VIDEO TELEPHONY

With the developments in modem technology and audio and video compression, real-time transmission of video and audio over the telephone network, also known as general switched telephone network (*GSTN*) or Public Switched Telephone Network (*PSTN*), is possible. The International Telecommunications Union (ITU) Recommendation V.92 takes advantage of the direct digital connection of an Internet service provider (ISP) to attain 56 kbits/s downstream (from the ISP to the user). For the upstream (from a user to the ISP), although the user has an analog connection to the telephone network, V.92 takes advantage of the pulse coded modulation used in digital telephone networks to attain upstream speed above 40 kbits/s [1], [2]. In this article, we concentrate on video and audio transmission over PSTN. Voice compression techniques can allow the transmission of speech at a rate of about 6–8 kbits/s with a quality that is comparable to that of 64 kbits/s pulse code modulation that is used in standard digital telephone networks. Video compression standards like H.264/AVC [21, 22] allow transmission of high quality video more efficiently than older video coding standards. Apart from H.264/AVC, the earlier video coding standard H.263 is popularly used in video telephony due to its ability to send acceptable quality video with bit rates below 20 kbits/s. Thus, real-time two-way video phone communication over the GSTN is feasible. This has led to the development of multimedia transmission standards like H.324. These standards can be implemented using general-purpose personal computers or specialized hardware.

In this article, we cover the fundamental technologies used for video phones. Since H.324 is the standard for video telephony over the GSTN, we use it to illustrate the underlying theories.

The Modem

Digital data cannot be transmitted directly over a telephone line. The reason for this is that the telephone network was originally designed for voice transmission only. Thus, the data have to be transformed to a form that is suitable for transmission over the telephone network. The device which accomplishes this is called a *modem*. The procedure that is performed at the transmitting end is called *modulation*. At the receiving end, the modulated signal has to be transformed back to digital data. This procedure is called *demodulation*. Since the communication is two-way, the modem has to be able to do both jobs. Actually, this is how modem got its name: *mo*dulation–*dem*odulation.

As mentioned earlier, the V.92 modem standard of the International Telecommunications Union–Telecommunications Standardization Sector (*ITU-T*) Recommendation takes advantage of the direct digital connection of an Internet service provider (ISP) to attain 56 kbits/s downstream (from the ISP to the user). However, for upstream the V.92 takes advantage of the pulse coded modulation connections to attain upstream speed above 40 kbits/s, which is higher than the 33.6 kbits/s, that was available using the older V.90 and V.34 standards. The actual bit rate achieved is highly dependent on the quality of the telephone connection and the capabilities implemented in the particular modems. At the beginning of the connection, the modems test the line and jointly agree on a bit rate. If the maximum bit rate of 56 kbits/s is not feasible, a lower bit rate is agreed upon and used. Clearly, in video telephony, users will typically not be connected to the telephone network digitally (like some ISPs), so a bit rate of 56 kbits/s will not be available. For these users, the available bit rate can be over 40 kbits/s, if V.92 is used, or up to 33.6 kbits/s, if V.90 or V34 is used. More information about the V.92 modem standard can be found in Refs. 1 and 2.

Video Compression

It is well known that the representation of images in digital form requires a large amount of bits. For example, the representation of true color images requires 24 bits or 3 bytes per picture element (pixel). Thus, a 1024×1024 pixels color image requires 3 Mbytes of memory! In video telephony, much smaller image formats are used, such as QCIF, which corresponds to a size of 176×144 pixels. Even then, such an image would require $176 \times 144 \times 3 = 76{,}032$ bytes or 608,256 bits. If we want to transmit this information over a telephone line using V.92 modems (assuming a 40 kbits/s connection), this would take $608{,}256/40{,}000 = 15.2$ seconds! Clearly, in order to transmit image and video information over a telephone line, we need to reduce the amount of bits to be sent. This is the task of video compression. Compression is concerned with the minimization of the bits needed to represent digital data.

Compression techniques are divided into two categories: lossless and lossy. With lossless compression, no information is lost and the original image can be reconstructed exactly; while with lossy compression, information is lost and the original image cannot be recovered exactly. Lossy compression is far more efficient than lossless compression in reducing the number of bits needed for the representation of an image. The human eye can tolerate certain image imperfections, such as slight loss of chromatic information. However, other types of imperfections, such as edge blurring and image sequence flickering, are unacceptable. In this article, we will concentrate on lossy compression for both image and audio data. More information about image processing and image compression can be found in Refs. 3–5. Information on speech processing and compression can be found in Ref. 6.

Clearly, in most cases the intensity values of pixels that are close to each other in an image are highly correlated. Thus, some of the information that is contained in the image is redundant. In image and video compression we use techniques to reduce this spatial redundancy. Another type of redundancy found in digital video is temporal redundancy. We expect that two consecutive frames look quite similar. Thus, coding each one of them independently is not efficient. Video compression standards use methods to reduce both the spatial and temporal redundancy.

In motion-compensated video compression, what we try to do is code the motion that is present in the scene. Figure

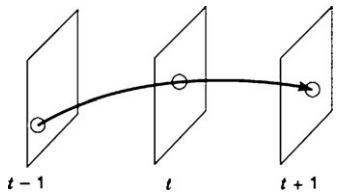

Figure 1. Motion-compensated video compression codes the motion that is present in the scene.

1 shows an object (a circle) which appears in three consecutive frames, but in different places. In this very simple case where the only object in the picture is a circle, we only need to code this object in the first frame and just code its motion in subsequent frames. This will lead to a much smaller number of bits to represent the sequence than required if each frame is coded independently. However, most video sequences are not so simple. Thus, more elaborate methods have been invented to code the motion efficiently and compensate for the fact that objects and background can appear and disappear from the scene. The motion estimation problem is a challenging one. Although the motion is three-dimensional, we are only able to observe its two-dimensional projection onto each frame. In addition, most motion estimators assume that pure translation can describe the motion from one frame to the next. Because the estimation of the motion is imperfect, we should also code the error that remains after we compensate for the motion.

A very important category of video coding algorithms are block-based algorithms. Such algorithms are used in most standards. It is interesting to note that the same basic algorithm can be used for either low (H.263) or high (MPEG-2) bit rates. Only the details of the algorithm change. In the following, we will describe the basics of the block-based algorithms.

The first frame in the video sequence is coded independently, just like a regular image. This is called an *Intra frame*. The compression is done as follows: First, the image is divided into blocks (usually of size 8×8) and the discrete cosine transform (*DCT*) of each block is taken. The DCT maps each block to another 8×8 block of real number coefficients. This is done to decorrelate the pixels. Then, the DCT coefficients are quantized; that is, they are represented by a finite set of values, rather than the whole set of real numbers. Clearly, information is lost at this point. Finally, the quantized coefficients are coded into bits using a lossless variable length coder and the bits are transmitted.

After the first frame, subsequent frames are usually coded as *Inter frames;* that is, interpicture prediction is used. This technique uses the same concept as differential pulse code modulation. A prediction image is formed based on the previously reconstructed frame and some information on the relationship between the two frames, which is provided by the motion vectors. The difference between the actual frame and the prediction is then coded and transmitted.

At the beginning of the transmission of an Inter frame, the picture is divided into blocks (usually of size 16×16). Unless there is a scene change in the picture, we expect that each of the blocks in the current picture is similar to another 16×16 block in the previously reconstructed frame. Clearly, these two blocks do not have to be in the same spatial position. Thus, the coder uses a technique called *block matching* to identify the block in the previous frame that best matches the block in the current frame. The search is performed in the vicinity of the block in the current frame. Usually, the allowed displacement is up to 15 pixels. Block matching returns a vector which shows the relative spatial location of the best matching block in the previous frame. This vector is called the *motion vector*. Thus, we obtain a motion vector for each 16×16 block in the current frame. Now, we can obtain a prediction of the current frame using the previously reconstructed frame and the motion vectors. The prediction for each block of the current frame will be the block which is pointed to by the corresponding motion vector. Then, the prediction error—that is, the difference between the actual frame and the prediction—is coded using a method similar to the one described above for the Intra frames. The prediction error is transmitted along with the motion vectors. The decoder reconstructs the prediction error and builds the prediction image using the previous frame and the motion vectors. Finally, it produces the reconstructed frame by adding the prediction error to the predicted frame. If the prediction is good, we expect that the coding of the prediction error will require far less bits than the coding of the original picture. However, in some cases, the prediction for some blocks fails. This is the case when new objects are introduced in the picture or new background is revealed. In that case, certain blocks or the entire picture can be coded in Intra mode.

Speech Compression

The default speech compression standard for the multimedia communication standard H.324 is ITU Recommendation G.723.1. This standard belongs to a class of linear prediction analysis-by-synthesis coders. It can operate at two different bit rates, 6.3 and 5.3 kbits/s.

It should be emphasized that G.723.1, like all speech coders, is designed for compression of speech only. It is able to compress music and other audio signals as well, but the sound quality will be much lower. Speech codecs were developed by first understanding how speech is produced and then trying to establish models which can be used in a digital signal processing context to compress the speech signals. These models fail when nonspeech signals are used and, therefore, sound quality is degraded.

Speech is produced when air is expelled from the lungs and the resulting flow of air is passed through the vocal tract. The vocal tract begins at the opening between the vocal cords, which is called the *glottis*, and it ends at the lips. In between, there is the pharynx (the connection from the esophagus to the mouth) and the oral cavity (the mouth).

Speech is the acoustic wave that is radiated from this system when an air flow is perturbed by a constriction somewhere in the vocal tract. Speech sounds are classified into three classes according to their mode of excitation. These classes are voiced sounds, unvoiced sounds, and plosive sounds. In voiced sounds, the excitation is produced by forcing air through the glottis so that quasiperiodic pulses

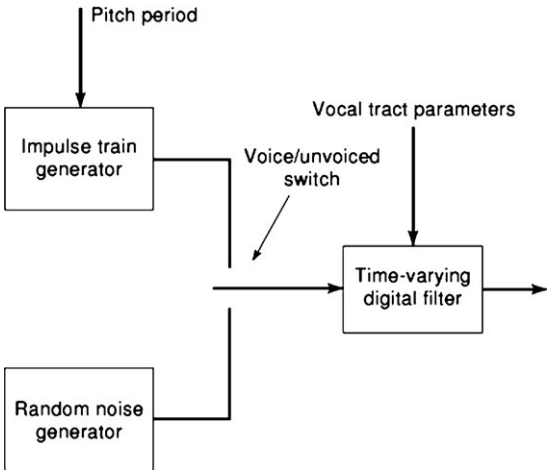

Figure 2. Block diagram of a simplified model for speech production. The excitation signal (impulse train for voiced speech and random noise for unvoiced speech) is input to the time-varying digital filter.

of air are created to excite the vocal tract. In unvoiced sounds, a constriction is formed at some point in the vocal tract, usually toward the mouth end, and air is forced through the constriction at a high enough speed to produce turbulence. Plosive sounds are created by making a complete closure of the vocal tract, building up pressure behind the closure and abruptly releasing it.

Digital speech processing researchers have established a relationship between the physical properties of the vocal tract and digital filters. Thus, they were able to model the vocal tract as an infinite impulse response (*IIR*) digital filter. A digital filter takes numbers as input and produces numbers as output. It performs multiplications and additions using the current and previous input values and, in the case of IIR filters, previous output values. Actually, this filter is an all-pole IIR filter, which means that only previous output values and the current input value are used. Figure 2 shows the block diagram of a simplified model for speech production. As seen in the figure, for voiced speech, we use an impulse train generator as the excitation signal. An impulse train is simply a sequence of samples that is zero except at multiples of the period where it is one. For unvoiced speech, the excitation signal comes from a random noise generator. The parameters of the digital filter vary slowly in time in the same way as the human vocal tract changes in shape. It should be stressed that this is a simplified model for speech production. Reference 5 describes this model in detail.

From the above discussion, we can see that if we know the excitation and the filter parameters, we can use a model that is similar to the above to reproduce the speech digitally. Thus, speech coders try to transmit the excitation and the filter parameters in an efficient way to achieve low bit rates.

MULTIMEDIA COMMUNICATION STANDARDS

Videophone communication consists of video and speech channels. Both of these channels have to share the bandwidth that is made available by the modem. Standards like H.263 and G.723.1 are defined as if they are the sole user of the communication channel. Thus, there should be defined in a way in which all channels can be transmitted simultaneously. There are also other things that have to be taken care of, such as audio–video synchronization, the transmission of control information, and error protection. All of these are addressed by multimedia communication standards. The standards also accommodate other types of channels, such as data channels and control channels. The current standard for low bit rate multimedia communication is ITU-T Recommendation H.324 (7, 8). Other standards exist for different applications. For example, H.310 is intended for asynchronous transfer mode networks and H.323 is designed for packet-switched local area networks. H.323 can also be used over any packet data network, such as the ones using the Internet protocol (*IP*). Thus, it can be used for multimedia communication over the Internet.

As mentioned earlier, a standard is needed that can combine video, audio, control, and other channels into a single bit stream which is to be transmitted over the modem. This procedure is called *multiplexing* and H.324 uses the H.223 multiplexing standard. The goal of this standard is to combine low multiplexer delay with high efficiency and the ability to handle bursty data traffic from a variable number of sources.

The H.245 multimedia system control protocol is used by H.324. The control channel carries messages governing operation of the H.324 system including capabilities exchange, mode preference requests, and opening and closing of logical channels. These are unidirectional bit streams with defined content which are identified by a unique number. There is always one control channel in the H.324 system. There can be any number of video, audio, and data logical channels.

THE H.263 VIDEO COMPRESSION STANDARD

The H.263 video compression standard (9, 10) employs the same general principles mentioned previously, but it is targeted at very low bit rates. The basic structure of the H.263 coder is shown in Fig. 3. It can be seen that it is very similar in principle to a DPCM system. The switch indicates the choice between Intra and Inter pictures or macroblocks. In the Intra case, the DCT of the image is taken and the coefficients are quantized. In the Inter case, the difference between the predicted and the actual picture is transmitted along with the motion vectors. It can be seen from the picture that the encoder actually includes the decoder. This is because the prediction has to be made using information that is available to the decoder. Thus, every encoded frame has to be decoded by the coder and be used for the prediction of the next frame instead of the original frame, since the actual frame is not available to the decoder. Finally, in both the Intra and Inter cases, the quantized coefficients are variable length coded.

Basic Structure of H.263

Pictures are made available to the coder as luminance and two color difference components (Y, C_R, and C_B). H.263 sup-

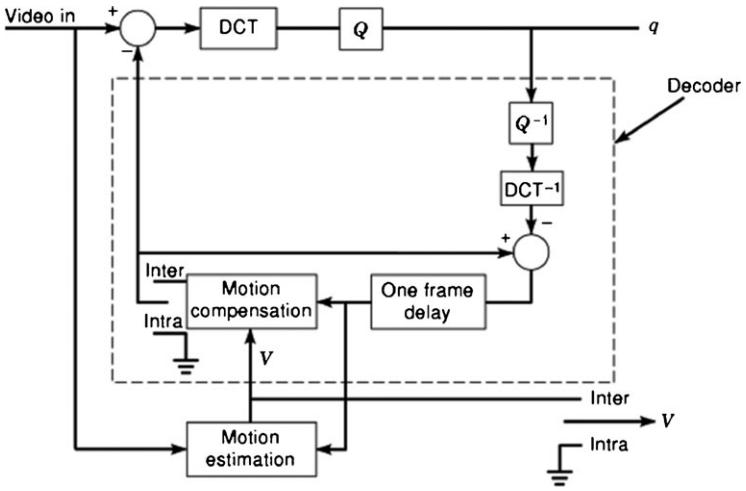

Figure 3. The basic structure of the H.263 coder. For Intra pictures, the DCT of the image is taken and the coefficients are quantized. For Inter pictures, the difference between the predicted and actual picture is used instead of the actual picture, and motion vectors are also transmitted. Variable length coding or arithmetic coding is used to encode the quantized DCT coefficients and motion vectors before transmission.

ports five standardized picture formats: sub-QCIF, QCIF, CIF, 4CIF, and 16CIF. For each of these formats, the luminance picture consists of dx pixels per line and dy lines per picture. Each of the color difference components consists of $dx/2$ pixels per line and $dy/2$ lines per picture. This means that the color difference components have been subsampled by a factor of two in both dimensions. This is done because, as mentioned previously, the human eye is not as sensitive to slight losses of chromatic information as it is to losses of luminance information. Thus, the chrominance information can be subsampled before it is made available to the coder. Table 1 gives the values of dx, dy, $dx/2$, and $dy/2$ for each of the picture formats.

Each picture is divided into groups of blocks (*GOB*s). A GOB consists of $k \times 16$ lines, depending on the picture format: $k = 1$ for sub-QCIF, QCIF, and CIF; $k = 2$ for 4CIF; and $k = 4$ for 16CIF. Each GOB is divided into macroblocks. A macroblock relates to 16 pixels by 16 lines of Y and to the spatially corresponding 8 pixels by 8 lines of C_R and C_B. It also corresponds to four luminance blocks and the two spatially corresponding color difference (chrominance) blocks. Each luminance or chrominance block consists of 8 pixels by 8 lines of Y, C_B, and C_R.

There are two main coding modes in H.263 depending on whether prediction is used or not. The coding mode in which prediction is applied is called INTER. The mode in which no prediction is applied is called INTRA. The INTRA coding mode can be signaled at the picture level (INTRA for I-pictures or INTER for P-pictures) or at the macroblock level for P-pictures. That is, a picture designated as INTER can contain some INTRA blocks if necessary. It should also be noted that the coder has the option not to code a specific macroblock if it determines that it is advantageous to do so. There also exists an optional PB-frames mode in which a PB-frame consists of two pictures being coded as one unit. Its name comes from the names of picture frames in recommendation H.262 (MPEG2) where there are P-pictures and B-pictures. Thus, a PB-frame consists of a P-picture which is predicted from the previous decoded P-picture or I-picture and one B-picture which is predicted from both the previous decoded P-picture and the P-picture which is currently being decoded. The "B" in B-pictures means "Bi-directional" because B-pictures may be bi-directionally predicted from the past and future pictures. B-pictures are always coded in INTER mode.

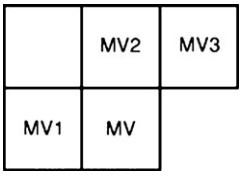

Figure 4. The candidate predictors for the differential coding of the motion vectors.

Motion Compensation

The coder will produce one motion vector per macroblock, or, if the advanced prediction mode is used, one or four motion vectors per macroblock. If the PB-frames mode is used, one additional delta vector can be transmitted for each macroblock in order to adapt the motion vectors for prediction of the B-macroblock. All optional modes will be described later in this article.

The motion vectors are coded differentially due to their redundancy. The difference between the actual motion vector and a predicted motion vector is coded and transmitted. The decoder obtains the actual motion vector by adding the transmitted vector difference to the predicted vector. The predicted vector is the median value of three candidate predictors MV1, MV2, and MV3. The candidate predictors are determined as shown in Fig. 4. Normally, MV1 is the previous motion vector, MV2 is the above motion vector and MV3 is the above right motion vector. However, there are certain exceptions: Candidate predictor MV1 is set to zero if the corresponding macroblock is outside the picture. The candidate predictors MV2 and MV3 are set to MV1 if the corresponding macroblocks are outside the picture. Finally, we check if the above right macroblock is outside the picture. In that case, MV3 is set to zero.

A positive value of the horizontal or vertical component of a motion vector indicates that the prediction is based on pixels in the referenced frame that are spatially to the right or below the pixels being predicted.

The H.263 standard does not specify how the motion vectors are determined. The standard essentially specifies

Table 1. Frame Sizes for Each of the H.263 Picture Formats

Pixel Format	Number of Pixels for Luminance (dx)	Number of Pixels for Luminance (dy)	Number of Pixels for Chrominance ($dx/2$)	Number of Pixels for Chrominance ($dy/2$)
Sub-QCIF	128	96	64	48
QCIF	176	144	88	72
CIF	352	288	176	144
4CIF	704	576	352	288
16CIF	1408	1152	704	576

the decoder and a valid bit stream. It is up to the coder to decide on issues such as the determination of the motion vector, mode selection, and so on.

DCT and Quantization

A separable two-dimensional discrete cosine transform (DCT) of size 8 × 8 is used for each block. The DCT is given by the following formula (3,4,11):

$$F(u,v) = \frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)\cos[\pi(2x+1)u/16]\cos[\pi(2y+1)v/16] \quad (1)$$

with $u, v, x, y = 0, 1, 2, \ldots, 7$, where

$$C(u) = \begin{cases} 1/\sqrt{2} & \text{if } u = 0 \\ 1 & \text{otherwise} \end{cases} \quad (2)$$

$$C(v) = \begin{cases} 1/\sqrt{2} & \text{if } v = 0 \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

x, y are the spatial coordinates in the original block domain and u, v are the coordinates in the transform domain.

For INTRA blocks, the DCT of the original sampled values of Y, C_B, and C_R are taken. For INTER blocks, the DCT of the difference between the original sampled values and the prediction is taken.

Up to this point, no information has been lost. However, as noted earlier, in order to achieve sufficient compression ratios, some information needs to be thrown away. Thus, the DCT coefficients have to be quantized. There is one quantizer for the first coefficient of each INTRA block and 31 quantizers for all other coefficients. Within a macroblock, the same quantizer is used for all coefficients except the first one (the transform dc value) for INTRA blocks. The dc value is quantized with a step size of eight. Each of the other 31 quantizers use equally spaced reconstruction levels with a central dead zone around zero and a step size in the range 2–62.

The quantization parameter (QP) is used to specify the quantizer. QP may take integer values from 1 to 31. The quantization step size is then 2 × QP. QP has to be transmitted along with the quantized coefficients. The following definitions are made:

- /Integer division with truncation toward zero.
- //Integer division with rounding to the nearest integer.

Thus, if COF is a transform coefficient to be quantized and LEVEL is the absolute value of the quantized version of the transform coefficient, the quantization is done as follows:

For INTRA blocks (except for the dc coefficient) we have

$$\mathbf{LEVEL} = |\mathbf{COF}|/(2 \times \mathbf{QP}) \quad (4)$$

For INTER blocks (all coefficients, including the dc) we have

$$\mathbf{LEVEL} = (|\mathbf{COF}| - \mathbf{QP}/2)/(2 \times \mathbf{QP}) \quad (5)$$

For the dc coefficient of an INTRA block we have

$$\mathbf{LEVEL} = \mathbf{COF}//8 \quad (6)$$

The quantized coefficients along with the motion vectors are variable length coded.

Procedures Performed at the Decoder

The following procedures are done at the source decoder after the variable length decoding or arithmetic decoding:

- Motion compensation
- Inverse quantization
- Inverse DCT
- Reconstruction of blocks

Motion Compensation. The motion vector for each macroblock is obtained by adding predictors to the vector differences. In the case of one vector per macroblock (i.e., when the advanced prediction mode is not used), the candidate predictors are taken from three surrounding macroblocks, as described previously. Also, half pixel values are found using bilinear interpolation, as shown in Fig. 5. The integer pixel positions are indicated by the X's and the half pixel positions are indicated by the O's. Integer pixel positions are denoted with the letters A, B, C, and D, and the half pixel positions are denoted with the letters a, b, c, and d. It should be noted that pixel A is the same as pixel a. Then, the half pixel values are

$$\mathbf{a} = \mathbf{A} \quad (7)$$

$$\mathbf{b} = (\mathbf{A} + \mathbf{B} + 1)/2 \quad (8)$$

$$\mathbf{c} = (\mathbf{A} + \mathbf{C} + 1)/2 \quad (9)$$

$$\mathbf{d} = (\mathbf{A} + \mathbf{B} + \mathbf{C} + \mathbf{D} + 2)/4 \quad (10)$$

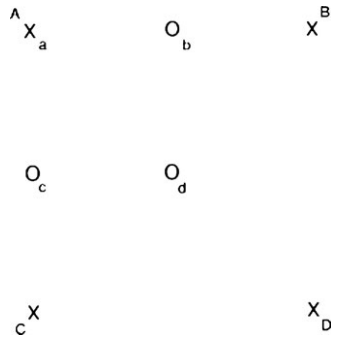

Figure 5. Half-pixel prediction is done using bilinear interpolation.

Inverse Quantization. It should be pointed out that the term "inverse quantization" does not imply that the quantization process is invertible. The quantization operation is clearly not invertible. This term simply implies the process of obtaining the reconstructed transform coefficient from the transmitted quantization level. Thus, if COF′ is the reconstructed transform coefficient and LEVEL is the absolute value of the quantized version of the transform coefficient, the inverse quantization is done as follows:

For INTRA blocks and INTER blocks, except for the dc coefficient, we have

$$|\mathrm{COF}'| = 0 \qquad \text{if } \mathrm{LEVEL} = 0 \tag{11}$$

Otherwise,

$$|\mathrm{COF}'| = 2 \times \mathrm{QP} \times \mathrm{LEVEL} + \mathrm{QP} \qquad \text{if QP is odd} \tag{12}$$

$$|\mathrm{COF}'| = 2 \times \mathrm{QP} \times \mathrm{LEVEL} + \mathrm{QP} - 1 \qquad \text{if QP is even} \tag{13}$$

The sign of COF is then added to obtain |COF′|:

$$\mathrm{COF}' = \mathrm{Sign}(\mathrm{COF}) \times |\mathrm{COF}'| \tag{14}$$

For the dc coefficient of an INTRA block:

$$\mathrm{COF}' = \mathrm{LEVEL} \times 8 \tag{15}$$

Inverse DCT. After inverse quantization, the resulting 8 × 8 blocks are processed by a separable two-dimensional inverse DCT of size 8 × 8. The output from the inverse transform ranges from –256 to 255 after clipping to be represented with 9 bits.

The inverse DCT is given by the following equation:

$$f(x,y) = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7} C(u)C(v)F(u,v)\cos[\pi(2x+1)u/16]\cos[\pi(2y+1)v/16] \tag{16}$$

with $u, v, x, y = 0, 1, 2, \ldots, 7$, where

$$C(u) = \begin{cases} 1/\sqrt{2} & \text{if } u = 0 \\ 1 & \text{otherwise} \end{cases} \tag{17}$$

$$C(v) = \begin{cases} 1/\sqrt{2} & \text{if } v = 0 \\ 1 & \text{otherwise} \end{cases} \tag{18}$$

x, y are the spatial coordinates in the original block domain and u, v are the coordinates in the transform domain.

Reconstruction of Blocks. After the inverse DCT, a reconstruction is formed for each luminance and chrominance block. For INTRA blocks, the reconstruction is equal to the result of the inverse transformation. For INTER blocks, the reconstruction is formed by summing the prediction and the result of the inverse transformation. The transform is performed on a pixel basis.

THE G.723.1 SPEECH COMPRESSION STANDARD

The G.723.1 (12) standard, like almost all recent speech coding standards, belongs to the class of linear prediction analysis-by-synthesis (*LPAS*) coders. Other members of that class are ITU Recommendations G.728 and G.729, as well as the current standards for digital cellular telephony in Europe (GSM) and North America (TDMA and CDMA).

Figure 6 shows a block diagram of an LPAS coder. LPAS coders use a model similar to the one described in the introduction for speech production. In this case, we have two filters instead of one: a long-term (*LT*) predictor synthesis filter and a short-term (*ST*) predictor synthesis filter. The decoded speech is produced by filtering the signal produced by the excitation generator through those two filters. The excitation signal is found by minimizing the mean-squared error over a block of samples. The error is simply the difference between the original and decoded signals. This means that the error for each sample is squared and the results are averaged over the block of samples.

As it can be seen in the figure, the error is weighted by passing through a weighting filter. The reason for this is that the auditory system is less sensitive to distortion which occurs at frequencies where the speech signal has high energy. If we do not use a weighting filter, minimization of the mean-squared error results in equal distribution of the error energy over all frequencies. This is not desirable since we can tolerate more error energy in frequency bands where the speech energy is high but we do not want a lot of error energy in frequency bands where the speech energy is low. The reason for this is that the speech signal will be "buried" by the noise in frequencies where the speech signal energy is not much larger than the error energy. In order to achieve the desired distribution of the error energy in the frequency spectrum, we pass the error signal through the weighting filter which amplifies the error signal in frequencies where the speech signal has low energy and attenuates it in frequencies where the speech signal has high energy. This makes the minimization procedure to shape the error signal energy as desired.

The short-term (ST) predictor filter is the IIR filter discussed in the introduction and models the short-term correlations (spectral envelope) in the speech signal. The predictor coefficients are determined from the signal using linear prediction techniques. The coefficients are adapted in time with rates varying from 30 to 400 times per second.

The long-term (LT) predictor filter models the long-term correlations (spectral fine structure) in the speech signal. Its parameters are a gain coefficient and a delay. For peri-

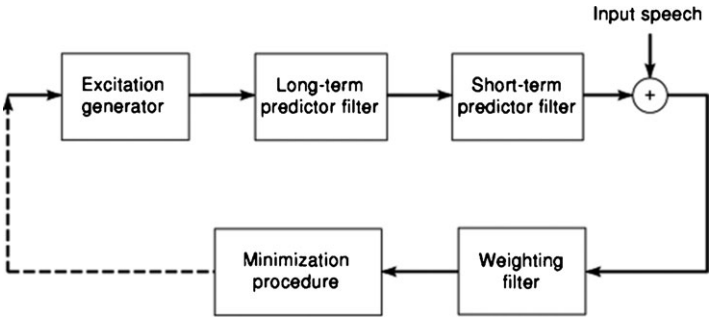

Figure 6. Block diagram of an LPAS coder. A long-term predictor (*LT*) and a short-term predictor filter (*ST*) are used. The filter coefficients are found using a minimization procedure. The excitation information along with the filter parameters are transmitted.

odic signals, the delay corresponds to the pitch period or an integral number of pitch periods. For nonperiodic signals, the delay is random. The LT predictor filter parameters are also adapted in time. The adaptation rates vary from 100 to 200 times per second. In many implementations, the LT predictor filter is replaced by an adaptive codebook. This codebook contains the previous excitation at different delays. The members of the codebook are searched and the one that provides the best fit is selected. Also, an optimal scaling factor is selected.

Clearly, we need to transmit the excitation information along with the filter parameters. We would like to do that using as few bits as possible. G.723.1 uses code-excited linear predictive (*aceLP*) coding to reduce the number of bits required for the transmission of the excitation information. Both the encoder and the decoder store the same set of C possible excitation sequences of length L in a codebook. Thus, the excitation for each frame is completely specified by an index to a vector of the codebook. This index can be found using an exhaustive search over all possible codebook vectors. We select the vector which produces the smallest error between the original and decoded signals.

As mentioned previously, G.723.1 can work in a 5.3 kbits/s mode and a 6.3 kbits/s mode. The 5.3 kbits/s mode uses algebraic codebook-excited linear prediction (*ACELP*). Only a few nonzero unit pulses are used in each codebook vector. Thus, more efficient search procedures can be used for the determination of the best vector instead of an exhaustive search.

In the 6.3 kbits/s mode, multipulse excitation with a maximum likelihood quantizer (*MP-MLQ*) is used. The frame positions are grouped into even-numbered and odd-numbered subsets. A sequential multipulse search is used to find the best vectors for the even-numbered subsets and the odd-numbered subsets. The set that results in the lowest total distortion is selected for the excitation. More information about the current speech compression standards can be found in Ref. 13.

THE H.324 MULTIMEDIA COMMUNICATION STANDARD

H.324 is the ITU-T standard for low bit rate GSTN networks (7, 8). The basic structure of H.324 is shown in Fig. 7. The basic parts of the standard are a multiplexer which mixes the various media types into a single bit stream (H.223), an audio compression algorithm (G.723.1), a video compression algorithm (H.263), and a control protocol which performs automatic capability negotiation and logical channel control (H.245).

Multiplexing

H.324 uses a multiplex standard, H.223, to mix the various streams of audio, video, data and the control channel together into a single bit stream for transmission over the V.92 modem [1].

Time division multiplexers (*TDM*) were considered unsuitable for H.324 because they cannot adapt to dynamically changing modem and payload data rates easily. Also, they are difficult to implement using software since they require complex frame synchronization and bit-oriented channel allocation.

These problems are avoided by using packet multiplexers. However, those suffer from blocking delay, where transmission of urgent data is delayed because the transmission of a large packet which has already started has to be completed first.

The H.223 multiplexer combines the best features of both TDM and packet multiplexers. It incurs less delay than them, has low overhead, and is extensible to multiple channels of each data type. H.223 consists of a lower multiplex layer and a set of adaptation layers. The lower multiplex layer mixes the different media streams, whereas the adaptation layers perform logical frame, sequence numbering, error detection, and error correction by retransmission. Each adaptation layer is suitable for a different type of information channel. There are three adaptation layers in H.324, AL1, AL2, and AL3. AL1 is intended primarily for variable-rate framed information. AL2 is primarily intended for digital audio and includes an 8 bit cyclic redundancy code (*CRC*). CRC is used to identify transmission errors. AL3 is primarily intended for digital video and includes provision for retransmission and a 16 bit CRC.

The Control Channel

The H.245 multimedia system control protocol is used by H.324. The control model of H.324 is based on the concept of logical channels. These are independent unidirectional bit streams with defined content. Each one is identified by a unique number which is arbitrarily chosen by the transmitter. There may be up to 65,335 logical channels. One of these channels is the control channel. There is exactly one control channel in one direction within H.324. It is carried on logical channel 0, a separate channel out-of-band from the various media streams. Logical channel 0 is considered already open when the H.324 starts up.

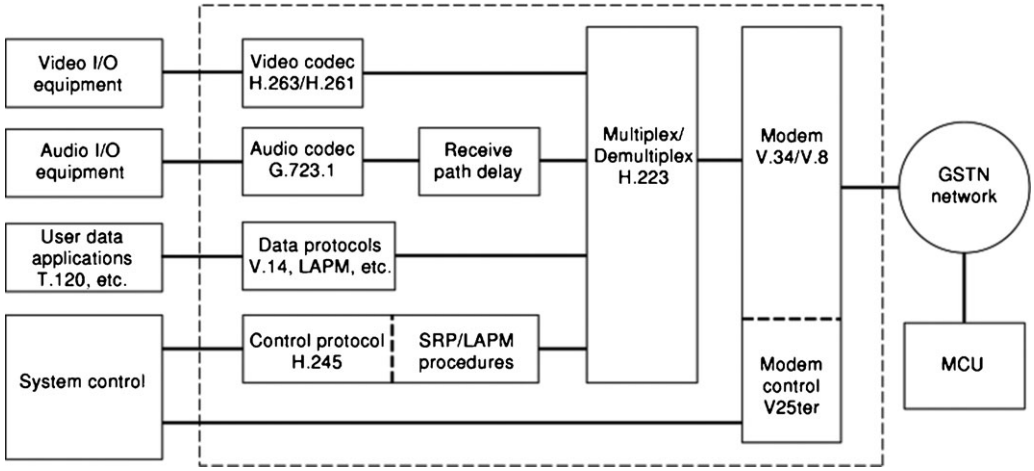

Figure 7. The basic structure of H.324. The basic parts of the standard are a multiplexer which mixes the various media types into a single bit stream (H.223), an audio compression algorithm (H.263), and a control protocol which performs automatic capability negotiation and logical channel control (H.245).

The control channel carries end-to-end control messages governing the operation of the H.324 system, including capabilities exchange, opening and closing of logical channels, mode preference requests, multiplex table entry transmission, flow control messages, and general commands and indications.

Video Channels

H.324 can send color motion video over any desired fraction of the available modem bandwidth. The H.324 standard uses H.263 as the preferred video coding algorithm, but H.261 is also supported to allow Internetworking with ISDN H.320 videoconferencing systems without the need to convert video formats.

H.324 video should achieve 5–15 frames per second, depending on picture format, bit rate, H.263 options in use, and the amount of complexity and movement in the scene. An H.245 control message allows the receiver to specify a preference for the tradeoff between frame rate and picture resolution. Video channels use H.223 adaptation layer 3 (*AL3*), which includes a 16 bit CRC and provision for retransmission at the option of the receiver.

Audio Channels

The baseline audio mode for H.324 is the G.723.1 speech compression standard. This standard supports two bit rates, a high rate at 6.3 kbits/s, and a low rate at 5.3 kbits/s. It provides near-toll-quality speech using a 30 ms frame size and a 7.5 ms lookahead. The audio channel uses adaptation layer AL2, which includes an 8 bit CRC on each audio frame or group of frames.

Audio–Video Synchronization. The H.263 and H.261 video codecs require some processing delay and transmission delay, while the G.723.1 speech codec involves much less delay. Thus, additional delay needs to be added in the audio path to achieve speech synchronization. H.245 is used to send a message containing the average time skew between the transmitted video and audio signals. This skew allows the receiver to calculate the correct audio delay, since the receiver is aware of the decoding delay of each stream.

Call Setup

There are seven phases in the call setup procedure, designated by the letters A through G. In Phase A, an ordinary telephone connection is established. In Phase B, a regular analog telephone conversation can take place before the actual multimedia communication. When either user decides to start the multimedia communication, Phase C takes place. The two modems communicate with each other and digital communication is established. Then, in Phase D, the H.324 terminals communicate with each other using the H.245 control channel. Detailed terminal capabilities are exchanged and logical channels are opened. In Phase E, the actual multimedia communication takes place. Phase F is entered when either user wishes to end the call. The logical channels are closed and a H.245 message is sent to the far-end terminal to specify the new mode (disconnect, back to voice mode, or another digital mode). Finally, in Phase G, the terminals actually enter the mode specified in the previous phase.

COMMERCIAL IMPLEMENTATIONS

There are many commercial video phone implementations. Some of them conform to standards and others are proprietary. Also, some are software-only and can be used with any general-purpose PC, with the addition of course of a camera, a video capture card, and a microphone. Others come with specialized hardware as well.

It should be noted that many of the commercial implementations use the H.323 standard instead of the H.324 standard. This is done because they can be used for communication over the Internet, even though both parties may be connected to the Internet via a telephone line and modem. The point-to-point protocols (*PPP*) are commonly used for TCP/IP communication (the protocol used by the Internet)

over a telephone line. TCP/IP is packet-based and it turns out that significant overhead is required for the packet headers. Thus, some of the bandwidth that is made available by the modem is wasted. However, using the Internet, we can achieve multimedia communication without incurring any long distance charges. The usage of cable modem along with DSL technology has enabled broadband Internet access in many countries. A cable modem is used to deliver broadband Internet access taking the advantage of the unused bandwidth of the cable television network. The bandwidth of the cable connection varies from 3 Mbits/s to 30 Mbits/s and the upstream speed ranges from 384 Kbits/s to 6 Mbits/s. The DSL modem on the other hand takes advantage of the unused frequencies in the telephone line and varies in speed from hundreds of kbits/s to few Mbits/s.

Some companies that currently offer software and/or hardware products for IP based video telephony are: Eyeball Networks, VocalTec (Internet Phone), ADIR VOIP Technologies, Microsoft Corporation (NetMeeting), Intel (Internet Video Phone), Creative Labs, Polycom, 3COM, and Target Technologies.

ADVANCED TOPICS

In this section, we give an introduction to the H.264 video coding standard and briefly cover IP based video telephony.

The H.264/AVC advanced video coding standard

This article has focused on the H.263 video compression standard. Although H.263 is still widely used in video telephony, the H.264/AVC is the latest video coding standard jointly developed by the Video Coding Experts Group (VCEG) of the ITU-T and the Moving Picture Experts Group (MPEG) of ISO/IEC. It uses state of the art coding tools and provides enhanced coding efficiency for a wide range of applications, including video telephony, video authoring, digital camera, etc. It uses Fidelity Range Extensions (FRExt), which provides a number of enhanced capabilities relative to the base specification. It is primarily targeted at providing significant improvements in coding efficiency for higher fidelity video materials. Some of the key techniques that provide this improvement are: Enhanced motion-compensated prediction (MCP), multiple reference pictures and generalized B-pictures, spatial intra prediction, small block-size transform in integer precision, content adaptive in-loop deblocking filter, advanced error robustness and suitability for use in wide variety of networks etc.

H.264/AVC has been developed to address a large range of applications, bit rates, resolutions, qualities, and services. Different applications have various requirements in terms of functionalities, i.e., error resiliency, compression efficiency, delay and complexity. In order to cater to the needs of various applications, H.264/AVC defines profiles and levels. A profile is a set of coding tools. There are three profiles viz. Baseline profile, Main profile and Extended profile. A level is a specified set of constraints imposed on values of the syntax elements in the bit stream.

The reader is referred to [21], [22], [23] for further reading on video coding.

IP Video Telephony

This article has focused on video telephony over the Public Switched Telephone Network. Thus, the available bit rates are, in the best case, in the neighborhood of 40 kbits/s. Nowadays, cable modems and Direct Subscriber Line (DSL) connections are widely available in many countries and offer Internet connections at much higher bit rates. Thus, IP video telephony in conjunction with DSL and cable modems can now offer video communications at a much higher quality than before.

IP video telephony [26] transmits the audio and video data over data networks using the Internet Protocol. Such networks can be the Internet or corporate Intranet. Data networks use packet switching that chops the data into small packets with an address to specify where to send them. Inside each packet is a header and payload. The payload specifies the packet content.

IP Video Telephony uses the H.323 standard that can be used over any packet data network, such as the ones using the Internet protocol (*IP*). Due to the interest in video telephony over IP, many of the existing commercial implementations use the H.323 standard.

Broadband

Broadband refers to the signaling method where the entire frequency range is divided into channels or frequency bins. Using broadband communications, multiple pieces of data can be sent simultaneously to increase the effective rate of transmission. Digital Subscriber Line (DSL) connection and cable Internet connection are both referred as broadband since they use different channels to send the digital information simultaneously with the audio signal or the cable television signal.

The cable Internet service is provided to a neighborhood by a single coaxial cable line. Hence, connection speed varies depending on how many people are using the service. On the other hand, a DSL connection uses a dedicated line from the subscriber to the telephone company. The examples of DSL technology include the High Data Rate Digital Subscriber Line (HDSL), Symmetric Digital Subscriber Line (SDSL), Asymmetric Digital Subscriber Line (ADSL), etc.

Cable modems send the data signal over the cable television infrastructure. They are used to deliver the broadband Internet access, taking advantage of the unused cable network bandwidth. DSL on the other hand uses conventional twisted wire pair for data transmission. ADSL [14] uses two frequency bands known as upstream and downstream bands. The upstream band is used for communication from the end user to the telephone central office while the downstream band is used for communicating from the central office to the end user. ADSL provides dedicated local bandwidth contrary to the cable modem which gives shared bandwidth. Hence, the upstream and downlink speed varies depending on the distance of the end user from the telephone office. Conventional ADSL has downstream speed of approximately 8 Mbits/s and upstream speed around 1 Mbits/s. Thus, improved quality video telephony is possible using the advances in modem technology and audio and video compression. As in the case of dial-up

modems, the upstream speed is the one that constrains the quality of video communications.

Further details can be found in [24], [25].

BIBLIOGRAPHY

1. Enhancements to Recommendation V.90, ITU-T Rec. V.92, 2000.
2. D. Y. Kim, *et al.* The V.92 The last Dial-Up Modem?, *IEEE Transactions on Communications*.Vol.**52**. No. 1,January 2004.
3. A. K. Jain *Fundamentals of Digital Image Processing*, New York: Prentice-Hall, 1989.
4. J. Lim *Two-Dimensional Signal and Image Processing*, New York: Prentice-Hall, 1990.
5. K. R. Rao J. J. Hwang *Techniques and Standards for Image, Video, Audio Coding*, Upper Saddle River, NJ: Prentice-Hall, 1996.
6. L. R. Rabiner R. W. Schafer *Digital Processing of Speech Signals*, New York: Prentice-Hall, 1978.
7. ITU-T Rec. H.324, Terminal for low bitrate multimedia communication, 1995.
8. D. Lindbergh The H.324 multimedia communication standard, *IEEE Commun. Mag*.,December, pp. 46–51, 1996.
9. ITU-T Rec. H.263, Video Codec for low bit rate communication, 1996.
10. K. Rijkse H.263: Video coding for low-bit-rate communication, *IEEE Commun. Mag*.,December, pp. 42–45, 1996.
11. K. R. Rao P. Yip *Discrete Cosine Transform*, San Diego, CA: Academic Press, 1990.
12. ITU-T Rec. G.723.1, Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbits/s, 1996.
13. R. V. Cox P. Kroon Low bit-rate speech coders for multimedia communication, *IEEE Commun. Mag*.,December, pp. 34–41, 1996.
14. K. Maxwell Asymmetric digital subscriber line: Interim technology for the next forty years, *IEEE Commun. Mag*.,October, pp. 100–106, 1996.
15. M. McCandless The MP3 revolution, *IEEE Intelligent systems and their applications*vol,**14**, no. 3,May-June 1999, Page(s) 8–9.
16. S. Shlien Guide to MPEG-1 audio standard, *IEEE Transactions on Broadcasting*, vol.**40**, no. 4, 1994, Page(s) 206–218.
17. H. B. Kyoung *et al.* Design optimization of MPEG-2 AAC coding, *IEEE Trans.. On Consumer Electronics*, vol.**47**, no. 4,Nov. 2001, Page(s) 895–903.
18. MPEG.Information technology – generic coding of moving pictures and associated audio, part. 3: Audio. International Standard IS 13818-3, ISO/IEC JTC1/SC29 WG11, 1994.
19. MPEG Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbits/s, part-3: Audio. International Standard IS 11172-3. ISO/IEC JTC1/SC29 WG11, 1992.
20. K. Brandenburg and G. Stoll, ISO-MPEG-1 Audio: a generic standard for coding of high quality digital audio. Inn. Gilchrist andCh. Grewin, editors, *Collected Papers on Digital Audio Bit-Rate Reduction*, Page(s) 31–42, AES, 1996.
21. D. Marpe, T. Wiegand and G. J. Sullivan, The H.264/AVC Advanced Video Coding Standard and its applications, *IEEE Communications Magazine*, Vol.**44**, No. 8,Aug. 2006, pp. 134–143.
22. G. Sullivan and T. Wiegand, Video Compression – From Concepts to the H.264/AVC Standard, *in Proceedings of the IEEE (Special Issue on Advances in Video Coding and Deliver)y*, Vol.**93**, No. 1,Page(s) 18–31,January 2005.
23. J. Ostermann, *et al.*, Video Coding with H.264/AVC: Tools, Performance and Complexity, *IEEE Circuits and Systems Magazine*, Vol.**4**, No. 1,Page(s) 7–28,First Quarter 2004.
24. J. Bingham, ADSL, VDSL and Multi carrier Modulation, *Wiley Series in Telecommunications and Signal Processing, John Wiley and Sons, Inc*. 2000, ISBN: 0–47 1-29099-8.
25. H. Kaaranen, *et al.*, UMTS Networks: Architecture, Mobility and Services, *John Wiley and Sons, Ltd*, 2001. ISBN 0471 48654
26. S. Servetto and M. Vetterli, High Bandwidth Internet Video Telephony, *in Proc. of the IEEE Packet Video Workshop,March 2000.*
27. *D. Mayers, Mobile Video Telephony for 3G Wireless Networks, Mc GrawHill, 2005.*

SAURAV K. BANDYOPADHYAY
LISIMACHOS P. KONDI
GUIDO M. SCHUSTER
AGGELOS K. KATSAGGELOS
State University of New York (SUNY) at Buffalo, Buffalo, NY
Hochschule fur Technik, Rapperswil (HSR), Switzerland
Northwestern University, Evanston, IL