Visual Telephony as an ISDN Application

Ming L. Liou

VISUAL TELEPHONY IS THE PROCESS OF COMMUnicating at a distance by using moving image as well as voice. Today a telephone conversation is virtually second nature to all people in developed nations, but the usage of visual telephony is still a very rare occurrence. The reasons for this situation are many. Several key factors are elaborated below.

A voice signal can be transported through a telephone network with a bandwidth of 3.4 kHz while a video signal, using a regular television signal as an example, requires a bandwidth of 4.3 MHz, a ratio of 1:1265. Even though a picture is worth a thousand words, almost no one would want to pay for a visual conversation costing a thousand times more than a telephone call, assuming that connections are tariffed on the basis of bandwidth usage. Consequently, bandwidth reduction is the necessary first step to bringing visual telephony closer toward reality.

Today a telephone conversation is virtually second nature to all people in developed nations, but the usage of visual telephony is still a very rare occurrence.

There are many different ways to reduce the bandwidth required to transport a video signal. A great deal of progress has been made in the last two decades. Some techniques employ analog methods, but most employ digital coding techniques. It is important to note that the coding algorithm implemented for reduced bandwidth transmission has to be known to the receiving party, so that the underlying video signal can be decoded to its original form. Although various kinds of video codecs (coder/decoders) exist today, they cannot communicate with each other because they employ different coding algorithms. Consequently, the second important step for a successful realization of visual telephony is that the compression technique has to be standardized.

Due to the tremendous complexity of compression algorithms, a great deal of sophisticated electronic circuitry is required for their implementation. This is the major reason why video codecs are very expensive today. For ubiquitous deployment of any future video service, the cost of codecs has to be drastically reduced. Therefore, the next important step is to attain significant cost reduction by employing VLSI technology.

The last, but not the least, important step is the availability of a transport network to provide such a service. This network has to be digital, of high capacity, fully duplex, switchable, and ubiquitous. The timing of the ISDN deployment appears to be quite appropriate for the application of visual telephony.

Every step mentioned above, compression, standardization, VLSI and network availability, is essential for the successful deployment of new low bit-rate video services. Past attempts in providing similar services, such as Picturephone® and Picturephone Meeting Services®, were not successful because of the fact that one or more of the above mentioned ingredients were missing. The situation today is drastically different from the past. It appears that visual telephony using ISDN will soon become a reality.

System Overview

A system block diagram for visual telephony is shown in Figure 1. With the end-to-end digital connectivity of ISDN, a user is able to call another user with any combination of video, audio, and data at various rates. The transmission coder will combine video, audio, data, together with framing and other information, and deliver the resultant bit stream to an ISDN access port. The transmission decoder will perform the inverse operation. At present, there are two kinds of ISDN access that have been defined: Basic Access, and Primary Access. Basic Access supports 2B+D channels where the bit-rate for a B channel is 64 kb/s and that of a D channel is 16 kb/s. Basic Access will be available to every household and business when ISDN is fully deployed. Due to the severely limited available bit-rate, Basic Access is suitable only for desk top face-to-face visual communication which is often referred to as videophone. Primary Access supports 23B+D channels where the bit-rate for both B and D is 64 kb/s. Since Primary Access requires a T1 line it is expected to be used mostly in business applications. Due to the additional available bit-rate, pictures transmitted by Primary Access can be more complex with better quality. Primary Access is therefore more suitable than Basic Access for video teleconferencing. In summary, for

®registered trademark of AT&T Bell Laboratories.

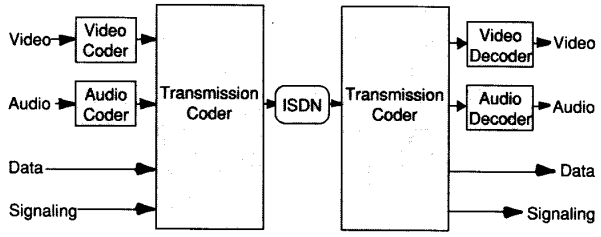


Figure 1. Visual telephony system architecture.

ISDN applications, the bit-rate for combined video and audio services is limited to px64 kb/s where p=1,2,....30 (for North American network, p = 1,2,...,23). The CCITT Specialists Group on Coding for Visual Telephony has been working on the standardization of a video coding algorithm for these rates with the anticipation that an international standard will be established in mid 1990. Standards activities will be discussed more later.

Video Signal Format

A video signal is composed of a sequence of pictures called frames. In some cases, each frame is composed of two interlaced fields for the purpose of eliminating flickering. Each frame contains a certain number of lines. After sampling, each line contains a certain number of picture elements called pixels or pels. A color video signal may be represented by three primary color components (i.e., red, green, and blue), or by a luminance and two chrominance components. The two chrominance components may be modulated and combined with the luminance component to form a composite signal.

At present, there are three major television standards: NTSC, PAL, and SECAM, each using a composite signal format that is not, unfortunately, compatible with the others. In addition, the bandwidth of these television signals is too large for economical visual telephony. One solution to this problem is to adopt an international standard for a video signal format with a reduced spatial resolution. The CCITT Specialists Group has recently adopted the Common Intermediate Format (CIF) and 1/4 CIF as the video signal formats for visual telephony. The parameters for these formats are shown in Figure 2. Note that the chrominance components U and V are subsampled 2:1 both horizontally and vertically. The numbers of coded pels per line are reduced as indicated in Figure 2 for ease of a specific coding algorithm to be described later. With 30 frames per second and 8 bits per pel, the uncompressed bit-rates for CIF and 1/4 CIF are 36.5 and 9.12 Mb/s respectively. Consequently, a compression ratio of approximately 570:1 is required for an ISDN B channel to carry a CIF video signal (which is not an easy task to achieve at a reasonable cost).

For this reason, 1/4 CIF will be used as the basic format for videophone applications with the CIF as an option. CIF will be most likely used for video teleconferencing applications since 1/4 CIF cannot provide enough resolution for such applications. Further reduction in bit-rate can be achieved by reducing the frame rate from 30 to 15, 10 or even 7.5. Consequently, for a B channel to carry a 1/4 CIF signal, a compression ratio ranging from 36:1 to 72:1 is needed by using other coding techniques. In other words, the available bits per pel is in the range of 0.1 to 0.2 which is extremely challenging especially for real-time implementation.

Video Coding Algorithm

Numerous video compression techniques [1–5] have been proposed in the last two decades and new ones are being developed everyday. It is not the intention of this article to conduct a thorough review in this area. Instead, the proposed video coding algorithm by the CCITT Specialist Group [6] will be briefly described here.

The basic objective of video coding is to compress the data rate by removing redundant information. There are two major categories of coding schemes (i.e., source coding and entropy coding). Source coding deals with source material and yields results which are lossy (i.e., picture quality is degraded). Source coding can be further divided into intra- and inter-frame coding. Intraframe coding is used for the first picture and for later pictures after a change of scene. Interframe coding is for sequences of pictures containing moving objects. Enthropy coding achieves compression by using the statistical properties of the signals and is, in theory, lossless.

The basic objective of video coding is to compress the data rate by removing redundant information.

The proposed coding algorithm uses both schemes. A simplified block diagram of the encoder is shown in Figure 3a and the decoder in Figure 3b. For source coding, a hybrid transform/DPCM (Differential Pulse Coded Modulation) with motion estimation is used. Here the DPCM is not operative for intraframe coding. For entropy coding, both one- and two-dimensional variable length codings are used.

Referring to Figure 4, each picture frame is first divided into macroblocks which consist of a 16 × 16 pels luminance block and the two corresponding 8 × 8 pels chrominance blocks (recall that colors have been subsampled). The 16 × 16 luminance block is further divided into four 8 × 8 sub-blocks. One important aspect of the proposed algorithm which differs from the previous ones is that the transform is performed on the 8 × 8 blocks (for both luminance and chrominance) to reduce the difficulty of real-time processing and motion estimation is performed on the 16 × 16 blocks (luminance only) to increase coding efficiency [6–8].

The type of transform used in this application is the Discrete Cosine Transform (DCT) [9]. It is performed on the difference between the 8 × 8 blocks of the current frame and the corresponding blocks of the predicted frame (which is obtained from the previous frame plus motion information). Obviously if a block contains no motion or the predicted value is exact, the input to the DCT will be a null matrix. For slowly moving pictures, the input matrix to the DCT will contain many zeros. The output of the DCT is a matrix of coefficients which represent energy in the two-dimensional frequency domain. In gen-

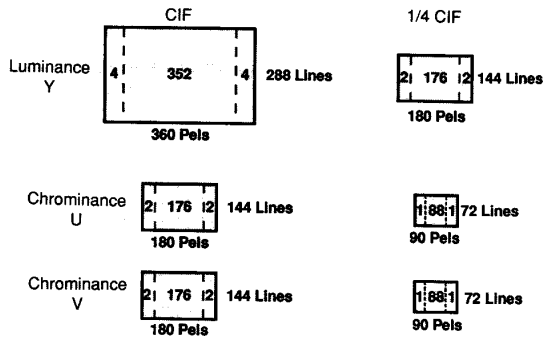
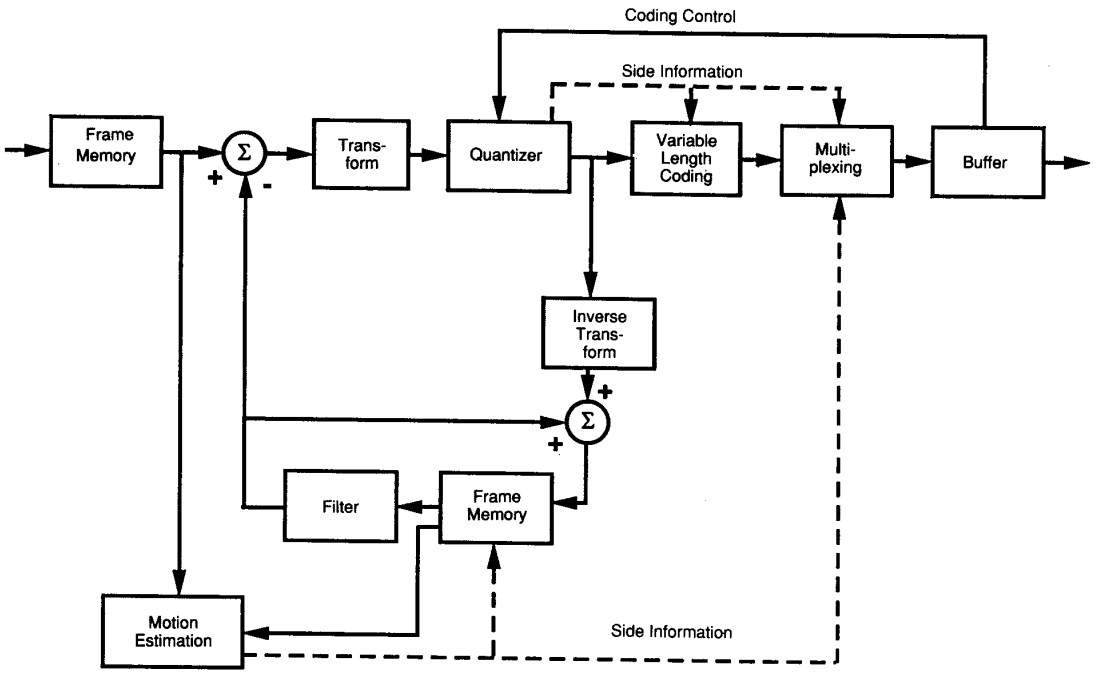
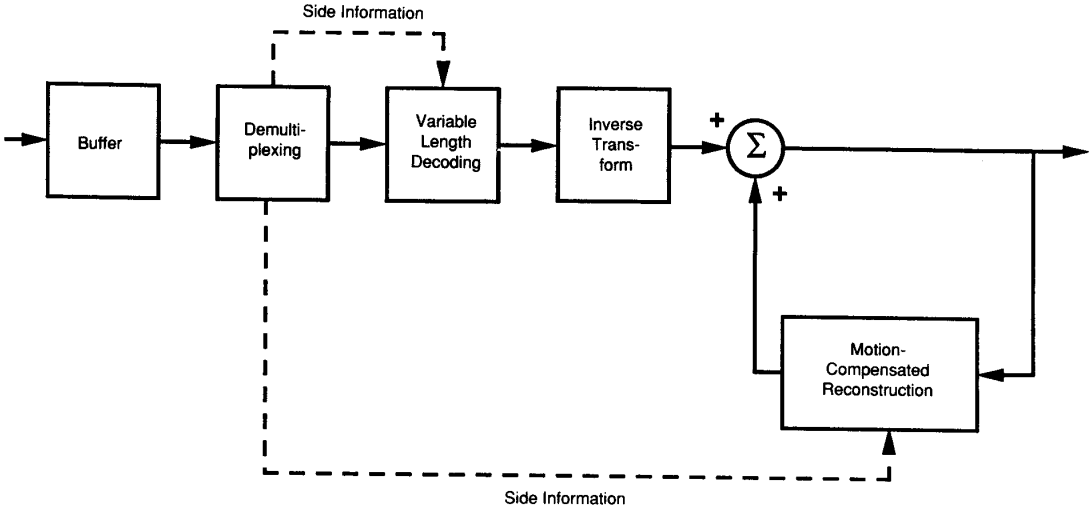


Figure 2. Picture formats for low-bit-rate video.



(a) Encoder



(b) Decoder

Figure 3. Video codec.

eral, most of the energy is concentrated at the upper left corner of the matrix, which is the low frequency region. If the coefficients are scanned in a zigzag manner, shown in Figure 5, the resultant sequence will contain long strings of zeros especially toward the end of the sequence. It is clear, at this point, that one of the major objectives of this compression algorithm is to create zeros and to bunch them together for efficient coding. For exactly the same reason, a variable threshold is also applied to this sequence before quantization. This is accomplished by increasing the threshold (a DCT coefficient is set to zero if it is less or equal to the threshold) when a string of zeros is detected.

A uniform quantizer is used after the transform. The step size of the quantizer can be adjusted by the transmission rate as

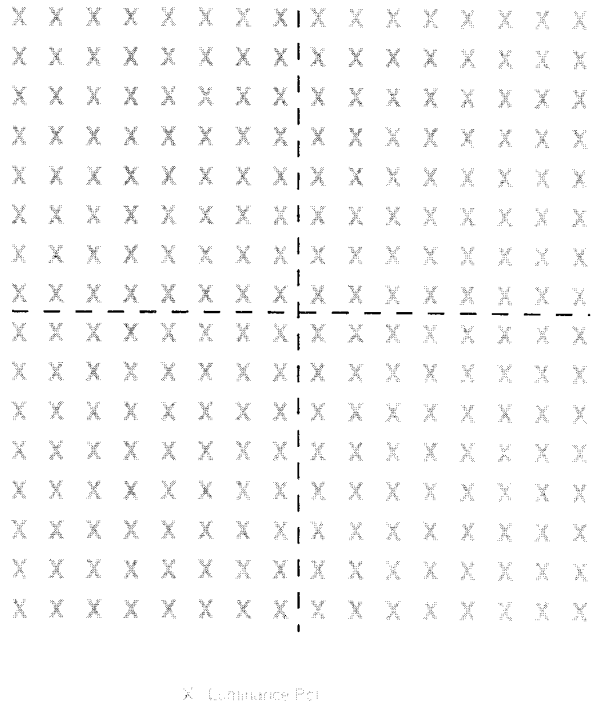


Figure 4. Composition of a macroblock.

indicated by the buffer occupancy. When the transmission rate reaches its limit, the step size will be increased so that less information needs to be coded which, of course, will result in a degraded picture. On the other hand, the step size will be decreased to improve picture quality when the transmission rate is below its limit.

To further increase coding efficiency, a two-dimensional variable length coding scheme is used for the sequences of quantized DCT coefficients. In a given sequence, the value of a non-zero coefficient (LEVEL) is defined as one dimension and the number of zeros preceding the non-zero coefficient (RUN) is defined as another dimension. The combination of the two is defined as an EVENT, i.e.,

$$\text{EVENT} = (\text{RUN}, \text{LEVEL}).$$

A shorter length code is assigned to an EVENT which occurs more frequently and vice versa. An EOB (End of Block) marker is also used to indicate that there are no more non-zero coefficients in the sequence.

The coded coefficient values are then multiplexed together with various side information such as block classification, quantization information, and differential motion vectors. The resultant bit-stream is finally sent to the buffer for transmission. It should be noted that some of the side information is also variable length coded.

The method used for motion estimation is to compare the current 16 × 16 luminance block with the luminance blocks in the previous frame within a specified tracking range and choose the one with the minimum total absolute change from the previous frame. The position of the chosen block is called the motion vector which is used to obtain the predicted values of the current block. For additional coding efficiency, the motion vectors are differentially coded followed by a Variable-Length Coder (VLC) for transmission as side information to a decoder.

A low-pass filter is used in the DPCM loop for the purpose of smoothing out the predicted values when needed.

In order to protect the coded bit-stream from various kinds of random noise, a forward error correction scheme known as the BCH (Bose-Chaudhuri-Hocquenghem) code with a block length of 511 containing 18 bits for error correction has been recommended.

The decoder performs the inverse operation of the encoder. Its architecture is much simpler than the encoder.

Audio Coding Algorithm

Speech coding is a relatively mature research topic. However, the standardization of a 16 kb/s audio codec is at least one year behind the video codec. This has a tremendous impact on videophone services since digital capacity is very limited.

With ISDN Basic Access there are three possible arrangements for videophone usage:

- 1. One B channel - 16 kb/s for audio and 48 kb/s for video.
- 2. Two B channels - 64 kb/s for audio and 64 kb/s for video.
- 3. Two B channels - 16 kb/s for audio and 112 kb/s for video.

Arrangements 1 and 3 require 16 kb/s audio codecs. Arrangement 1 is most economical. However, Arrangement 3 is extremely important since it will deliver much better picture quality. For the moment, most attention and activities are being directed toward Arrangement 2.

Two CCITT standards exist for a 64 kb/s audio codec, namely:

G.711—encodes 3.1 kHz (300 to 3400 Hz) audio into 64 or 56 kb/s using logarithmic PCM [10]

G.722—encodes 7 kHz audio into 64, 56, or 48 kb/s using subband ADPCM [11].

Implementation of the audio coding algorithms is relatively easier than that of the video codec. Generally audio codecs can be implemented using a single digital signal processor, a programmable VLSI chip, that is readily available on the market.

Video Codec Implementation

In order to reach the desired video compression ratio, tre-

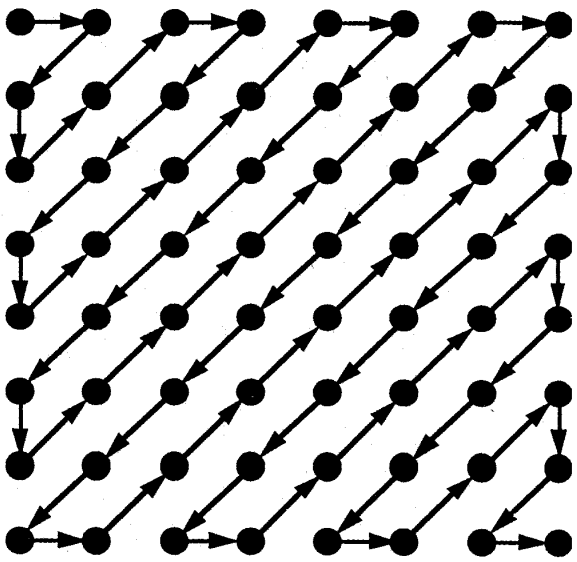


Figure 5. Zigzag scanning pattern of DCT coefficients.

mendous signal processing capability is required. The video coding algorithm is at least one order of magnitude more complex than that of the audio. In order to implement the video coding algorithm in real-time with reasonable cost, state-of-the-art VLSI technology has to be employed. Due to recent advances in digital electronics, high density and low power CMOS (Complementary Metal Oxide Semiconductor) technology is available for this application. With the possibility of mass market, the cost of videophone can be expected to be drastically reduced as a result of VLSI.

There are two distinct approaches for video codec implementation using VLSI. One approach uses high performance signal processors, and the other uses ASICs (Application Specific Integrated Circuits). Each approach has its advantages and disadvantages. Generally speaking, the approach using signal processors is more flexible, requires less development effort, but will end up with a more expensive video codec than the approach using ASICs. More details about the two approaches are given below.

Implementation Using High Performance Signal Processors

A high performance signal processor is a single chip VLSI electronic circuit designed for signal processing applications. It is programmable and optimized for number crunching capabilities such as multiplication and accumulation. Its input and output can be interfaced with other electronic circuits for real-time operations.

Generally speaking, there are two approaches to implementation of a video codec using high performance signal processors: the functional approach and the distributed approach.

There are various types of high performance signal processors suitable for video codec implementation. They include DSPs (Digital Signal Processors), Transputers and VSPs (Video Signal Processors). Each type has its own distinct architecture and characteristics. It is not the intention of this article to present a detailed survey on this subject area. However, a brief description of the application of these signal processors to real-time implementation of video codecs is presented.

As mentioned earlier, considerable signal processing is required to compress a CIF video source signal to an ISDN rate for real-time transmission. At present, no single signal processor of any type can handle this task alone. In fact, a large number of them must be intricately interconnected in a multiprocessor system to meet video signal processing requirements.

Generally speaking, there are two approaches to implementation of a video codec using high performance signal processors: the functional approach and the distributed approach. The functional approach replaces a functional module such as a DCT/Inverse DCT (IDCT) or motion estimator by an appropriate number of signal processors depending on its computational complexity. The distributed approach, on the other hand, is to assign one signal processor for each small image region within a frame, and use as many signal processors as needed in parallel to meet the required processing speed. In any event, a sizable engineering effort, both in hardware and software design, is needed. Some comments on each type of signal processor for video codec implementation are given below.

Various types of DSPs with varying degree of capabilities have appeared on the market [12][13]. The key features of a DSP include word-length (mostly 16 or 32 bits), floating or fixed point computation, and speed as measured by MIPS (Million Instructions Per Second) or more appropriately by MACs (Multiplications and Accumulations) per second.

The DSPs have been commonly used for implementing audio codecs in the past. Usually one or two DSPs are required, depending on the degree of coding complexity. An order of magnitude more DSPs are needed to implement a video codec. Due to the complexity and speed demand of video coding algorithms, parallel processing with multiple DSPs is necessary. Since most DSPs are not designed for parallel processing, their use in implementation of video codecs is complicated and inefficient. In addition, the on-chip memory is usually too small for video application which further reduces the efficiency of the DSPs. On the other hand, one outstanding advantage of using DSPs for video codec implementation is that they are readily available with excellent engineering support.

A Transputer [14] is a 32-bit high speed Reduced Instruction Set Computer (RISC) designed for parallel processing. It is, therefore, in some respects more suitable than the DSPs for video codec implementation [15]. The present capability of a Transputer is limited by its processing power (< 1.5 M FLOPS) and the size of on-chip memory (4K bytes). Consequently, many more Transputers are needed to implement a video code than DSPs.

In order to eliminate certain deficiencies of DSPs and Transputers, signal processors especially tailored for video application have been developed recently [16–21]. They normally have a reduced word-length and instruction set with appropriate hardware architecture to support parallel processing. Not only is the architecture highly pipelined, but it is also reconfigurable for various tasks. The sizes of RAM and ROM are optimized for video applications. The degree of programmability depends on the particular architecture used. One VSP is designed for the distributed approach [16]. A few are optimized for a functional approach [17] [18]. Others can be used in either approach [19–21]. In any event, a VSP is certainly more efficient than the DSPs and Transputers. Unfortunately, they are not yet generally available on the market and can thus only be used by the original designers.

ASIC-Based Implementation

Video codecs implemented with high performance signal processors are still expensive for widespread customer acceptance. For ubiquitous videophone usage, cost is the most serious concern. Consequently, many ASICs have to be designed and used. The development cost for ASICs is, in general, higher than that of high performance signal processors. However, with the recent advances in VLSI design tools, the development cost of ASICs has been greatly reduced. The concern about loss of flexibility by using ASICs is disappearing because the proposed coding algorithm is becoming an international standard. With the large potential worldwide market volume, the ASIC-based approach is ideal for mass production that in turn will drive the cost even lower. The VLSI research in this area has already begun and some key signal processing modules in the video codec have already been implemented, which will be briefly discussed below.

Discrete Cosine Transform Chip

Among various transform techniques for image compression, the Discrete Cosine Transform (DCT) [9] is perhaps the most popular and effective in practical applications. The CCITT Specialists Group on Coding for Visual Telephony has chosen the 8×8 DCT and Inverse DCT (IDCT) as the key signal processing modules for video compression. A two-

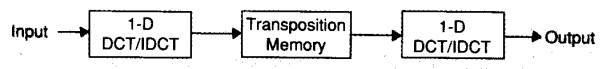


Figure 6. 2-D DCT/IDCT by row-column decomposition.

dimensional (2-D) DCT/IDCT pair is defined as:

$$Z = C^t X C$$

$$X = C Z C^t$$

where X and Z are $N \times N$ blocks of pixel data and transform coefficients respectively and C (the superscript t denotes the transpose) is the $N \times N$ DCT coefficient matrix with element C_k defined by

$$C_{kl} = \sqrt{\frac{2}{N}} \cos\left[\frac{(2k-1)(l-1)\pi}{2N}\right] \qquad (1)$$

for $k = 1,2,\ldots,N$, $= 2,3,\ldots,N$, and $C_{kl} = N^{-1/2}$ for $l = 1$

The above matrix equation appears to be simple and straight forward and hence should not be a source of concern. However, when it needs to be computed in real-time, the required processing power is very high. For the case when $N = 8$, the number of MACs required for each 8×8 block is 1024. Since there are six 8×8 blocks in a macroblock, 396 macroblocks in a CIF frame, 30 frames per second, the total processing power needed is almost 73 million MACs per second. Using a straight forward approach, it would require more than two AT&T DSP16As or seven TI TMS320C25s to just process the DCT alone. Of course, no one is using this approach at present since various fast algorithms for computing DCTs do exist [22] [23]. Even so, it still needs about 14 million MACs per second to compute the DCTs. The same requirement is true for computing the IDCTs. Because of this extremely demanding processing power, as well as its clearly defined inputs/outputs for other possible applications, a single chip DCT/IDCT becomes a prime candidate for ASIC implementation. Much research effort has already been carried out on this subject area in recent years with impressive results [24–28].

The 2-D DCT/IDCT defined earlier can be implemented by the row-column decomposition technique (shown in Figure 6) where the input is read row by row. The output of the first 1-D DCT (or IDCT) is XC (or ZC^t). The transposition memory performs matrix transposition to yield C^tX^t (or CZ^t). Similarly, the output of the second 1-D DCT (or IDCT) gives C^tX^tC (or CZ^tC^t) which is the transpose of Z (or X). Thus the output of the 2-D DCT/IDCT is written column by column. From this figure, it can be seen that the main signal processing engine is the 1-D DCT/IDCT that can be implemented using various approaches. One general category of approaches may be called algorithmic with the objective of minimizing the number of multipliers [24] [25]. The other general category of approaches may be called distributed arithmetic with the objective of minimizing chip area and maximizing regularity [26–28]. In any event, the technology for a single chip 2-D DCT/IDCT appears to be at hand.

In order to allow innovation and competition, the implementation technique for DCT/IDCT should not be a subject for standardization. However, to reduce mismatch problems between a coder and decoder using different implementations of IDCT, a set of accuracy requirements on IDCT is required and has recently been approved by the CCITT Specialists Group on Coding for Visual Telephony [6].

Motion Estimation Chip

A block-matching motion estimation technique has been adopted for interframe coding as mentioned earlier. Various algorithms can be used to produce a motion vector with varying degrees of accuracy. However, within a given tracking range, the most thorough approach is the so called "full search" technique. The criterion used for motion estimation is the minimum absolute differences with

Motion Vector $(V, H) =$

$$\underset{v,h}{Min} \sum_{i=1}^{16}\sum_{j=1}^{16} \left| a(i,j) - b(i+v, j+h) \right| \qquad (2)$$

where $a(i,j)$ is the luminance pixel value in a 16×16 macroblock in the current frame, $b(i+v,j+h)$ is the corresponding luminance pixel value in a 16×16 macroblock that is shifted v pixels vertically and h pixels horizontally in the previous frame, and (V, H) is the value of (v,h) which yields the minimum of the double sums of absolute differences in a given tracking range. Again, the above equation appears to be simple and straight forward with the basic processing element involving only subtractions, calculating absolute values, and accumulations. However, to execute it for real-time processing presents another dimension of challenge.

Assuming CIF format operating at 30 frames per second with a tracking range of -8 to +7 pixels, the total computational requirement is about 780 million operations per second where an operation is defined as a subtraction, followed by calculating the absolute value and an accumulation. In addition, an equal amount of data access times is also required. Clearly this is an extremely challenging task that deserves special attention.

From the above equation, the basic Processing Element (PE) can be constructed (as shown in Figure 7), where L is a latch and

$$f(v,h) = \sum_{i=1}^{16}\sum_{j=1}^{16} \left| a(i,j) - b(i+v, j+h) \right| \qquad (3)$$

The input data a and b are sequences of 16×16 numbers that can be accessed simultaneously from two separate memories. Of course the above PE can be used repeatedly for different values of v and h to find a minimum value of (v,h) within a given tracking range. However, this would require a computation speed and data access time of 1.3 ns per operation that is unlikely to be attained with current technology. Therefore some form of parallelism is necessary to ease the speed and data access requirements yet make it possible to fit the entire circuitry onto a single chip. Several VLSI architectures using linear or quadratic systolic arrays have been recently proposed which appear to be very promising [29–31].

It should also be noted that other criteria such as the minimum squared differences and the minimum cross correlation function can also be used to determine the best match. However, they can not be easily implemented on a single chip at pres-

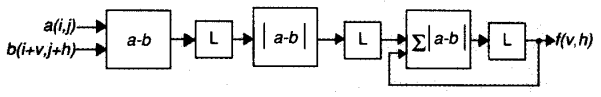


Figure 7. A processing element for motion estimation.

Table I.

Input Symbol	FLC	VLC
a	000	00
b	001	01
c	010	100
d	011	101
e	100	110
f	101	1110
g	110	11110
h	111	11111

ent. If such motion estimators are available in the future, they can be compatibly introduced into encoders.

Variable Length Codec Chip

The last ASIC to be discussed here is the variable length codec chip. A Variable Length Code (VLC) can be used to represent a set of symbols or events with different probabilities of occurrance. If properly designed, a VLC can, on the average, achieve additional compression compared with a Fixed Length Code (FLC). A typical VLC table, together with FLC, is shown in Table I.

The VLC is constructed such that no code word is the prefix of any other code word. This guarantees the unique decodability. Compression is achieved when *a* and *b* occur much more frequently than the rest of the symbols. In this case, only two bits, instead of three, per symbol are needed most of the time. In the CCITT proposed video coding algorithm a large number of VLCs have been used with the 2-D VLC for DCT coefficients being the most complex and extensive. As described earlier, an EVENT in the 2-D VLC is defined as a combination of RUN and LEVEL where LEVEL is the value of a non-zero coefficient with a dimension of 256 and RUN is the number of zeros preceding the non-zero coefficient with a dimension of 64. A straight forward implementation would require a VLC table having more than 16 thousand entries! Since more than 99% of the entries are statistically improbable, they are represented by 14-bit FLCs (6-bit for RUN and 8-bit for LEVEL) with a 6-bit prefix code (ESCAPE) which can be detected and decoded separately. The resulting VLC table would contain only 128 entries, which is much easier to handle. The coding of a VLC is relatively simple. It can be accomplished using a look-up table. However, its counterpart requires some ingenuity because the length of a VLC has to be determined before it can be decoded. Several techniques for VLC decoding have been proposed in the past [32–35]. In particular, a parallel approach employing a barrel shifter and PLAs (Programmable Logic Arrays) or CAM/RAM (Content Addressable Memory/Random Access Memory) modules [35] appear to be suitable for VLSI implementation.

Standardization Activities

Standardization of any particular technology is often viewed as an impedance or obstacle for innovation. However, for video coding, it is a necessary step toward ubiquitous deployment of future video services. Standardization of video coding would not only guarantee interworking of codecs manufactured by different vendors but also would create opportunity for a mass market and stimulate competition. Therefore, the CCITT Specialists Group on Coding for Visual Telephony has been charged with the responsibility of recommending a standard coding algorithm of video signals for digital transmission at px64 kb/s where $p = 1,2,\ldots,30$. These rates were chosen to coincide with the ISDN rates so that the end-to-end international digital connectivity is ensured.

The standardization on coding for visual telephony is well timed. The ISDN transport network needed for low bit-rate video services will soon be deployed. VLSI technology to achieve cost effective codecs is at hand. Some key concepts of the video coding algorithm have recently been invented. The CCITT Specialist Group has been careful to standardize items that are considered necessary and leave others for innovation. For example, the techniques used to implement DCT/IDCT and motion estimation are not subject to standardization. This is also the case for pre- and post-processing of video signals. Consequently, certain proprietary technology can be incorporated into video codecs by different vendors for product differentiation and competition for the market share.

Standardization of video coding would not only guarantee interworking of codecs manufactured by different vendors but also would create opportunity for a mass market and stimulate competition.

The standardization process for video coding is somewhat uncommon in that any proposed item has to be verified and evaluated visually. This is being done first by using computer simulation and then with flexible hardware. Four different moving color test sequences have been adopted by the CCITT Specialists Group for computer simulations. The coded version of the 115th frame of each sequence are shown in Figures 8 to 11 for illustration (printed in black and white only). Unfortunately it is difficult to evaluate picture quality from these figures due to the lack of motion. However, it has been judged by the members of the Specialists Group that the quality of the sequences processed by computer simulation of the proposed algorithm is acceptable for the purpose. Various versions of flexible hardware have also been built and tested for compatibility among codecs as well as for real-time performance evaluation. Since it is flexible, certain parameters of the proposed algorithm can be adjusted for optimization. It is expected that a CCITT Standard on px64 kb/s video codec designated as H.261 will be established in mid 1990.

Summary

Judging from the favorable conditions and events described earlier, visual telephony using ISDN will soon become a reality. Affordable, good quality videophone services using the ISDN Basic Access should be in high demand in the early 90s. Using advanced signal processing techniques, the digital transmission capacity of a pair of copper wires can be pushed up to 800 kb/s within a Carrier Serving Area (CSA) [36]. Consequently, ISDN Primary Access can be made widely available with two pairs of copper wire. When this happens, wide spread usage of video teleconferencing services is possible. It is also possible to distribute (with some additional processing delay) compressed regular television or movies with VCR quality through such a network. Although it is difficult to predict how much additional improvement one can expect, there is a limit in the transmission capacity a copper-based system can provide. This translates into a limit in the picture quality a low bit-rate video service can achieve. As users begin to rely on video services and develop needs for higher quality pictures, a natu-

Figure 8. Miss America sequence simulation result. Frame number: 115, bit rate: 64 kb/s, frame rate: 10 frames/s, S/N (luminance): 37.7 dB.

Figure 9. Clair sequence simulation result. Frame number: 115, bit rate: 64 kb/s, frame rate: 10 frames/s, S/N (luminance): 38.2 dB.

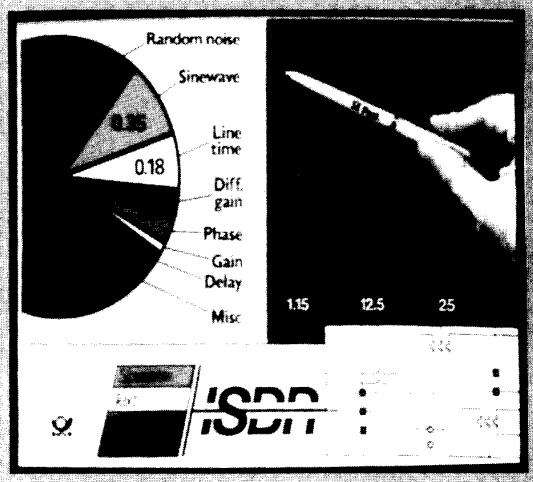


Figure 10. Swing sequence simulation result. Frame number: 145, bit rate: 64 kb/s, frame rate: 10 frames/s, S/N (luminance): 34.5 dB.

Figure 11. Salesman sequence simulation result. Frame number: 115, bit rate: 64 kb/s, frame rate: 10 frames/s, S/N (luminance): 30.9 dB.

ral transition will be to use a fiber-based Broadband ISDN.

From this perspective, visual telephony using ISDN can be viewed as the beginning of a new era in telecommunication services that has evolved from voice to data and finally to video.

Acknowledgments

The author wishes to thank Dr. Ali Tabatabai for providing the simulation results shown in Figures 8 through 11. He is also grateful to Drs. Jules A. Bellisio, T. Russell Hsing, Joseph W. Lechleider and Didier J. Le Gall for carefully reviewing this article.

References

[1] W. K. Pratt, Ed., *Image Transmission Techniques*, Academic Press, New York, 1979.

[2] A. N. Netravali and J. O. Limb, "Picture Coding: A Review", *Proc. IEEE*, vol. 68, no. 3, pp. 366–406, Mar. 1980.

[3] A. K. Jain, "Image Data Compression: A Review", *Proc. IEEE*, vol. 69, no. 3, pp. 349–389, Mar. 1981.

[4] H. G. Musmann, P. Pirsch, and H.-J. Grallert, "Advances in Picture Coding", *Proc. IEEE*, vol. 73, no. 4, pp. 523–548, Apr. 1985.

[5] A. N. Netravali and B. G. Haskell, *Digital Pictures—Representation and Compression*, Plenum Press, New York and London, 1988.

[6] "Description of Reference Model 8 (RM8)", Doc. #525, CCITT SG XV Working Party XV/4, Specialist Group on Coding for Visual Telephony, June, 1989.

[7] H. Gharavi, "Low Bit-Rate Video Transmission for ISDN Application", *IEEE Trans. Circuits and Systems*, vol. CAS-35, no. 2, pp. 258–26', Feb. 1988.

[8] H. Gharavi, "Method and Apparatus for Low Bit-Rate Interframe Video Coding", U.S. Patent No. 4,821,119, Apr. 11, 1989.

[9] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete Cosine Transform", *IEEE Trans. Computer*, vol. C-23, pp. 90–93, Jan. 1974.

[10] CCITT Recommendation G.711, *CCITT Blue Book*, vol. III, fasc. III.6, Melbourne, Nov. 1988.

[11] CCITT Recommendation G.722, *CCITT Blue Book*, vol. III, fasc. III.4, Melbourne, Nov. 1988.

[12] E. A. Lee, "Programmable DSP Architectures: Part I", *IEEE ASSP Magazine*, vol. 5, no. 4, pp. 4–19, Oct. 1988

[13] E. A. Lee, "Programmable DSP Architectures: Part II", *IEEE ASSP Magazine*, vol. 6, no. 1, pp. 4–14, Jan. 1989.

[14] Inmos: Transputer Reference Manual, Oct. 1986.

[15] N. Ohta, et. al., "Performance of a Multicomputer Type DSP System 'NOVI' for Motion Picture Coding", *Proc. 1988 IEEE Int'l. Symp. on Circuits and Systems*, pp. 2305–2308, June 1988.

[16] I. Tamitani, et. al., "A Real-Time Video Signal Processor Suitable for Motion Picture Coding Applications", *IEEE Trans. Circuits and Systems*, vol. CAS-36, no. 10, pp. 1259–1266, Oct.1989.
[17] K. Ano, M. Toyokura, and T. Araki, "A 30 ns (600 MOPS) Image Processor with a Reconfigurable Pipeline Architecture", *Proc. IEEE 1989 Custom Integrated Circuits Conf.*, pp. 24.4.1–4, May, 1989.
[18] S. R. Barraclough, et. al., "The Design and Implementation of the IMS A110 Image and Signal Processor", *Proc. IEEE 1989 Custom Integrated Circuits Conf.*, pp. 24.5.1–4, May 1989.
[19] C. M. Huizer, et al., "A Programmable 1400 MOPS Video Signal Processor", *Proc. IEEE 1989 Custom Integrated Circuits Conf.*, pp. 24.3.1–4, May 1989.
[20] A. H. M. van Roermund, et. al., "A General Purpose Programmable Video Signal Processor", *IEEE Trans. Consumer Electronics*, vol. 35, no. 3, pp. 249–258, Aug. 1989.
[21] T. Murakami, et. al., A DSP Architectural Design for Low Bit-Rate Motion Video Codec", *IEEE Trans. Circuits and Systems*, vol. CAS-36, no. 10, pp. 1267–1274, Oct. 1989.
[22] W. H. Chen, C. H. Smith, and S. C. Fralick, "A Fast Computational Algorithm for the Discrete Cosine Transform", *IEEE Trans. Commun.* vol. COM-25, pp. 1004–1009, Sept. 1977.
[23] B. G. Lee, "A New Algorithm to Compute Discrete Cosine Transform", *IEEE Trans. Acoustic, Speech, Signal Process.* vol. ASSP-32, no. 6, pp. 1243–1245, Dec. 1984.
[24] F. Jutand, et. al., "A 13.5 MHz Single Chip Multiformat Discrete Cosine Transform", *Proc. SPIE Symp. on Visual Commun.* vol. 845, pp. 6–12, Oct. 1987.
[25] M. Vetterli and A. Ligtenberg, "A Discrete Fourier-Cosine Transform Chip", *IEEE J. Select Areas Commun.* vol.SAC-4, pp. 49–61, Jan. 1986.
[26] M. T. Sun, L. Wu, and M. L. Liou, A Concurrent Architecture for VLSI Implementation of Discrete Cosine Transform", *IEEE Trans. Circuits and Systems*, vol. CAS-34, pp. 992–994, Aug. 1987.
[27] M. L. Liou, M. T. Sun, and L. Wu, "Two-Dimensional Discrete Cosine Transform Processor", U.S. Patent 4,791,598, Dec. 13, 1988.
[28] M. T. Sun, T. C. Chen, and A. M. Gottleib, "VLSI Implementation of a 16 x 16 Discrete Cosine Transform", *IEEE Trans. Circuits ansd Systems*, vol. CAS-36, no. 4, pp. 610–617, Apr. 1989.
[29] T. Komarek and P. Pirsch, "Array Architectures for Block Matching Algorithms", *IEEE Trans. Circuits and Systems*, vol. CAS-36, no. 10, pp. 1301–1308, Oct. 1989.
[30] L. D. Vos and M. Stegherr, "Parameterizable VLSI Architectures for the Full-Search Block-Matching Algorithm", *IEEE Trans. Circuits and Systems*, vol. CAS-36, no. 10, pp. 1309–1316, Oct. 1989.
[31] K. M. Yang. M. T. Sun, and L. Wu, "A Family of VLSI Designs for the Motion Compensation Block-Matching Algorithm", *IEEE Trans. Circuits and Systems*, vol. CAS-36, no. 10, pp. 1317–1325, Oct. 1989.
[32] Cocke, et. al., "Method and Apparatus for Decoding Variable-Length Codes Having Length-Indicating Prefixes", U.S. Patent no. 3,701,111, Oct. 1972.
[33] M. E. Lukacs, "Variable Word Length Coding for High Data Rate DPCM Video Codes", *Proc. of Picture Coding Symp.*, pp. 54–56, Tokyo, Japan , Apr. 1986.
[34] J. W. Peake, "Decompaction", *IBM Technical Disclosure Bulliten*, vol. 26, no. 9, pp. 4794–4797, Feb. 1984.
[35] M. T. Sun, K. M. Yang, and K. H. Tzou, "High-Speed Programmable ICs for Decoding of Variable-Length Codes", Applications of Digital Image Processing XII, Andrew Tescher, Ed., *Proc. SPIE* vol. 1153, Aug. 1989.
[36] J. W. Lechleider, "Line Codes for Digital Subscriber Lines", *IEEE Comm. Mag.*, vol. 27, no. 9, pp. 25–32, Sept. 1989.

Biography

Ming L. Liou received his B.S. from National Taiwan University in 1956, M.S. from Drexel University in 1961, and Ph.D. from Stanford University in 1964, all in electrical engineering.

He is presently District Manager of the Signal Processing Systems Research Group at Bellcore. The current research activities of the Gruop now include low-bit-rate voice and video, image compression techniques, Advanced Television, VSLI architecture and implementation of signal processing systems for audio and visual applications. Prior to this, Dr. Liou and his group were heavily involved in the digital subscriber line transceiver research for ISDN Basic Access.

Dr. Liou joined AT&T Bell Labs in 1963 as a Member of the Technical Staff and had held various supervisory positions until 1984 when he joined Bellcore. During his twenty-one year career at AT&T Bell Labs, Dr. Liou worked on the analysis and computer aided design of various transmission components, circuits, terminals, and systems. He has published over forty technical papers including two award winning papers in the IEEE Transactions on Circuits and Systems.

Dr. Liou served as Editor of the *IEEE Transactions on Circuits and Systems* from 1979 to 1981 and the President of the IEEE Circuits and Systems Society in 1988. He is a Fellow of IEEE and a member of Sigma Xi and Eta Kappa Nu.