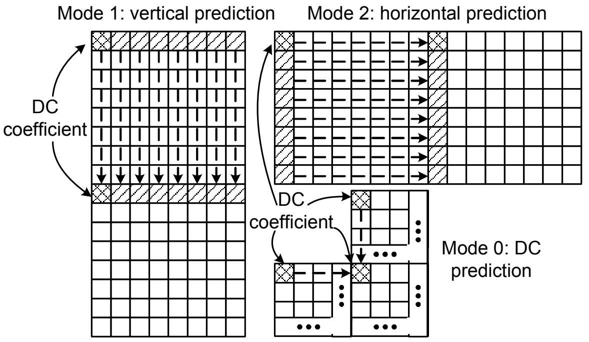

Fig. 2 Modes for H.263 intra-prediction

In the above-mentioned algorithms, intra-prediction is performed in transform domain. In contrast, the algorithms in the latest video coding standards, such as H.264/AVC (Wiegand *et al.*, 2003a) and AVS (developed by the Audio Video Coding Standard Working Group of China) (Yu *et al.*, 2009), perform intra-prediction in pixel domain, followed by a transform operation applied to the prediction errors. These techniques deploy more directional prediction modes to conduct more accurate prediction, and additionally, make predictions between prediction modes of the neighboring blocks to further remove redundancy. After H.264/AVC and AVS, a lot of new intra-prediction techniques have sprung up during recent years, such as bi-directional intra-prediction (Shiodera *et al.*, 2007; Ye and Karczewicz, 2008), intra-prediction based on template matching (Tan *et al.*, 2006; Guo, *et al.*, 2008; Zheng *et al.*, 2008), multidirectional intra-prediction (Tsukuba *et al.*, 2007), and so on. All of these techniques are trying to make more accurate prediction and thus smaller prediction error variance, while at the expense of transmitting more extra mode bits.

Thus far, as intra-prediction techniques develop, coding efficiency has been significantly improved by employing more refined and more flexible intra-prediction modes. Because of the non-stationary statistical properties of video signal, the properties of different regions in a video frame differ greatly, which demands multiple coding modes customized for coding different types of scene statistics to achieve high compression efficiency. In fact, the employment of multiple coding modes is a kind of adaptability in some sense; i.e., for different local regions with different statistics, the encoder adaptively selects the most efficient coding modes to achieve the best coding performance. Therefore, the more efficient coding modes utilized, the better adaptation to local statistics obtained, thus the better coding efficiency achieved. With the increasingly available coding modes, however, the overhead bits used to indicate mode information also increase inevitably, and this has become a big restraint on compression performance, especially in the low-bitrate applications.

3.2 Inter-prediction coding

For video sources, similar moving objects and background usually appear in consecutive video frames, and thus pixels in the current frame can be more efficiently predicted from pixels of previously coded frames than from nearby pixels within the same frame. This is known as motion compensated prediction.

The strategy for predicting motion of objects in video sequence is vital for achieving high compression efficiency. Since the establishment of H.261, MC-based inter-prediction techniques involved in video compression have developed rapidly in many aspects, such as reference frames available for MC, block size of MC, accuracy of MC and motion vector (MV), prediction for MV, interpolation filter for sub-pixel MC, and motion model for MC, etc.

In H.261, the only available reference frame is the closest previously coded frame, and MC is based on the block size of 16×16 only, with full-pixel MV accuracy and a simple MV prediction from the left neighboring block. In the subsequent MPEG-1 and MPEG-2, backward and bi-directional prediction, and half-pixel MC are introduced. After that, in MPEG-4 and H.263, MC block size of 8×8 is employed in addition to the size of 16×16, and the median of the three MVs of neighboring of blocks is used as the predictor of the current MV, with the accuracy of 1/4-pixel. In H.264/AVC, presently the most advanced standard in terms of compression efficiency, some new features are adopted to further improve the prediction efficiency, such as variable block sizes between 4×4 and 16×16 for MC, multiple and long-term reference frames (Wiegand *et al.*, 1999; Girod and Flierl, 2002), hierarchical B frame (Schwarz *et al.*, 2006), weighted prediction (Boyce, 2004), and in-loop deblocking filter. All the above-mentioned techniques are based on block-based MC prediction, which is the most established and implemented strategy employed in all international MPEG or ITU-T video compression standards. There are other MC strategies such as global motion compensation, sprite motion compensation (Kauff *et al.*, 1997; Smolic *et al.*, 1999b), segmentation-based (Karczewicz *et al.*, 1996) or object-based motion compensation (Smolic *et al.*, 1999a). These MC strategies extend the basic block-based motion prediction model toward segment- and model-based approaches and attempt further precise prediction by understanding the content (or even semantics) in video sequence and exploiting more complex, but also more efficient, motion models.

Since the establishment of H.264/AVC, a lot of coding tools involved in inter-prediction have been proposed to further improve the coding efficiency based on H.264/AVC, such as adaptive interpolation filter (AIF) (Wedi, 2002; 2006; Vatis *et al.*, 2005; Vatis and Ostermann, 2006; 2009; Ugur *et al.*, 2007;

Wittmann and Wedi, 2008; Rusanovskyy *et al*., 2008; 2009), extended MC block size (Chen *et al*., 2008; Kim *et al*., 2008), 1/8-pixel accuracy MV (Ostermann and Narroschke, 2006), decoder side MV derivation (Kamp *et al*., 2008; 2009), competition-based MV prediction (Won *et al*., 2009), and so forth. Among all of these techniques, AIF provides the most notable coding performance. In AIF schemes, the interpolation filter coefficients are adapted once per video frame to the non-stationary statistical properties of the video signal, by minimizing the prediction error variance.

Similar to the development of intra-prediction techniques, the coding gains of many inter-prediction techniques come from exploiting more coding modes, including variable MC block size, multiple reference frames, and so forth. As mentioned above, the employment of multiple coding modes improves adaptation to the non-stationary statistics of video signal at the expense of transmitting overhead bits. It should be noted that while many algorithms make adaptations on block level, AIF techniques perform adaptations on frame level.

3.3 Summary

By reviewing the development contexts of both intra- and inter-prediction technologies, we can identify common features. First, as the techniques become more and more efficient and flexible, the encoders and decoders become more and more complex. Second, as more and more coding modes are introduced, e.g., from the DC only intra-prediction to the nine intra-prediction modes, and from the 16×16 only MC block size to the variable MC block sizes between 4×4 and 16×16, more overhead bits are needed to send mode information. Thus, a balance between the overhead bits and achieved coding gain is needed. Third, it seems that the encoder is acquiring more responsibilities, such as motion estimation and mode decision, with the increasing coding modes, while the decoder usually works in a passive mode; that is, the decoder always works in accordance with instructions from the encoder, and never decides which MC block size should be used or what the MV will be, even though these can be indicated by the context or previously coded information. What if the decoder derived the mode information itself, without transmitting the overhead bits? The decoder side MV derivation strategy might be a practical example of this concept. Perhaps, in the future, the overhead bits can be saved with an intelligent enough decoder, and thus compression efficiency could be further improved.

4 Transform coding

Transform coding is a strategy that has been studied extensively during the past two decades and has become a very popular tool for ‘lossy’ still image and video coding. Many transform strategies have been proposed for image and video compression and the most popular transforms tend to fall into two categories: block-based and image-based. Block-based transforms operate on blocks of $N\times N$ image or residual samples. Block transforms have low memory requirements and are well-suited to compression of block-based motion compensation residuals but tend to suffer from artifacts at block edges. Image-based transforms operate on an entire image or frame (or a large section of the image known as a ‘tile’). Image transforms such as discrete wavelet transform (DWT) have been shown to outperform block-based transforms for still image compression, but they tend to have higher memory requirements and do not ‘fit’ well with block-based motion compensation. Examples of block-based transforms include KLT and ever-popular DCT, and the most popular image transform is the DWT (Vetterli and Kovacevic, 1995; Lee, 2005) implemented in JPEG 2000 (ITU-T and ISO/IEC, 2000) and MPEG-4. In this section, we will focus on the block-based transforms, the most popular and well-established techniques involved in almost all international video coding standards to date.

4.1 Function of transform coding

It is always said that transform coding can efficiently decorrelate the data to be coded, or can highly compact the signal energy. Does this mean that transform itself can reduce the information of the data to be coded or transmitted? Actually, the entropy of the transformed data is exactly the same as that of the original data.

In a block-based video coding scheme, an input image is split into disjoint pixel blocks, in which the pixels are highly correlated, and the pixel blocks can be considered as a vector source. In order to achieve the optimal rate-distortion (RD) performance, we usually want to know the RD bound of the data source. Calculation or estimation of the RD bound of a vector source, however, is very difficult or even impractical, and quantization and entropy coding applied directly to the original image or residual data becomes very complex. Transform coding solves this problem. After transformation, the components of the decorrelated image data are almost independent, and thus can be processed independently as scalar sources, which can attain a similar RD performance as treating the image data as a vector source.

Therefore, the function of transform in video coding is to convert a complex problem of processing a vector source to a simple problem of processing independent scalar sources, while attaining similar compression performance.

4.2 KLT and DCT

KLT is an optimum transform for data decorrelation, and can achieve minimum bitrate for a given distortion in squared-error measurement (Jayant and Noll, 1984). KLT is input-dependent, i.e., derivation of basis vectors of KLT is based on the covariance matrix of input data, and thus the basis vectors changes with the statistics of input data. There seem to be no fast, efficient algorithms for the operation of KLT. Because of the above-mentioned limitations, KLT is usually used to bound the performance of the suboptimal but practical transform algorithms, and is rarely utilized in practical applications.

Since natural image signals can be modeled by AR(1) statistics, the covariance matrix of image data (1D) can be derived as a Toeplitz matrix with block structure described in Eq. (1), and eigenvalues and eigenvectors of the covariance matrix are shown in Eqs. (2) and (3) respectively (Ray and Driver, 1970):

$$\boldsymbol{C}=\begin{pmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{N-1} \\ \rho & 1 & \rho & \cdots & \rho^{N-2} \\ \rho^2 & \rho & 1 & \cdots & \rho^{N-3} \\ \vdots & \vdots & \vdots & & \vdots \\ \rho^{N-1} & \rho^{N-2} & \rho^{N-3} & \cdots & 1 \end{pmatrix}, \tag{1}$$

$$\lambda_i=\frac{1-\rho^2}{1-2\rho\cos\omega_i+\rho^2}, \tag{2}$$

$$v_{ij}=\sqrt{\frac{2}{N+\lambda_i}}\sin\left[\omega_i\left(j-\frac{N-1}{2}\right)+\frac{(i+1)\pi}{2}\right], \tag{3}$$

$$0\le i,j\le N-1,$$

$$\tan(N\omega)=\frac{-(1-\rho^2)\sin\omega}{(1+\rho^2)\cos\omega-2\rho}, \tag{4}$$

where ρ denotes the adjacent-sample correlation coefficient, N denotes the order of KLT, and ω_i are the roots of Eq. (4). As ρ approaches 1, Eq. (5) is the regressive form of Eq. (3):

$$v_{0j}=\frac{1}{N},\ v_{ij}=\sqrt{\frac{2}{N}}\cos\left(\frac{\pi(2j+1)i}{2N}\right), \tag{5}$$

which are right the basis vectors of DCT. Since ρ of natural images is very close to 1, performance of DCT is virtually indistinguishable from that of KLT. Benefiting from an abundance of matured, fast algorithms, DCT is widely used in almost all video coding standards.

4.3 Development of transform techniques

Investigation of transform coding techniques can be traced back to 1970s, and block-based DCT was introduced in 1974. In 1988 DCT was adopted by JPEG. Since then, DCT has been widely employed by almost all video coding standards before H.264/AVC. In these standards, however, DCT was generally specified only within an error tolerance bound, due to the impracticality of obtaining an exact match to the ideal specified inverse transform. As a result, each decoder design would produce a slightly different decoded video, which would cause 'drift' between encoder and decoder representation of video and reduce effective video quality (Hinds *et al*., 2007). In order to solve this problem, fixed-point 8×8 IDCT and DCT standard was issued (ISO/IEC, 2008), and integer cosine transform (ICT) (Malvar *et al*., 2003) was introduced into the latest video coding standards, H.264/AVC.

Unlike the popular 8×8 floating point DCT defined in previous standards, the 4×4 ICT defined in H.264/AVC can be computed bit-exactly in integer arithmetic to avoid inverse transform mismatch problems. In addition, the 4×4 ICT in H.264/AVC can be computed with 16-bit additions and shifts only, and without multiplications, which minimizes computational complexity. In order to further improve the efficacy of the transform, a hierarchical block transform scheme is employed in H.264/AVC (Wiegand *et al*., 2003a). As an extension of ICT, pre-scaled integer transform (PIT) (Zhang *et al*., 2008) was adopted by AVS to keep all merits of ICT, and further reduce implementation complexity of the decoder, which is especially important and beneficial for implementation on low-end processors.

During recent years, in order to take advantage of adaptability to further improve compression efficiency, the strategies of adaptive block-size transform (Wien, 2003) and mode dependent directional transform (MDDT) (Ye and Karczewicz, 2008) were proposed and adopted into the high profile of H.264/AVC and KTA platform, respectively.

4.4 Summary

Transform coding has played an important role due to its superior ability of data decorrelation. As predictive coding techniques have become more sophisticated, however, prediction residues are getting smaller and correlations between prediction residual samples are getting lower. For this reason, some techniques have attempted to apply quantization and entropy coding directly onto the prediction residues (Narroschke, 2006) without transform. Performance gain was shown together with more complicated and precise prediction strategies.

It should be noted that adaptability was also employed into transform coding, and again, it brought performance gains.

5 Quantization

Video coding usually can be divided into two categories, namely lossless coding and lossy coding. Lossless coding can give bit-identical reconstructed data to the original data with only a moderate amount of compression, while lossy coding can achieve much higher compression ratio with the allowance of reconstruction distortion. For many applications, however, ‘lossless’ is not necessary, but a high compression ratio is much preferred, so lossy coding is widely used in most image and video coding standards.

Lossy coding is achieved through quantization, with the goal of encoding the data from a source (characterized by its probability density function) into as few bits as possible such that reproduction may be recovered from the bits with as high a quality as possible. Quantization includes scalar quantization, which deals with a single random variable, usually with low computational complexity, and vector quantization, which deals with a vector of random variables, usually with high computational complexity increasing rapidly as the dimension order is increased. Scalar quantization techniques are involved in most image and video coding standards with the combination of transform coding.

5.1 Quantization techniques in video coding

In the MC/DCT hybrid video coding schemes, uniform scalar quantization schemes are usually utilized to quantize the transform coefficients, and the quantization step size, which determines the quantization strength, is transmitted to the receiver. The available quantization step sizes in the early video coding schemes are in the form of 2 times the quantization parameter (QP), where QP is an integer larger than 0. However, this scheme has a non-uniform relative increment between the adjacent quantization step sizes, and thus the adjustment of bitrates or video quality is too coarse at the high bitrates while too fine at the low bitrates. This cannot fulfill the needs of applications requiring fine adjustment of the bitrates or video quality in a large range. For this reason, new quantization schemes were proposed, in which the quantization step sizes are in the form of $2^{\mathrm{QP}/c}$ where c is a constant. In this scheme, a relative increment between the adjacent quantization step sizes is $2^{1/c}$, which is constant, and thus the bitrates or video quality can be adjusted smoothly.

In the early quantization schemes in video coding, all of the transform coefficients in a block were quantized with the same quantization step size. The viewer is more sensitive, however, to reconstruction distortions related to low spatial frequencies than to high frequencies, and thus frequency adaptive weighting quantization techniques of the coefficients according to the human visual perception are often employed to improve the visual quality of the decoded images for a given bitrate. We can change quantization characteristics for each transform coefficient depending on the sensitivity of the human visual system and thus compress video more efficiently.

In order to achieve further improvements in coding efficiency, some adaptive quantization techniques have been proposed, such as adaptive rounding techniques (Sullivan and Sun, 2005) and trellis-based RD optimal quantization techniques (Wen *et al*., 2000; Jiang *et al*., 2005; Karczewicz *et al*., 2008). All of these techniques deal with the problem of efficiently rounding the quantized transform coefficient values to an integer.

5.2 Summary

Quantization is an essential part of lossy coding systems. Scalar quantization techniques are widely utilized in most image and video coding standards. According to the sensitivity of the human visual system, weighting quantization techniques are deployed to remove the visual redundancy of video data, where subjective distortion is taken into consideration as a distortion measure. It is interesting that we can again find the trace of 'adaptability' in the rounding techniques of quantizing transform coefficients.

6 Entropy coding

Entropy coding is a lossless data compression scheme. The purpose of entropy coding in video compression is to convert a series of symbols representing elements of the video sequence into a compressed bitstream suitable for transmission or storage, with as few bits as possible. Input symbols may include quantized transform coefficients, motion vectors, mode information and other syntax elements. There are two basic entropy coding techniques widely used in image and video coding standards, namely variable-length coding (VLC) and arithmetic coding. Compared to VLC, arithmetic coding usually provides further improvement in compression efficiency, with relatively high computational complexity.

6.1 Variable length coding

VLC methods in video coding are all syntax based; i.e., for every variable-length coded syntax element, there is a corresponding code table to map syntax values to codewords. All of the variable-length coded syntax elements are coded based on Huffman coding (Huffman, 1952), and VLC tables are defined based on probability distributions of 'generic' video materials. Among all of the syntax elements to be transmitted or stored, quantized transform coefficients (TCOEFs) usually take up most of the bitrates, with more attention given to them. More VLC tables are deployed for the coding of TCOEF, rather than other syntax elements such as mode information, MV and coded block pattern (CBP).

For coding the variable-length coded syntax elements other than TCOEF, in almost all of the video coding standards before H.264/AVC, different VLC tables were designed for each of these syntax elements, both stored at the encoder and decoder. In contrast to this, H.264/AVC uses a single infinite-extent codeword table for all of these syntax elements. Thus, only mapping to the single codeword table is customized according to statistics of syntax elements. The single codeword table is an exp-Golomb code with very simple and regular decoding properties, which greatly reduces the implementation complexity.

For coding TCOEF, a 2D TCOEF array is first reordered into a 1D sequence that typically contains one or more clusters of non-zero coefficients near the start, followed by strings of zero coefficients. The number of zeros preceding a non-zero coefficient (run) and the value of this nonzero coefficient (level) are highly correlated, and thus 'run' and 'level' are jointly coded as a run-level pair. Since the entropy of a joint distribution is smaller than the sum of the entropy of correlated individual distributions, it is efficient to encode the run-level pair as one symbol. To extend the concept of joint entropy coding and further improve compression performance, the end of block information (last) is jointly coded with run and level to form a run-level-last pair, in the MPEG-4 and H.263 video coding standards. All of the above-mentioned VLC methods, however, deploy only one VLC table to code all joint pairs of run-level or run-level-last. Higher order statistical dependencies on a syntax element level are mostly neglected. Since conditional entropy of a random variable is never larger than its entropy without any condition, the context-adaptive variable-length coding (CAVLC) method is employed in H.264/AVC, wherein multiple VLC tables matching with different conditioned statistics of TCOEFs are used and switched depending on conditions of previously transmitted syntax elements. The entropy coding performance is further improved compared to single-table VLC methods.

6.2 Arithmetic coding

The fundamental disadvantage of the VLC methods described above is that only a codeword containing an integral number of bits can be assigned to each symbol of an alphabet. In fact, the optimal number of bits for a symbol is usually a fractional number, and this depresses the coding efficiency of VLC methods. Arithmetic coding provides a practical alternative to Huffman coding that can more closely approach theoretical maximum compression ratios. In arithmetic coding, a sequence of data symbols are converted into a single fractional number and the optimal fractional number of bits required to represent each symbol can be approached.

During its long gestation in the 1970s and early 1980s, arithmetic coding was widely regarded more as an academic curiosity than a practical coding technique. Since having been first employed as a standardized entropy coding tool in JPEG, arithmetic coding has been widely utilized by many image and video coding standards, such as JPEG2000, MPEG-4, H.263, and the latest H.264/AVC.

The early arithmetic coding methods involved in video coding systems utilized the generic multi- alphabet arithmetic coder, which needed multiplication operations and thus involved a considerable amount of computational complexity. All probability models are non-adaptive in the sense that their underlying probability distributions are assumed to be static, which results in loss of adaptability to the non-stationary characteristics of video signal. In addition, these methods are syntax-based, and higher order statistical dependencies on the syntax element level are mostly neglected.

Context-adaptive binary arithmetic coding (Marpe *et al.*, 2003) in H.264/AVC avoids the drawbacks of the previous arithmetic coding methods successfully. On one hand, the usage of adaptive codes permits adaptation to non-stationary symbol statistics, and on the other hand, the context modeling of CABAC uses the statistics of already coded syntax elements to estimate conditional probabilities used for switching several estimated probability models. The binary arithmetic coding core engine and its associated probability estimation are specified as multiplication-free low-complexity methods using only shifts and table look-ups. All of these advantages of CABAC make it the state-of-the-art entropy coding technique.

Low power and high frame rate/resolution requirements for future video coding applications make parallelizable implementation necessary. The CABAC entropy coding engine has been identified as a key bottleneck in the H.264/AVC video decoder (Sze *et al.*, 2008). Parallelism is difficult to achieve with the existing H.264/AVC CABAC due to its inherent serial nature. For this reason, there have been several proposals (Segall and Zhao, 2008; Guo *et al.*, 2009) to address this critical problem.

6.3 Summary

The core technologies of entropy coding, VLC and arithmetic coding are mature. Adaptability plays an extremely important role in improving entropy coding efficiency. The state-of-the-art techniques of both arithmetic coding and variable-length coding are context-adaptive. Making sufficient use of contexts and well-tuning with other coding tools may lead to more efficiency with entropy coding. Parallelism should always be considered in the design of entropy coding schemes.

7 Trends and conclusion

From the analysis and discussion about the developments of video compression technologies in this paper, we can see that many compression improvements were made by increasing adaptability (AIF, CAVLC, CABAC, MDDT, etc.). An adaptive algorithm can fit its behavior to the changes of video content, or the local properties within a small area, and this is extremely beneficial for the non-stationary statistics of video signal. There are usually two ways to realize adaptation. One is based on mode selection or parameter calculation and the other is based on context modeling. In the former case, for different local regions with different statistics, the encoder adaptively selects the most efficient coding modes or calculates the optimum parameters to achieve the best coding performance. However, overhead bits have to be transmitted to indicate the mode or parameter information, and this disadvantage restricts the compression performance seriously, especially in low-bitrate applications. In the latter case, adaptation is based on the modeling of context, which refers to previously coded syntax elements, reconstructed pixels or other information that are available at both the encoder and the decoder. We know that the conditional entropy of a symbol, with highly correlated context as condition, is much smaller than the entropy without any condition. Therefore, modeling context more elaborately and using adaptability more widely will lead to further improvement in compression performance. Accordingly, the decoder will have to take more responsibility and better estimate the state of the source from the information received based on distortion models, source models, etc.

Hybrid video codecs achieve their compression performance by employing multiple coding modes that are adaptively selected for coding different regions with different statistical properties. One key problem in video compression is the determination of which coding modes or parameter settings should be utilized for a given part of the video signal to achieve the best compression efficiency in the RD sense. Lagrangian rate-distortion optimization techniques (Ortega and Ramchandran, 1998; Sullivan and Wiegand, 1998; Wiegand *et al*., 2003b) solve this problem in a simple but effective way, and thus have formed the most widely accepted approach in recent standard development. In these RD optimization techniques, however, the distortion between the original and compressed video sequences is typically evaluated as a mathematical error measurement, e.g., MSE, even though this type of measurement does not correlate well with perceptual quality and cannot lead to subjective optimal decisions. Therefore, objective distortion metrics considering perceptual quality, especially content-adaptive perceptual quality, are meaningful. The distortion metrics will play an important role in future video compression algorithms.

It remains to be seen the extent to which adaptability, context modeling and other strategies can be developed into mature algorithms to compress video more efficiently than today's standardized codecs. Future video compression algorithms may employ more adaptability, more refined temporal and spatial prediction models with better distortion metrics. The cost to users is the significant increase of implementation complexity at both the encoder and decoder. Fortunately, it seems that bitrates have a slower doubling time than computing power, so the disadvantage of increasing implementation complexity may one day be balanced with much improved processor capabilities.

References

Ahmed, N., Natarajan, T., Rao, K.R., 1974. Discrete cosine transform. *IEEE Trans. Comput.*, **C-23**(1):90-93. [doi:10.1109/T-C.1974.223784]

Boyce, J.M., 2004. Weighted Prediction in the H.264/MPEG AVC Video Coding Standard. Proc. Int. Symp. on Circuits and Systems, **3**:789-792. [doi:10.12109/ISCAS.2004.1328865]

Chen, P., Ye, Y., Karczewicz, M., 2008. Video Coding Using Extended Block Sizes. ITU-T Q.6/SG16 VCEG, VCEG-AJ23, San Diego, USA.

Cover, T.M., Thomas, J.A., 2003. Elements of Information Theory. Tsinghua University Press, Beijing, China, p.234-237.

Girod, B., 1987. The efficiency of motion-compensating prediction for hybrid coding of video sequences. *IEEE J. Sel. Areas Commun.*, **5**(7):1140-1154. [doi:10.1109/JSAC.1987.1146632]

Girod, B., Flierl, M., 2002. Multi-Frame Motion-Compensated Video Compression for the Digital Set-Top Box. Int. Conf. on Image Processing, **2**:1-4. [doi:10.1109/ICIP.2002.1039870]

Guo, X., Huang, Y., Lei, S., 2009. Ordered Entropy Slices for Parallel CABAC. ITU-T Q.6/SG16 VCEG, VCEG-AK25, Yokohama, Japan.

Guo, Y., Wang, Y., Li, H., 2008. Priority-Based Template Matching Intra Prediction. IEEE Int. Conf. on Multimedia and Expo, p.1117-1120. [doi:10.1109/ICME.2008.4607635]

Hinds, A.T., Reznik, Y.A., Yu, L., Ni, Z., Zhang, C., 2007. Drift analysis for integer IDCT. *SPIE*, **6696**: Article 14, p.1-16. [doi:10.1117/12.740220]

Huffman, D., 1952. A method for the construction of minimum redundancy codes. *Proc. IRE*, **40**(9):1098-1101. [doi:10.1109/JRPROC.1952.273898]

ISO/IEC JTC 1, 1993. Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 Mbits/s-Part 2: Video. ISO/IEC 11172-2 (MPEG-1 Part 2). Geneva, Switzerland.

ISO/IEC JTC 1, 1999. Coding of Audio-Visual Objects-Part 2: Visual. ISO/IEC 14496-2 (MPEG-4 Part 2). Geneva, Switzerland.

ISO/IEC JTC 1, 2008. Fixed-Point 8×8 Inverse Discrete Cosine Transform and Discrete Cosine Transform. Information Technology-MPEG Video Technologies-Part 2: ISO/IEC 23002-2. Geneva, Switzerland.

ITU-T, 1993. Video Codec for Audiovisual Services at px64 kbits/s. ITU-T Rec. H.261. Geneva, Switzerland.

ITU-T, 2000. Video Coding for Low Bit Rate Communication. ITU-T Rec. H.263. Geneva, Switzerland.

ITU-T and ISO/IEC, 1992. Digital Compression and Coding of Continuous-Tone Still Images. ITU-T Rec. T.81 and ISO/IEC 10918-1. Geneva, Switzerland.

ITU-T and ISO/IEC JTC 1, 1994. Generic Coding of Moving Pictures and Associated Audio Information-Part 2: Video. ITU-T Rec. H.262 and ISO/IEC 13818-2 (MPEG-2 Part 2). Geneva, Switzerland.

ITU-T and ISO/IEC JTC 1, 2000. JPEG2000 Image Coding System. ITU-T Rec. T.800 and ISO/IEC 15444-1. Geneva, Switzerland.

Jain, J.R., Jain, A.K., 1981. Displacement measurement and its application in interframe image coding. *IEEE Trans. Commun.*, **29**(12):1799-1808. [doi:10.1109/TCOM.1981.1094950]

Jayant, N.S., Noll, P., 1984. Digital Coding of Waveforms. Prentice-Hall, Englewood Cliffs, New Jersey, p.62-64, 524-546.

Jiang, W., Wang, J., Sun, J., 2005. Rate-distortion based quantization level adjustment for H.264. *Electron. Lett.*, **41**(16):903. [doi:10.1049/el:20051644]

Kamp, S., Evertz, M., Wien, M., 2008. Decoder Side Motion Vector Derivation for Inter Frame Video Coding. 15th IEEE Int. Conf. on Image Processing, p.1120-1123. [doi:10.1109/ICIP.2008.4711956]

Kamp, S., Bross, B., Wien, M., 2009. Fast Decoder Side Motion Vector Derivation for Inter Frame Video Coding. Picture Coding Symp., p.1-4. [doi:10.1109/PCS.2009.5167453]

Karczewicz, M., Nieweglowski, J., Lainema, J., Kalevo, O., 1996. Video Coding Using Motion Compensation with Polynomial Motion Vector Fields. 1st Int. Workshop on Wireless Image/Video Communications, p.26-31. [doi:10.1109/WIVC.1996.624638]

Karczewicz, M., Ye, Y., Chong, I., 2008. Rate Distortion Optimized Quantization. ITU-T Q.6/SG16 VCEG, VCEG-AH21, Antalya, Turkey.

Kauff, P., Makai, B., Rauthenberg, S., Golz, U., de Lameillieure, J.L.P., Sikora, T., 1997. Functional coding of video using a shape-adaptive DCT algorithm and object-based motion prediction toolbox. *IEEE Trans. Circ. Syst. Video Technol.*, **7**(1):181-196. [doi:10.1109/76.554429]

Kim, J., Na, T., Kim, C., Lee, B., Kim, M., 2008. Enlarging MB Size for High Fidelity Video Coding Beyond HD. ITU-T Q.6/SG16 VCEG, VCEG-AJ21, San Diego, USA.

Lee, D.T., 2005. JPEG 2000: retrospective and new developments. *Proc. IEEE*, **93**(1):32-41. [doi:10.1109/JPROC.2004.839613]

Malvar, H.S., Hallapuro, A., Karczewicz, M., Kerofsky, L., 2003. Low-complexity transform and quantization in H.264/AVC. *IEEE Trans. Circ. Syst. Video Technol.*, **13**(7):598-603. [doi:10.1109/TCSVT.2003.814964]

Marpe, D., Schwarz, H., Wiegand, T., 2003. Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. *IEEE Trans. Circ. Syst. Video Technol.*, **13**(7):620-636. [doi:10.1109/TCSVT.2003.815173]

Narroschke, M., 2006. Extending H.264/AVC by an Adaptive Coding of the Prediction Error. The 25th Picture Coding Symp., O5-3.

Ortega, A., Ramchandran, K., 1998. Rate-distortion methods for image and video compression. *IEEE Signal Process. Mag.*, **15**(6):23-50. [doi:10.1109/79.733495]

Ostermann, J., Narroschke, M., 2006. Motion Compensated Prediction with 1/8-Pel Displacement Vector Resolution. ITU-T Q.6/SG16 VCEG, VCEG-AD09. Hangzhou, China.

Ray, W., Driver, R.M., 1970. Further decomposition of the Karhunen-Loève series representation of a stationary random process. *IEEE Trans. Inf. Theory*, **16**(6):663-668. [doi:10.1109/TIT.1970.1054565]

Rusanovskyy, D., Ugur, K., Gabbouj, M., Lainema, J., 2008. Video Coding with Pixel-Aligned Directional Adaptive Interpolation Filters. IEEE Int. Symp. on Circuits and Systems, p.704-707. [doi:10.1109/ISCAS.2008.4541515]

Rusanovskyy, D., Ugur, K., Hallapuro, A., Lainema, J., Gabbouj, M., 2009. Video coding with low-complexity directional adaptive interpolation filters. *IEEE Trans. Circ. Syst. Video Technol.*, **19**(8):1239-1243. [doi:10.1109/TCSVT.2009.2022708]

Schwarz, H., Marpe, D., Wiegand, T., 2006. Analysis of Hierarchical B Pictures and MCTF. IEEE Int. Conf. on Multimedia and Expo, p.1929-1932. [doi:10.1109/ICME.2006.262934]

Segall, A., Zhao, J., 2008. Entropy Slices for Parallel Entropy Decoding. ITU-T SGI 6/Q.6 Doc. COM16-C405. Geneva, Switzerland.

Shannon, C.E., 1948. A mathematical theory of communication. *Bell Syst. Techn. J.*, **27**:379-423, 623-656.

Shannon, C.E., 1959. Coding Theorems for a Discrete Source with a Fidelity Criterion. IRE National Convention Record, Part 4, p.142-163.

Shiodera, T., Tanizawa, A., Chujoh, T., 2007. Block Based Extra/Inter-Polating Prediction for Intra Coding. IEEE Int. Conf. on Image Processing, **6**:445-448. [doi:10.1109/ICIP.2007.4379617]

Smolic, A., Makai, B., Sikora, T., 1999a. Real-time estimation of long-term 3-D motion parameters for SNHC face animation and model-based coding applications. *IEEE Trans. Circ. Syst. Video Technol.*, **9**(2):255-263. [doi:10.1109/76.752093]

Smolic, A., Sikora, T., Ohm, J.R., 1999b. Long-term global motion estimation and its application for sprite coding, content description, and segmentation. *IEEE Trans. Circ. Syst. Video Technol.*, **9**(8):1227-1242. [doi:10.1109/76.809158]

Sullivan, G.J., Sun, S., 2005. On dead-zone plus uniform threshold scalar quantization. *SPIE*, **5960**: Article 33, p.1-14. [doi:10.1117/12.631550]

Sullivan, G.J., Wiegand, T., 1998. Rate-distortion optimization for video compression. *IEEE Signal Process. Mag.*, **15**(6):74-90. [doi:10.1109/79.733497]

Sze, V., Demircin, M.U., Budagavi, M., 2008. CABAC Throughput Requirements for Real-Time Decoding. ITU-T Q.6/SG16 Doc. VCEG-AJ31. San Diego, USA.

Tan, K.T., Ghanbari, M., 2000. A multi-metric objective picture-quality measurement model for MPEG video. *IEEE Trans. Circ. Syst. Video Technol.*, **10**(7):1208-1213. [doi:10.1109/76.875525]

Tan, T.K., Boon, C.S., Suzuki, Y., 2006. Intra Prediction by Template Matching. IEEE Int. Conf. on Image Processing, p.1693-1696. [doi:10.1109/ICIP.2006.312685]

Tsukuba, T., Yamamoto T., Tokumo Y., Aono T., 2007. Adaptive Multidirectional Intra Prediction. ITU-T Q.6/SG16 VCEG, VCEG-AG05, Shenzhen, China.

Ugur, K., Lainema, J., Gabbouj, M., 2007. Adaptive Interpolation Filter with Flexible Symmetry for Coding High Resolution High Quality Video. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, p.1013-1016. [doi:10.1109/ICASSP.2007.366082]

van den Branden Lambrecht, C.J., Verscheure, O., 1996. Perceptual quality measure using a spatiotemporal model of the human visual system. *SPIE*, **2668**:450-461. [doi:10.1117/12.235440]

Vatis, Y., Ostermann, J., 2006. Locally Adaptive Non-Separable Interpolation Filter for H.264-AVC. IEEE Int. Conf. on Image Processing, p.33-36. [doi:10.1109/ICIP.2006. 313148]

Vatis, Y., Ostermann, J., 2009. Adaptive interpolation filter for H.264/AVC. *IEEE Trans. Circ. Syst. Video Technol.*, **19**(2):179-192. [doi:10.1109/TCSVT.2008.2009259]

Vatis, Y., Edler, B., Nguyen, D.T., Ostermann, J., 2005. Motion- and Aliasing-Compensated Prediction Using a Two-Dimensional Non-Separable Adaptive Wiener Interpolation Filter. ICIP IEEE Int. Conf. on Image Processing, **2**:894-897. [doi:10.1109/ICIP.2005.1530200]

Vetterli, M., Kovacevic, J., 1995. Wavelets and Subband Coding. Prentice-Hall, Englewood Cliffs, New Jersey, p.414-464.

Wedi, T., 2002. Adaptive Interpolation Filter for Motion Compensated Prediction. Int. Conf. on Image Processing, p.509-512. [doi:10.1109/ICIP.2002.1039999]

Wedi, T., 2006. Adaptive interpolation filters and high-resolution displacements for video coding. *IEEE Trans. Circ. Syst. Video Technol.*, **16**(4):484-491. [doi:10.1109/TCSVT.2006.870856]

Wen, J., Luttrell, M., Villasenor, J., 2000. Trellis-based R-D optimal quantization in H.263+. *IEEE Trans. Image Process.*, **9**(8):1431-1434. [doi:10.1109/83.855437]

Wiegand, T., Zhang, X., Girod, B., 1999. Long-term memory motion-compensated prediction. *IEEE Trans. Circ. Syst. Video Technol.*, **9**(1):70-84. [doi:10.1109/76.744276]

Wiegand, T., Sullivan, G.J., Bjontegaard, G., Luthra, A., 2003a. Overview of the H.264/AVC video coding standard. *IEEE Trans. Circ. Syst. Video Technol.*, **13**(7):560-576. [doi:10.1109/TCSVT.2003.815165]

Wiegand, T., Schwarz, H., Joch, A., Kossentini, F., Sullivan, G.J., 2003b. Rate-constrained coder control and comparison of video coding standards. *IEEE Trans. Circ. Syst. Video Technol.*, **13**(7):688-703. [doi:10.1109/TCSVT.2003.815168]

Wien, M., 2003. Variable block-size transforms for H.264/AVC. *IEEE Trans. Circ. Syst. Video Technol.*, **13**(7):604-613. [doi:10.1109/TCSVT.2003.815380]

Wittmann, S., Wedi, T., 2008. Separable Adaptive Interpolation Filter for Video Coding. 15th IEEE Int. Conf. on Image Processing, p.2500-2503. [doi:10.1109/ICIP.2008.4712301]

Won, K., Yang, J., Jeon, B., 2009. Motion Vector Coding Using Decoder-Side Estimation of Motion Vector. IEEE Int. Symp. on Broadband Multimedia Systems and Broadcasting, p.1-4. [doi:10.1109/ISBMSB.2009.5133843]

Wu, H., Yu, Z., Winkler, S., Chen, T., 2001. Impairment Metrics for MC/DPCM/DCT Encoded Digital Video. 22nd Picture Coding Symp., p.129-131.

Ye, Y., Karczewicz, M., 2008. Improved H.264 Intra Coding Based on Bi-Directional Intra Prediction, Directional Transform, and Adaptive Coefficient Scanning. 15th IEEE Int. Conf. on Image Processing, p.2116-2119. [doi:10.1109/ICIP.2008.4712205]

Yu, L., Chen, S., Wang, J., 2009. Overview of AVS video coding standards. *Signal Process.: Image Commun.*, **24**(4):247-262. [doi:10.1016/j.image.2009.02.003]

Zhang, C., Yu, L., Lou, J., Cham, W., Dong, J., 2008. The technique of prescaled integer transform: concept, design and applications. *IEEE Trans. Circ. Syst. Video Technol.*, **18**(1):84-97. [doi:10.1109/TCSVT.2007.913749]

Zheng, Y., Yin, P., Escoda, O.D., Li, X., Gomila, C., 2008. Intra Prediction Using Template Matching with Adaptive Illumination Compensation. 15th IEEE Int. Conf. on Image Processing, p.125-128. [doi:10.1109/ICIP.2008.4711707]