Emerging H.26L Standard:

Overview and TMS320C64x Digital Media Platform Implementation

White Paper

UB Video Inc.

Suite 400, 1788 west 5th Avenue

Vancouver, British Columbia, Canada V6J 1P2

Tel: 604-737-2426; Fax: 604-737-1514

www.ubvideo.com

H.26L: Introduction

Digital video is being adopted in an increasingly proliferating array of applications ranging from video telephony and videoconferencing to DVD and digital TV. The adoption of digital video in many applications has been fuelled by the development of video coding standards, and many video coding standards have emerged targeting different application areas. These standards provide the means needed to achieve interoperability between systems designed by different manufacturers for any given application, hence facilitating the growth of the video market. The ITU-T[1] is now one of two formal organizations that develop video coding standards - the other being ISO/IEC JTC1[2]. The ITU-T video coding standards are called recommendations, and they are denoted with H.26x (e.g., H.261, H.262, H.263 and H.26L). The ISO/IEC standards are denoted with MPEG-x (e.g., MPEG-1, MPEG-2 and MPEG-4).

The ITU-T recommendations have been designed mostly for real-time video communication applications, such as video conferencing and video telephony. On the other hand, the MPEG standards have been designed mostly to address the needs of video storage (DVD), broadcast video (broadcast TV), and video streaming (e.g., video over the internet, video over DSL, video over wireless) applications. For the most part, the two standardization committees have worked independently on the different standards. The only exception has been the H.262/MPEG-2 standard, which was developed jointly by the two committees. Recently, the ITU-T and the ISO/IEC JTC1 have agreed to join their efforts in the development of the emerging H.26L standard, which was initiated by the ITU-T committee. H.26L is being adopted by the two committees because it represents a departure in terms of performance from all existing video coding standards. Figure 1 summarizes the evolution of the ITU-T recommendations and the ISO/IEC MPEG standards. Please see [1] and [2] for more information on the H.263 and MPEG-4 video coding standards.

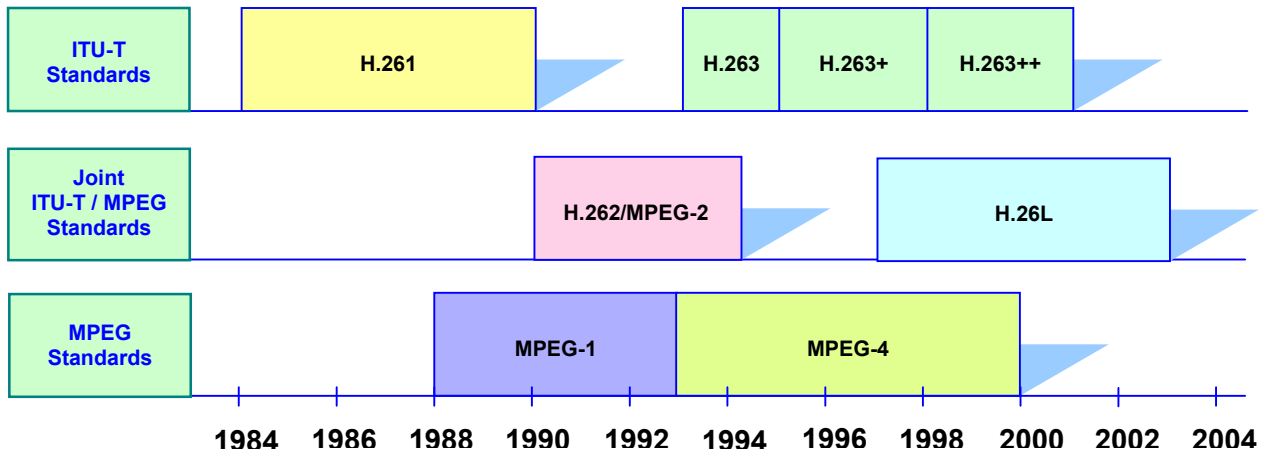


Figure 1. Progression of the ITU-T Recommendations and MPEG standards.

UB Video has been developing a complete video processing solution that is based on the H.26L video coding standard and that is optimized for the Texas Instruments TMS320C64x digital media platform family of DSPs. UBLive-26L-C64, UB Video's efficient and high-quality H.26L-based video processing solution, and the high-performance TMS320C64x family of DSPs, represent a compelling integrated software/hardware

[1] The International Telecommunications Union, Telecommunication Standardization Sector.
[2] International Standardization Organization and the International Electrotechnical Commission, Joint Technical Committee number 1.

solution to most high- performance video applications. The purpose of this paper is to present an overview of the emerging H.26L standard and the TMS320C64x digital media platform, as well as a discussion of UBLive-26L-C64. First, an overview of the H.26L standard and its benefits are presented, followed by a more detailed description of the main H.26L video coding features. The Texas Instruments TMS320C64x family of DSPs is then presented. UBLive-26L-C64's main features are then described, and its benefits for real-time video applications are finally discussed.

H.26L: Overview

The main objective behind the H.26L project is to develop a high-performance video coding standard by adopting a "back to basics" approach where simple and straightforward design using well-known building blocks is used. The ITU-T Video Coding Experts Group (VCEG) has initiated the work on the H.26L standard in 1997. Towards the end of 2001, and witnessing the superiority of video quality offered by H.26L-based software over that achieved by the existing most optimized MPEG-4 based software, ISO/IEC MPEG joined ITU-T VCEG by forming a Joint Video Team (JVT) that took over the H.26L project of the ITU-T. The JVT objective is to create a single video coding standard that would simultaneously result in a new part (likely Part-10) of the MPEG-4 family of standards and a new ITU-T (likely H.264) Recommendation. The H.26L development work is an on-going activity, with the first version of the standard is expected to be finalized technically before the end of this year 2002 and officially before the end of the year 2003.

The emerging H.26L standard has a number of features that distinguish it from existing standards, while at the same time, sharing common features with other existing standards. The following are some of the key features of H.26L:

1. **Up to 50% in bit rate savings**: Compared to H.263v2 (H.263+) or MPEG-4 Simple Profile, H.26L permits an average reduction in bit rate by up to 50% for a similar degree of encoder optimization at most bit rates.

2. **High quality video**: H.26L offers consistently high video quality at all bit rates, including low bit rates.

3. **Adaptation to delay constraints**: H.26L can operate in a low-delay mode to adapt to real-time communications applications (e.g., videoconferencing), while allowing higher processing delay in applications with no delay constraints (e.g. video storage, sever-based video streaming applications).

4. **Error resilience**: H.26L provides the tools necessary to deal with packet loss in packet networks and bit errors in error-prone wireless networks.

5. **Network friendliness**: A new feature is the conceptual separation between a Video Coding Layer (VCL), which provides the core high-compression representation of the video picture content, and a Network Adaptation Layer (NAL), which packages that representation for delivery over a particular type of network. This facilitates easier packetization and better information priority control.

The above features can be translated into a number of advantages for different video applications. Examples of these advantages are discussed for the specific application of video conferencing at the end of the paper.

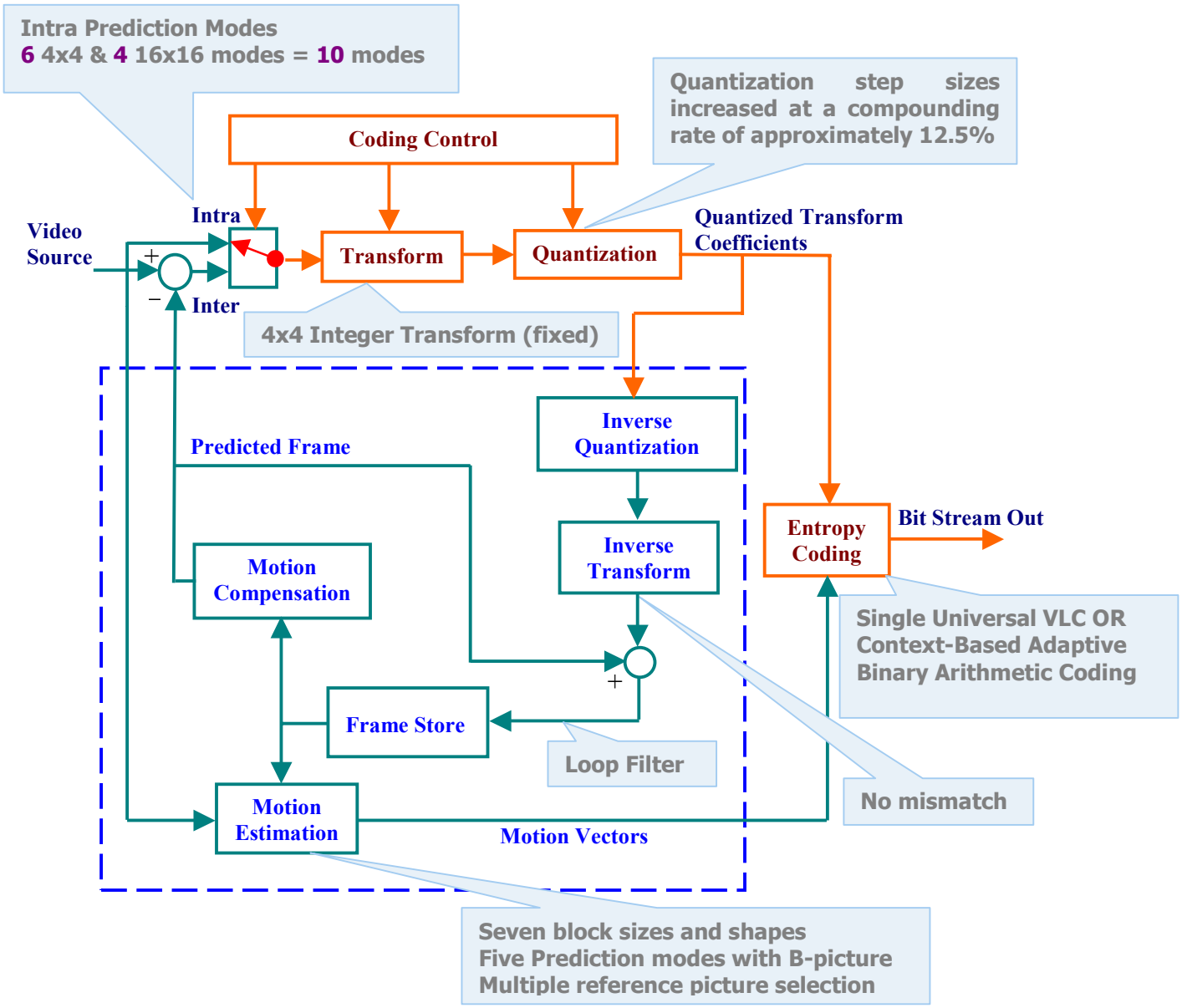


Figure 2. Block diagram of the H.26L encoder.

H.26L: Technical Description

Overview

The main objective of the emerging H. 26L standard is to provide a means to achieve substantially higher video quality compared to what could be achieved using any of the existing video coding standards. Nonetheless, the underlying approach of H.26L is similar to that adopted in previous standards such as H.263 and MPEG-4, and consists of the following four main stages:

1. Dividing each video frame into blocks of pixels so that processing of the video frame can be conducted at the block level.

2. Exploiting the spatial redundancies that exist within the video frame by coding some of the original blocks through transform, quantization and entropy coding (or variable-length coding).

3. Exploiting the temporal dependencies that exist between blocks in successive frames, so that only changes between successive frames need to be encoded. This is accomplished by using motion estimation and compensation. For any given block, a search is performed in the previously coded one or more frames to determine the motion vectors that are then used by the encoder and the decoder to predict the subject block.

4. Exploiting any remaining spatial redundancies that exist within the video frame by coding the residual blocks, i.e., the difference between the original blocks and the corresponding predicted blocks, again through transform, quantization and entropy coding.

From the coding point of view, the main differences between H.26L and the other standards are summarized in Figure 2 through an encoder block diagram. From the motion estimation/compensation side, H.26L employs blocks of different sizes and shapes, higher resolution sub-pel motion estimation, and multiple reference frame selection. In the transform side, H.26L uses an integer based transform that approximates the DCT transform used in previous standards, but does not have the mismatch problem in the inverse transform. In H.26L, entropy coding can be performed using either a single Universal Variable Length Codes (UVLC) table or using Context-based Adaptive Binary Arithmetic Coding (CABAC). These and other features are discussed in more detail in the following sections.

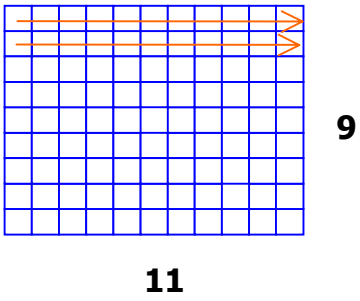


Figure 3. Subdivision of a QCIF picture into 16x16 macroblocks.

Organization of the Bitstream

As mentioned above, a given video picture is divided into a number of small blocks referred to as macroblocks. For example, a picture with QCIF resolution (176x144) is divided into 99 16x16 macroblocks as indicated in Figure 3. A similar macroblock segmentation is used for other frame sizes. The luminance component of the picture is sampled at these frame resolutions, while the chrominance components, Cb and

Cr, are downsampled by two in the horizontal and vertical directions. In addition, a picture may be divided into an integer number of "slices", which are valuable for resynchronization should some data be lost.

Intra Prediction and Coding

Intra coding refers to the case where only spatial redundancies within a video picture are exploited. The resulting frame is referred to as an I-picture. I-pictures are typically encoded by directly applying the transform to the different macroblocks in the frame. As a consequence, encoded I-pictures are large in size since a large amount of information is usually present in the frame, and no temporal information is used as part of the encoding process. In order to increase the efficiency of the intra coding process in H.26L, spatial correlation between adjacent macroblocks in a given frame is exploited. The idea is based on the observation that adjacent macroblocks tend to have similar properties. Therefore, as a first step in the encoding process for a given macroblock, one may predict the macroblock of interest from the surrounding macroblocks (typically the ones located on top and to the left of the macroblock of interest, since those macroblocks would have already been encoded). The difference between the actual macroblock and its prediction is then coded, which results in fewer bits to represent the macroblock of interest as compared to when applying the transform directly to the macroblock itself.

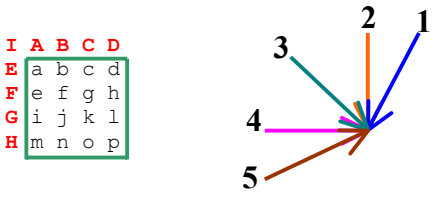


Figure 4. Intra prediction modes for 4x4 luminance blocks.

In order to perform the intra prediction mentioned above, H.26L offers six modes for prediction of 4x4 luminance blocks, including DC prediction (Mode 0) and five directional modes, labelled 1 thru 5 in Figure 4. This process is illustrated in Figure 4, in which pixels A to I from neighbouring blocks have already been encoded and may be used for prediction. For example, if Mode 2 is selected, then pixels a, e, i and m are predicted by setting them equal to pixel A, and pixels b, f, j and n are predicted by setting them equal to pixel B. For regions with less spatial detail (i.e. flat regions), H.26L also supports 16x16 intra coding, in which one of four prediction modes is chosen for the prediction of the entire macroblock. Finally, the prediction mode for each block is efficiently coded by assigning shorter symbols to more likely modes, where the probability of each mode is determined based on the modes used for coding surrounding blocks.

Inter Prediction and Coding

Inter prediction and coding is based on using motion estimation and compensation to take advantage of the temporal redundancies that exist between successive frames, hence, providing very efficient coding of video sequences. When a selected reference frame for motion estimation is a previously encoded frame, the frame to be encoded is referred to as a P-picture. When both a previously encoded frame and a future frame are chosen as reference

Motion Estimation is where H.26L makes most of its gains in coding efficiency.

frames, then the frame to be encoded is referred to as a B-picture. Motion estimation in H.26L supports most of the key features found in earlier video standards, but its efficiency is improved through added flexibility and functionality. In addition to supporting P-pictures (with single and multiple reference frames) and B-pictures, H.26L supports a new inter-stream transitional picture called an SP-picture. The inclusion of SP-pictures in a bit stream enables efficient switching between bit streams with similar content encoded at different bit rates, as well as random access and fast playback modes. The following four sections describe in more detail the four main motion estimation features used in H.26L namely, (1) the use of various block sizes and shapes, (2) the use of high-precision sub-pel motion vectors, (3) the use of multiple reference frames, and (4) the use of de-blocking filters in the prediction loop.

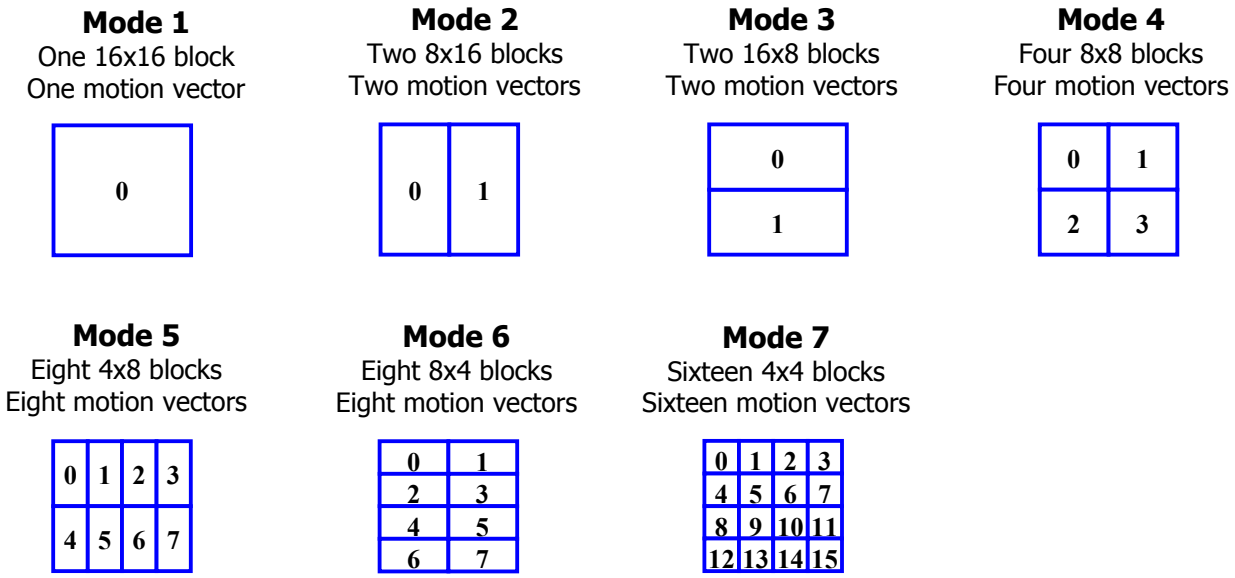


Figure 5. Different modes of dividing a macroblock for motion estimation in H.26L.

Block sizes

Motion compensation on each 16x16 macroblock can be performed using a number of different block sizes and shapes. These are illustrated in Figure 5. Individual motion vectors can be transmitted for blocks as small as 4x4, so up to 16 motion vectors may be transmitted for a single macroblock. Block sizes of 16x8, 8x16, 8x8, 8x4, and 4x8 are also supported as shown. The availability of smaller motion compensation blocks improves prediction in general, and in particular, the small blocks improve the ability of the model to handle fine motion detail and result in better subjective viewing quality because they do not produce large blocking artifacts.

Using seven different block sizes and shapes can translate into bit rate savings of more than 15% as compared to using only a 16x16 block size.

Motion Estimation Accuracy

Using ¼-pel spatial accuracy can yield more than 20% in bit rate savings as compared to using integer-pel spatial accuracy.

The prediction capability of the motion compensation algorithm in H.26L is further improved by allowing motion vectors to be determined with higher levels of spatial accuracy than in existing standards. Quarter-pixel accurate motion compensation is currently the lowest-accuracy form of motion compensation in H.26L (in contrast with prior standards based primarily on half-pel accuracy, with quarter-pel accuracy only available elsewhere in the newest versions of MPEG-4), while eighth-pixel accuracy is being adopted as a feature that will likely be useful for increased coding efficiency at high bit rates and high video resolutions.

Multiple reference picture selection

The H.26L standard offers the option of having multiple reference frames in inter picture coding. Up to five different reference frames could be selected, resulting in better subjective video quality and more efficient coding of the video frame under consideration. Moreover, using multiple reference frames might help making the H.26L bit stream error resilient. However, from an implementation point of view, there would be additional processing delays and higher memory requirements at both the encoder and decoder.

Using 5 reference frames for prediction can yield 5-10% in bit rate savings as compared to using only one reference frame.

De-blocking filter

H.26L specifies the use of an adaptive deblocking filter that operates on the horizontal and vertical block edges within the prediction loop in order to remove artefacts caused by block prediction errors. The filtering is generally based on 4x4 block boundaries, in which two pixels on either side of the boundary may be updated using a 3-tap filter. The rules for applying the current de-blocking filter are intricate and quite complex. Therefore, substantial efforts are being made to reduce the complexity of the de-blocking filter, which will almost certainly change before the H.26L standard is finalized.

Using the deblocking filter yields a substantial improvement in subjective quality.

Integer Transform

The information contained in a prediction error block resulting from either intra prediction or inter prediction is then re-expressed in the form of transform coefficients. H.26L is unique in that it employs a purely integer spatial transform (an approximation of the DCT) which is primarily 4x4 in shape, as opposed to the usual floating-point 8x8 DCT specified with rounding-error tolerances as used in earlier standards. The small shape helps reduce blocking and ringing artifacts, while the precise integer specification eliminates any mismatch issues between the encoder and decoder in the inverse transform.

Quantization and Transform Coefficient Scanning

The quantization step is where a significant portion of data compression takes place. In H.26L, the transform coefficients are quantized using scalar quantization with no widened dead-zone. Thirty-two different quantization step sizes can be chosen on a macroblock basis – this being similar to the abilities of prior standards (H.263 supports thirty-one, for example), but in H.26L the step sizes are increased at a compounding rate of approximately 12.5%, rather than increasing it by a constant increment. The fidelity of chrominance components is improved by using finer quantization step sizes as compared to those used for the luminance coefficients, particularly when the luminance coefficients are coarsely quantized.

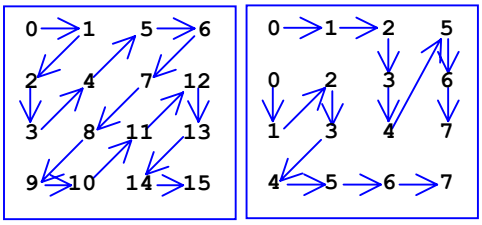


Figure 6. Two scan patterns for H.26L: single scan (left) and double scan (right).

The quantized transform coefficients correspond to different frequencies, with the coefficient at the top left hand corner in Figure 6 representing the DC value, and the rest of the coefficients corresponding to different nonzero frequency values. The next step in the encoding process is to arrange the quantized coefficients in an array, starting with the DC coefficient. Two different coefficient-scanning patterns are available in H.26L (Figure 6). The simple zigzag scan is used in most cases, and is identical to the conventional scan used in earlier video coding standards. The zigzag scan arranges the coefficient in an ascending order of the corresponding frequencies. The double scan is used only for intra blocks that use a small quantization step size, where subdivision of the scan into two parts improves coding efficiency.

Entropy Coding

The last step in the video coding process is entropy coding. So far, H.26L has adopted two approaches for entropy coding. The first approach is based on the use of Universal Variable Length Codes (UVLCs) and the second is based on Context-Based Adaptive Binary Arithmetic Coding (CABAC). However, substantial efforts are currently being made to adopt a single approach, which will likely be based on the adaptive use of special VLCs. Nonetheless, we next discuss briefly the current two approaches:

Universal VLC

Entropy coding that is based on the use of Variable Length Codes (VLCs) is the most widely used method for the compression of quantized transform coefficients, motion vectors, and other encoder information. VLCs are based on assigning shorter codewords to symbols with higher probabilities of occurrence, and longer codewords to symbols with less frequent occurrences. The symbols and the associated codewords are organized in look-up tables, referred to as VLC tables, which are stored at both the encoder and decoder.

In some video coding standards such as H.263, a number of VLC tables are used, depending on the type of data under consideration (e.g., transform coefficients, motion vectors). H.26L offers a single universal VLC table that is to be used in entropy coding of all symbols in the encoder, regardless of the type of data those symbols represent. Although the use of a single UVLC table is simple, is has a major disadvantage, which is that the single table is usually derived using a static probability distribution model, which ignores the correlations between the encoder symbols.

Context-Based Adaptive Binary Arithmetic Coding (CABAC)

Arithmetic coding makes use of a probability model at both the encoder and decoder for all the syntax elements (transform coefficients, motion vectors). To increase the coding efficiency of arithmetic coding, the

underlying probability model is adapted to the changing statistics with a video frame, through a process called context modeling.

Context modeling provides estimates of conditional probabilities of the coding symbols. Utilizing suitable context models, given inter-symbol redundancy can be exploited by switching between different probability models according to already coded symbols in the neighborhood of the current symbol to encode. Different models are often maintained for each syntax element (e.g., motion vectors and transform coefficients have different models). If a given symbol is non-binary valued, it will be mapped onto a sequence of binary decisions, so-called *bins*. The actual *binarization* is done according to a given binary tree – and in this case the UVLC binary tree is used. Each binary decision is then encoded with the arithmetic encoder using the new probability estimates, which have been updated during the previous context modeling stage. After encoding of each bin, we adjust upward the probability estimate for the binary symbol that was just encoded. Hence, the model keeps track of the actual statistics.

The use of Context based Adaptive Binary Arithmetic Coding in H.26L yields a consistent improvement of approximately 10% in bit savings.

H.26L: TMS320C64x Digital Media Platform Implementation

Designed to support both fixed and portable video/imaging applications, TI's digital media platform of DSPs combines the silicon, software, system-level expertise and support necessary to bring OEMs to market quickly with differentiated products. Together with its third party network members such as UB Video, TI has invested heavily in the video/imaging end equipment market. Products and support tools within the digital media platform are optimized to meet customers' unique performance needs.

The TMS320C64x DSP Platform

The TMS320C64x™ DSPs are the highest-performance fixed-point DSP generation in the TMS320C6000™ family of DSPs from Texas Instruments. The TMS320C64x DSP core delivers the highest performance at the lowest power consumption of any available DSP in the market to date. The TMS320C64x DSPs have broad capabilities and will enable multimedia communications and the full convergence of video, voice and data on all types of broadband networks. With the 600Mhz processing power of TI's TMS320C64x DSPs today and the planned next generation 1.1GHz C64x devices, these DSPs are most suited to overcome the complexity and computational requirements of H.26L and to deliver high-quality full-screen video for most broadband video applications.

The H.26L standard derives most of its performance gains from improved motion estimation. The TI TMS32064x family of DSPs from Texas Instruments has been well optimized for the type of operations performed in motion estimation. Consequently, the TM320C64x family of DSPs represents an ideal and powerful platform on which to run H.26L-based video coding software.

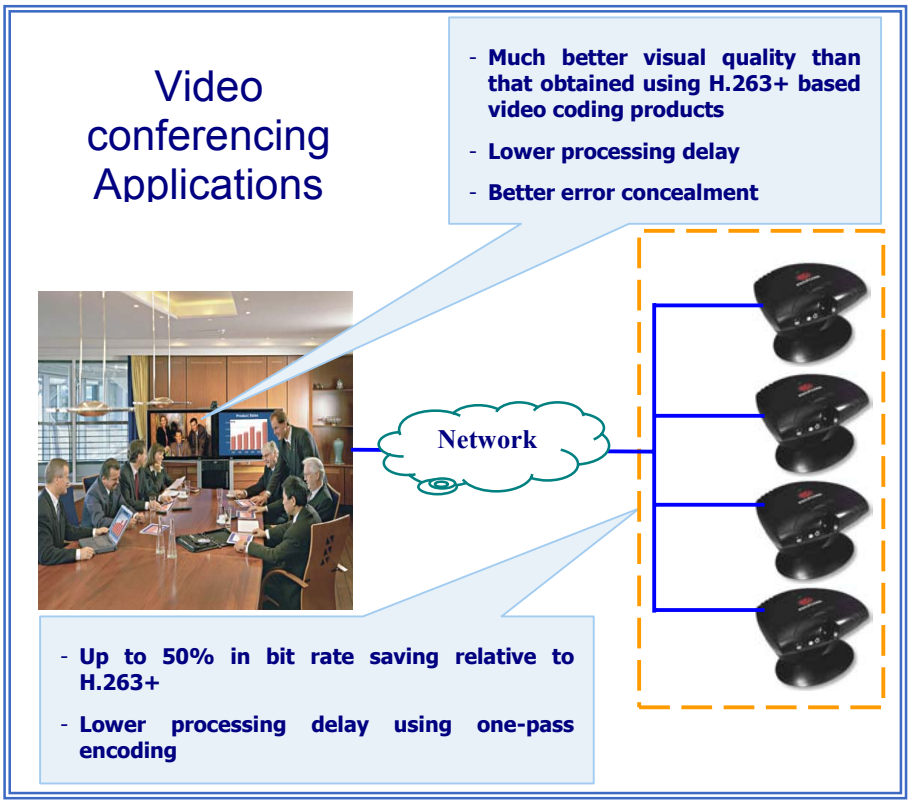
The TMS320C64x core is based on the second-generation VelociTI™ very-long-instruction-word (VLIW) architecture (VelocTI.2™) developed by Texas Instruments (TI), making the corresponding DSPs an ideal choice for multichannel and multifunction applications such image and video processing. The C64x™ is a code-compatible member of the C6000™ DSP platform.

A C64x™ DSP is capable of a performance of up to 4800 million instructions per second (MIPS) at a clock rate of 600 MHz. The C64x™ DSP core processor has 64 general-purpose registers of 32-bit word length and eight highly independent functional units-two multipliers for a 32-bit result and six arithmetic logic units (ALUs)- with VelociTI.2™ extensions. The VelociTI.2™ extensions in the eight functional units include new instructions to accelerate the performance in key applications and extend the parallelism of the VelociTI™ architecture. The C64x can produce four 16-bit multiply-accumulates (MACs) per cycle for a total of 2400 million MACs per second (MMACS), or eight 8-bit MACs per cycle for a total of 4800 MMACS. The C64x DSPs also have application-specific hardware logic, on-chip memory, and additional on-chip peripherals similar to the other C6000™ DSP platform devices.

The C64x uses a two-level cache-based architecture and has a powerful and diverse set of peripherals. The Level 1 program cache (L1P) is a 128-Kbit direct-mapped cache and the Level 1 data cache (L1D) is a 128-Kbit 2-way set-associative cache. The Level 2 memory/cache (L2) consists of an 8-Mbit memory space that is shared between program and data space. The L2 memory can be configured as mapped memory, cache, or combinations of the two.

UBLive-26L-C64: H.26L-Based Solution on the TMS320C64x for Real-Time Video Communication Applications

UB Video's UBLive-26L-C64 is an H.26L-based solution on the TMS320C64x platform for real-time video communication applications. The main features of the UBLive-26L-C64 are: (1) very efficient algorithms that help significantly reduce the implementation complexity, (2) TMS320C64x optimizations that take full advantage of the special features in the C64x architecture, and (3) intelligent pre- and post-processing stages that yield the best video quality in the market. The above features make UBLive-26L-C64 an ideal solution for many real-time video communication applications. To best illustrate the benefits UBLive-26L-C64 offers, an example real time application, video conferencing, is discussed next.



The major requirements for a typical video conferencing session are consistently good video quality even using a limited bandwidth, low delay and robustness to packet loss. In a video conferencing call involving audio, video and data, the video component typically consumes most of the available bandwidth for the call. Therefore, there would be a much wider acceptance of video conferencing systems over low bandwidth

networks if the required bandwidth for the video component part would be significantly reduced. This is precisely where UB Video's UBLive-26L-C64 brings most of its value, mainly reducing the bandwidth requirements by as much as 50% as compared to using H.263-based video solutions. For example, many of the video conferencing calls currently take place at around 384 kbps, of which video consumes about 320 kbps. For essentially the same video quality that would be obtained with H.263+ at 320 kbps, UB Video's UBLive-26L-C64 requires as little as 160 kbps, hence enabling the conference call to take place using a lower bandwidth network.

UB Video's UBLive-26L-C64 also offers a very low processing latency, a key requirement in video conferencing applications. Many of the existing video conferencing solutions still perform two-pass encoding in order to guarantee a satisfactory video quality, however the two-pass encoding method can introduce an objectionable delay during a conference call. UBLive-26L-C64 guarantees an excellent video quality even for single-pass encoding, thereby reducing processing latency.

Although most of the current video conferencing calls take place over local private networks, there is still a certain level of packet loss that takes place during packet transmission. UBLive-26L-C64 offers error resilience features at the encoding side and error concealment features at the decoder side that combat effectively packet loss even when the loss rate is high. This guarantees consistently good video quality during a conference call over an error prone packet network.

The UBLive-26L-C64 Demonstration Software: Description

The UB Video UBLive-26L-C64 demonstration software runs on the TMS320C64x Test Evaluation Board (TEB) from Texas Instruments. A block diagram of the demo is shown in Figure 7. The TEB is a platform where real-time API encode/decode function calls are done and where algorithms are tested for compliance. The way the UB Video demonstration software works is as follows: A camera captures frames at a rate of 30 frames per second and a resolution of 640x480. In the case where the SIF resolution mode is selected, frames are scaled down to SIF resolution. The output frame is then passed to the encoder API, and then to the decoder API. The decoded frame is converted from YUV to RGB and then displayed on the screen along with the original (possibly) scaled video frame.

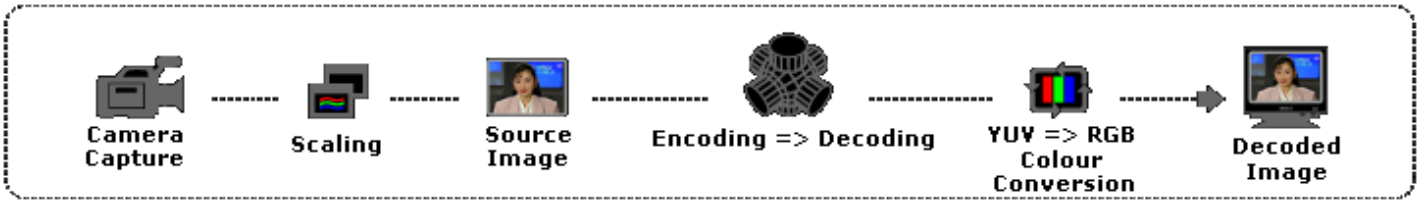


Figure 7. Block diagram of the UBLive-26L-C64 demo.

For more information on UBLive-26L-C64, please contact UB Video at www.ubvideo.com.

References

[1] H.263 Standard - Overview and TMS320C6000 Implementation. UB Video Inc. www.ubvideo.com.
[2] MPEG-4 Standard. UB Video Inc. www.ubvideo.com.