Content-Based FGS Coding Mode Determination for Video Streaming Over Wireless Networks

Bin-Feng Hung and Chung-Lin Huang, *Member, IEEE*

***Abstract*—Video streaming is the major subject of Amendment for MPEG-4 and it is developed in response to the growing needs on a video-coding standard for the video communication. The fine-granular scalability (FGS) combined with the temporal scalability addresses a variety of challenging problems in delivering video. The FGS video encoder makes the coding mode decision based on the video content and the current available bandwidth in order to achieve higher perceptual video quality. In this paper, we develop a mode selection method to find the most suitable scalable coding mode from six coding schemes: FGS, FGST, FGS-SE, and FGST with background composition based on the contents of the video sequences.**

***Index Terms*—Background composition, coding mode selection, fine-granular scalability (FGS), scalable video coding, video streaming.**

I. Introduction

REAL-TIME video transmission over the wireless networks has become reality due to the increasing popularity of personal computers and the maturity of network transmission technology. However, the current quality of streaming video over the wireless networks still needs a great deal of improvement before the network video can be accepted as an alternative broadcasting media. The main obstacle in designing such systems is the varying characteristics of the networks (i.e., bandwidth variations, packet loss, and network congestion). To cope with these problems and provide quality-of-service (QoS) guarantees, several scalable coding schemes have been proposed for networks video streaming. One of these techniques is the MPEG-4 fine-granular scalability (FGS) scheme [1]–[3], [14], that provides a new level of abstraction between the encoding and transmission process by supporting signal-to-noise ratio (SNR), spatial, and temporal scalabilities through the enhancement layers.

To ensure a good visual quality for FGS video streams over the networks, different kinds of video are suitable for different kinds of scalable coding schemes and an effective bit allocation needs to be employed that allows the enhancement of specific objects within a video sequence. Thus, a hybrid temporal-SNR FGS scheme [4] and a content-based selective enhancement scheme [5] have been adopted by MPEG-4 as the video streaming standards. However, any single scalable coding scheme is not effective for streaming the video over the networks because the amount of motion-activity and image quality varies considerably on a scene basis. In this paper, we develop a mode decision algorithm that can select the most suitable MPEG-4 FGS related scheme to encode the video sequence for network video streaming. The decision making is based on the features extracted from the base layer bit stream (i.e., the content of the video sequence) and the transmission bandwidth, and it results in the minimal perceptual distortion of the coded video sequences.

Schaar and Radha [4] proposed two strategies based on the PSNR value and the base layer information to determine the temporal-SNR tradeoff. The first method is a simple heuristic rate allocation algorithm, it encodes a video sequence with different frame rates and chooses the most suitable frame rate according to the peak-signal-to-noise ratio (PSNR) values, which is not a good measurement for the perceptual quality of the video sequences. The second method measures the video sequence's temporal activity and texture complexity so that a rate control algorithm may find the best tradeoff between individual image quality and motion smoothness. The video sequences may be coded in different schemes such as FGS or FGST. However, they did not mention how to make decision based on these features nor consider the available transmission bandwidth. Turaga and Chen [8] develop a mode decision method for the coding process such as intra/inter mode decision, and frame skipping etc. Their method (modeled as a binary hypothesis testing problem) is well understood in traditional classification theory.

In this paper, we propose a content-based coding mode determination method to select the most suitable one from the six coding FGS schemes: FGS, FGST, FGS-SE (FGS combined with selected enhancement) and FGST-BC (FGST combined with background composition) for the input coding unit (CU). A coding unit consists of a sequence of consecutive video frames. We extract the spatial and temporal features from the video sequences by using the information that can be easily extracted from each CU. The extracted features are combined with the available transmission bandwidth as a feature vector. We can make the coding mode decision based on the feature vectors extracted from the video sequences. Similar to [8], we model the coding mode decision problem as a hypothesis testing problem which is well-understood as a traditional classification problem. However, the difference is in that we convert the problem of minimization of total cost to a standard maximum-likelihood problem so that the number of decision modes can be extended

Manuscript received October 15, 2002; revised May 1, 2003. This work was supported in part by the National Science Council, Taiwan, R.O.C. under Project NSC 91-2213-E007-006.

B.-F. Hung is with Etron Technology, Inc., HsinChu 300, Taiwan, R.O.C.

C.-L. Huang is with the Electrical Engineering Department, National Tsing Hua University, HsinChu 300, Taiwan, R.O.C (e-mail: clhuang@ee.nthu.edu.tw).

Digital Object Identifier 10.1109/JSAC.2003.815229

to more than two. Besides, we utilize the spatial-temporal distortion metric [7] as the error measurement and apply the decision making for each CU instead of each frame to fully utilize the spatial-temporal characteristics of the video sequence.

II. Overview of the FGS

Three types of techniques have been proposed for FGS in MPEG-4, bit-plan coding of the discrete cosine transform (DCT) coefficients [11], wavelet coding of image residue [12], and matching pursuit coding of image residue [13]. However, the first one has been chosen due to its comparable coding efficiency and implementation simplicity. The basic idea of FGS is to encode a video sequence into a base layer and an enhancement layer. The base layer encoder uses a nonscalable coding to reach the lower bound of the bit-rate range. The enhancement layer encoder codes the difference between the original picture and the reconstructed picture using bit-plane coding of the DCT coefficients. The bit stream of the FGS enhancement layer may be truncated, and the decoder still may reconstruct the video from the incomplete bit stream. To further improve the video quality and the flexibility of the codec, several modifications of FGS have been proposed as follows.

A. FGS With Content-Based Selective Enhancement [5]

FGS coding scheme combined with the adaptive quantization and prioritized transmission of specific regions of a video sequence provides good visual image quality. FGS-based adaptive quantization is achieved through bitplane shifting of the selected macroblocks within an FGS enhancement-layer frame. The encoder shifts up the set of coefficients of the designated macroblock by a number of bitplanes relative to the nonenhanced macroblocks coefficients. This adaptive quantization tool is referred to as selective enhancement (SE), since the selected macroblocks within a frame can be enhanced relatively to the others. The purpose of FGS-SE is to improve the image quality of the selected region at the cost of deteriorating the other regions.

B. Hybrid Temporal-SNR FGS [4]

A limitation of the current FGS implementation is that the frame rate is locked to the original base layer frame rate, independent of the available bandwidth. In [4], a hybrid temporal-SNR scalability scheme is proposed to support: 1) SNR scalability while maintaining the same frame rate; 2) temporal scalability by increasing only the frame rate; and 3) both SNR and temporal scalabilities. In addition to the standard SNR FGS frames, this hybrid structure includes multicast-capable (MC) residual frames in the enhancement layer. Each FGST picture is predicted from the base layer frames that do not coincide temporally with the designated FGST picture, thus, this leads to the desired temporal scalability feature.

C. Temporal Scalability With Background Composition [1], [6]

There are two modes of temporal scalability in MPEG-4: standard mode and the background composition mode. In the second mode, only the motion smoothness of the selected object is enhanced in the enhancement layer and other regions are sacrificed to save the total bit rate. For temporal scalability with background composition, in the enhancement layer, only the selected objects are coded further to achieve higher frame rate than that of the base layer. As shown in Fig. 1, frames "a" and "d" of the enhancement layer are the copies of the corresponding frames in the base layer. Other frames in the enhancement layer are obtained by overlapping the selected objects onto the "background," which is made of the the two base layers of the preceding and succeeding frames.

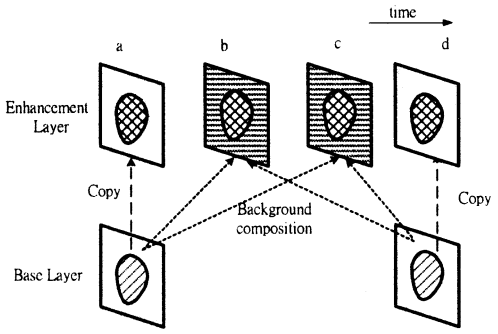


Fig. 1. Basic concept of temporal scalability with background composition.

D. Rate Control for Hybrid Temporal-SNR FGS

The structure of the FGS hybrid temporal-SNR scheme allows the tradeoffs between temporal resolution and SNR improvements for video streaming. The decision is based on the available bandwidth, video sequence's characteristic, and possible user preferences. If the video sequence's motion activity and texture complexity can be characterized by PSNR value or the base layer information, then we can make better temporal-SNR tradeoff.

III. Coding Mode Determination

In video streaming, we need to consider the tradeoff between the image quality of each frame and the temporal smoothness of the sequence based on the contents of the video sequence and the available transmission bandwidth. Furthermore, the content-based MPEG-4 coding scheme also needs to make another tradeoff between the video quality of the foreground objects and the background. The above two concerns can be modeled as a cost minimization problem and the total cost is defined by a certain combination of the perceptual spatial and temporal distortion measurement. Here, we convert a mode decision problem into a hypothesis testing problem.

To decide which coding mode is optimal, we may try all the possible coding modes, evaluate the cost corresponding to each mode, and choose the one with the smallest cost. However, such an exhaustive search approach is impractical for real time implementation due to its complexity. An alternative method is to identify the feature vectors that can be easily computed from the video data and used as good indicators to the optimal mode selection. We want to build a classifier that takes the feature vector (from each input CU) as input and comes up with the probability of the most probable hypothesis, which then enables us to make coding mode selection appropriately.

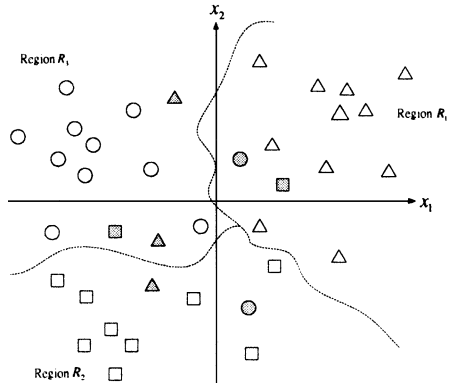


Fig. 2. Partitioning the feature space.

To build a classifier, in the training phase, we collect the feature vectors from the video data and then use an exhaustive search to find their corresponding coding modes that lead to the minimal perceived distortion. Then, we may estimate the probability density function (pdf) for these feature vectors under different hypotheses. The pdfs of the feature vectors under each hypothesis are modeled as a mixture of Gaussians. Once we have these pdf distributions, we use the maximum perceptual ratio test for a new input feature vector (corresponding the input CU) to determine the most likely hypothesis, and then use the result to make the coding mode decision.

A. Transforming Mode Decision Into Classification Problem

A mode decision problem is equivalent to a hypothesis-testing problem. Suppose there are M kinds of decisions $\{D_1, D_2, \ldots, D_M\}$ and let the corresponding costs be $\{C_1, C_2, \ldots, C_M\}$, respectively. The goal of the strategy is to make the decision based on the minimal cost. To make the optimal mode decision, one can try all the coding modes and choose the one that generates the lowest cost. However, computing all the actual costs C_i before making a decision is very computationally intensive as this involves trying all coding modes to determine the minimal cost. So, we need to identify the features that provide a good estimate of the cost for a coding mode determination but with less computation. We make a decision based on the M hypotheses, (e.g., $H_1, H_2, \ldots,$ or H_M), where H_k : if $C_k = \min\{C_1, C_2, \ldots, C_M\}$, $k = 1 \ldots M$.

Having collected a training set of L CUs, we exhaustively compute the costs of L CUs in M coding modes, e.g., $c_{i,j}$, $i = 1 \ldots L$, $j = 1 \ldots M$, where $c_{i,j}$ denotes the cost of assigning CU i to coding mode j. For each CU, we identify K features to form a feature vector $\boldsymbol{X} = [x_1, x_2, \ldots, x_{\mathrm{K}}]^T$. Given the feature vector $\boldsymbol{X}$ the classifier selects the most probable hypothesis.

Assume that we extract two features from each CU in a training set to form a feature vector with two components $\boldsymbol{X} = [x_1, x_2]^T$ and suppose that there are a total of three modes. Then, we can exhaustively compute the costs of all CUs in all three modes, and classify all the CUs into three groups (shown as triangles, squares, and circles in Fig. 2), which represent three hypotheses, H_1, H_2, and H_3 respectively. Our objective is to partition the feature space into three regions, R_1, R_2, or R_3, and make decision D_1, D_2, or D_3, for the CU of which the feature vectors are located. To make the optimal partitions, we need to minimize the total additional cost of each of the misclassified CUs, i.e., the dark symbols in Fig. 2.

The mode decision process can be treated as a problem of the additional cost minimization. It may be written as

$$\mathrm{Min}_{R_1 \ldots R_M} \left[\sum_i d_i \, |_{X_i \in R_l} + \cdots + \sum_i d_i \, |_{X_i \in R_M} \right] \quad (1)$$

where $d_i \; |_{X_i \in R_1} = \; |c_{ij} - c_{i1}| \; |_{j \neq l}$ that indicates the additional cost of the misclassified vector X_i (i.e., CU i) to region $R_j \; |_{j \neq l}$ rather than the correct region R_1. These regions may consist of noncontiguous subregions and the boundaries between them may be arbitrarily shaped. Hence, we formulate the problem of cost minimization in terms of choosing the regions instead of specifying the linear boundaries or the thresholds separating them. Partitioning the feature space to minimize the total cost is similar to the traditional classification problem. In a traditional classification problem, there is a set of probabilities corresponding to each feature vector in the different hypotheses, whereas in our problem, there is a set of costs $\{c_{ij}, i = 1, \ldots, L, j = 1, \ldots, M\}$ corresponding to each CU. Thus, the first step to formulate our problem as a classification problem is to convert the set of cost $\{c_{ij}\}$ into the probability densities.

Since our problem is multimodel classification problem, the cost minimization process is much more complex than the bimodel problem. However, for each CU i, if the cost for selecting mode j is much smaller than selecting the other modes, i.e., $c_{ij} \ll c_{ik}$, for $j \neq k$, then we have more confidence in choosing mode j as the best decision. To simplify the classification process, instead of minimizing the total additional cost [i.e., (1)], we maximize the total cost saving of the correctly classified CUs, which can be written as

$$\mathrm{Max}_{R_1 \ldots R_M} \left[\sum_i |c_{i,\mathrm{worst}} - c_{i,l}| \, \Big|_{X_i \in R_l} + \cdots + \sum_i |c_{i,\mathrm{worst}} - c_{i,M}| \, \Big|_{X_i \in R_M} \right] \quad (2)$$

where $c_{i,\mathrm{worst}}$ is the cost of the worst mode for CU i (with feature vector X_i). Since the cost differences between each two coding modes must be in proportional to the probability density for a certain hypothesis, we may find the pdf of the feature vector X_i by counting its appearance in region R_j, with $|c_{i,\mathrm{worst}} - c_{ij}|$ times (as shown in Fig. 3).

Here, we let

$$N_l = \sum_i |c_{i,\mathrm{worst}} - c_{i,1}| \, \Big|_{Xi \in R_l}, \ldots,$$

$$N_M = \sum_i |c_{i,\mathrm{worst}} - c_{i,M}| \, \Big|_{X_i \in R_M}, \; N = N_1 + \cdots + N_M$$

and the probability of each hypothesis is $P(H_l') = N_1/N, \ldots P(H_M') = N_M/N$. So, we have,

$$\frac{|c_{i,\mathrm{worst}} - c_{i,1}|}{N_l} = p\,(X = X_i \mid H_l'), \ldots, \frac{|c_{i,\mathrm{worst}} - c_{i,M}|}{N_M} = p\,(X = X_i \mid H_M').$$

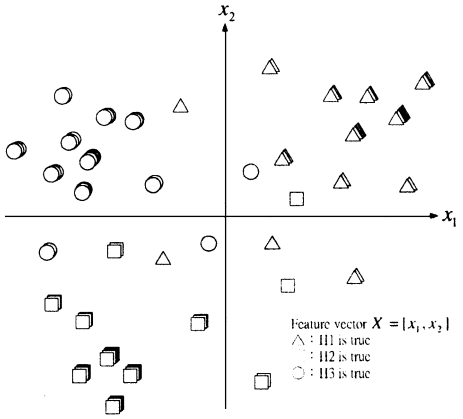


Fig. 3. Repeating the feature vectors by cost difference.

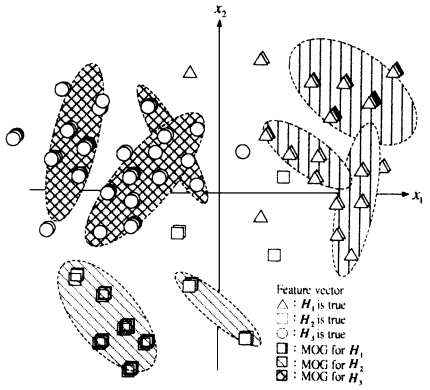


Fig. 4. Modeling the data by mixture of Gaussians.

Then, (2) can be rewritten as:

$$\begin{aligned}\text{Max}_{R_1 \ldots R_M} &\left[\frac{N_1}{N}\sum_i \frac{|c_{i,\text{worst}} - c_{i,l}|}{N_1}\bigg|_{X_i \in R_1} + \cdots \right. \\ &\left. + \frac{N_M}{N}\sum_i \frac{|c_{i,\text{worst}} - c_{i,M}|}{N_M}\bigg|_{X_i \in R_M}\right] \\ =\text{Max}_{R_1 \ldots R_M} &\left[P(H_1')\sum_{X_i \in R_1} p(X = X_i | H_1') + \cdots \right. \\ &\left. + p(H_M')\sum_{X_i \in R_M} p(X = X_i \mid H_M')\right]. \quad (3)\end{aligned}$$

Therefore, we have converted the original problem of minimizing the total cost as shown in (1) into a maximal hypothesis testing problem.

B. Continuous Probability

Since the number of the feature vectors is limited, the conditional pdf of X_i, i.e., $p(X_i \mid H_1')$ is a discrete function. Instead of using these discrete pdfs, we can model these probability data by using a continuous pdf consisting of a mixture of Gaussians.

$$f_k(X) = \sum_{j=1}^{k} \pi_j \psi(X_j; \theta_j),\ \pi_1 + \cdots + \pi_k = 1,\ \pi_j \geq 0$$

where

$$\begin{aligned}\psi(X_j; \theta_j) = (2\pi)^{-k/2} |S_j|^{-1/2} \\ \cdot \exp\left(-0.5\,(X_j - m_{xj})^T S_j^{-1}\,(X_j - m_{Xj})\right).\end{aligned}$$

To classify a feature vector, we need to know the probability of occurrence of that vector. However, the pdf for a new input vector is not known since it may not present in the training data set. By modeling the pdf of the feature vector using a mixture of Gaussians, we ensure that any input feature vector may be effectively classified. These Gaussian mixtures are trained based on the modified feature vectors using the Greedy expectation maximization (GEM) algorithm [9]. Fig. 4 shows the trained Gaussian mixtures in the modified feature space, in which H_1, H_2, and H_3 are modeled by a mixture of Gaussians with 3, 3, and 2 distributions, respectively. Each ellipse represents a Guassian distribution and the center of the ellipse denotes the mean of the Guassian; the diameter of the ellipse denotes the variance of the Gaussian.

Using the continuous pdf, (3) can be rewritten as

$$\begin{aligned}\text{Max}_{R_1 \ldots R_M} &\left[P(H_1')\int_{X \in R_1} p(X \mid H_1')\,dX + \cdots \right. \\ &\left. + P(H_M')\int_{X \in R_M} p(X \mid H_M')\,dX\right]. \quad (4)\end{aligned}$$

The function $p(\cdot)$ corresponds to the continuous pdf comprising a mixture of Gaussians, and it is obvious that the maximization problem in (4) can be simplified as $H_i = \arg \text{Max}_{1 \leq i \leq M}\{P(H_i')p(X \mid H_i')\}$.

Hence, to classify a feature vector X, we calculate the probability of the feature vector assigned to each mode and select the mode with the highest probability. The entire classification scheme may be summarized as follows: 1) given the training data and the cost, we count the feature vector (assigned to mode i) a number of times based on the cost difference between the current mode and the worst mode; 2) we use the GEM algorithm [9] to estimate the *a priori* probabilities $P(H_i')$, as well as the class conditional probability density functions $p(X \mid H_i')$, as the Gaussian mixture; and 3) with these continuous pdfs, we compute the probability of input feature vector assigned to each mode and select the most possible one.

IV. Coding Mode Determination

Here, we apply the classifier to determine which coding mode is the most suitable one for the current input CU. Each CU consists of a video sequence of 30 frames, and the base layer encoder defines a CU as a group of picture (GOP) with 5 frames/s. The quantization errors of DCT coefficients and the skipped frames are considered by the enhancement layer encoder. For different types of video sequence, we apply different types of enhancement layer encoders, e.g., for high motion sequences, we emphasize the temporal smoothness, whereas for low motion and high texture complexity sequences, we highlight the spatial details. Therefore, for different video sequence, we need to develop different enhancement layer encoders. The features of each CU are extracted from the base layer encoder and combined with the current transmission bit rate as a feature vector, which can be used to determine the optimal coding mode.

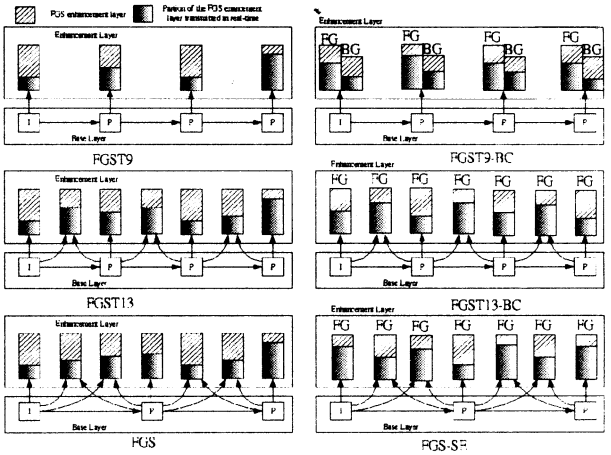


Fig. 5. Illustration of bit rate allocation for six coding modes.

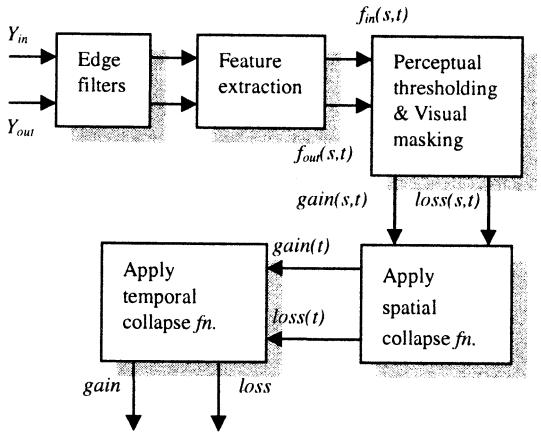


Fig. 6. Perceived distortion metric.

A. FGS Coding Modes

The temporal scalability with background composition is included in MPEG-4 standard [1] for non-FGS temporal scalability, however the background composition [6] has not been included in MPEG-4 standard for FGST. Since this functionality is compatible to FGST and it provides a flexible scheme for bit rate allocation for different objects in the video sequence, we add FGST with background composition as another coding scheme. There are six coding modes: FGS, FGS-SE, FGST9, FGST9-BC, FGST13, FGST13-BC of which the properties are briefly summarized as follows (Fig. 5).

1) FGS. FGS is similar to SNR scalability, and the enhancement layer frame rate is the same as the base layer (5 frames/s), thus, the image quality per frame is improved in the enhancement layer.
2) FGS-SE. The additional property of FGS-SE is that the enhancement layer encoder uses bit plane shifting to improve the image quality of the selected region.
3) FGST9. One B frame is inserted between two base layer frames in the enhancement layer, thus, the total frame rate is 9 frames/s. The temporal smoothness is improved in the enhancement layer, however, the image quality per frame is sacrificed.
4) FGST9-BC. The selected object is coded in the enhancement layer and the background region is formed by background composition. Thus, the spatial-temporal quality of the object is enhanced at the cost of deteriorating quality of the background region.
5) FGST13. Two B frames are inserted between two base layer frames in the enhancement layer, thus, the total frame rate is 13 frames/s.
6) FGST13-BC. The difference between FGST13-BC and FGST9-BC is that the total frame rate is 13 frames/s rather than 9 frames/s.

Fig. 6 shows the bit rate allocation under these modes. The six coding modes provide different tradeoffs between the spatial and the temporal video quality.

B. Cost Function

To measure the perceived quality under each coding mode, we use a spatial-temporal distortion metric [7], from which the distortion is computed by means of comparing the edges statistics in the same spatial-temporal (S-T) region of the test sequence and the reference sequence. To evaluate this metric, first the luminance components of the input and output video streams are processed using the horizontal and vertical edge enhancement filters. These processed streams are partitioned into the S-T regions in which the features that quantify the spatial activity as a function of angular orientation are extracted. These are then clipped to emulate perceptibility thresholds. Distortions due to gains and losses in feature values are calculated using the functional relationships between the input and output feature values that emulate the visual masking. These distortions are then collapsed over space and time. The choice of the edge enhancement filters and the perceptibility thresholds are optimized based on their correlation with the perceptual distortions as shown in Fig. 6. The details of all operation blocks are demonstrated as follows.

1) S-T Region Size: The horizontal and vertical edge enhanced input and output video streams are divided into localized S-T regions. Features are then extracted from each S-T region by calculating statistics over the S-T region, which includes eight horizontal pixels, eight vertical lines and six video frames.

2) Description of Features: For a given image pixel located at row i, column j, and time t, the horizontal and vertical edge enhancement filter responses will be denoted as $H(i,j,t)$ and $V(i,j,t)$, respectively. These responses can be converted into polar coordinates (R,θ) using the relationships

$$R(i,j,t) = \sqrt{H(i,j,t)^2 + V(i,j,t)^2}$$
$$\theta(i,j,t) = \tan^{-1}\left[\frac{V(i,j,t)}{H(i,j,t)}\right]$$

where $(i,j,t) \in \{S-T \text{ region}\}$. The first feature (f_1) is defined as $f_1 = \{stdev[R(i,j,t)]\}|_p$. This feature is computed as standard deviation (*stdev*) over the S-T region and then clipped at the perceptibility threshold of $P(P = 12)$. It is sensitive to

the changes in the overall amount of spatial activity within a given S-T region. The second feature (f_2) is defined as

$$f_2 = \frac{\{\text{mean}[HV(i,j,t)]\}|_p}{\{\text{mean}[\overline{HV}(i,j,t)]\}|_p}$$

where $HV(i,j,t) = R(i,j,t)$ if $R(i,j,t) \geq r_{\min}$ and $m\pi/2 - \Delta\theta < \theta(i,j,t) < m\pi/2 + \Delta\theta$, otherwise, $HV(i,j,t) = 0$, $m = 0,\ldots,3$. Similarly, $\overline{HV} = R(i,j,t) = R(i,j,t)$ if $R(i,j,t) \geq r_{\min}$, and $m\pi/2 - \Delta\theta < \theta(i,j,t) < (m+1)\pi/2 - \Delta\theta$, otherwise $\overline{HV} = R(i,j,t) = 0$. This feature is sensitive to changes in the angular distribution. The parameters are selected as $r_{\min} = 20$, $\Delta\theta = 0.052\,36$, and $P = 4$.

3) Impairment Masking: The gain and the loss of f_1 or f_2 must be examined separately, since they produce fundamentally different effects on the quality of perception (e.g., loss of the spatial activity due to blurring and gain of the spatial activity due to noise of blocking). For a given S-T region, the gain and the loss distortions f_1 or f_2 are defined as

$$\text{gain}(s,t) = pp\left\{\log_{10}\left[\frac{f_{\text{out}}(s,t)}{f_{\text{in}}(s,t)}\right]\right\}$$

$$\text{loss}(s,t) = np\left\{\frac{f_{\text{out}}(s,t) - f_{\text{in}}(s,t)}{f_{\text{in}}(s,t)}\right\}$$

where pp is the positive part operator (i.e., negative values are replaced with zero) and np is the negative part operator (i.e., positive values are replaced with zero).

4) Spatial Collapsing Function: It is computed for each temporal index t as the average of the worst 5% of the measured distortions over the spatial index s, this produces a time history of the gain and loss samples (i.e., $\text{gain}(t)$ and $\text{loss}(t)$) which must then be temporally collapsed. It can be mathematically written as: $\text{gain}(t) = \text{gain}(s_1,t) + \cdots + \text{gain}(s_5,t)$ for $\text{gain}(s_1,t) \geq \cdots\cdots \geq \text{gain}(s_5,t) \geq \cdots \geq \text{gain}(s_{100},t)$ and $\text{loss}(t) = \text{loss}(s_1,t) + \cdots + \text{loss}(s_5,t)$ for $\text{loss}(s_1,t) \leq \cdots \leq \text{loss}(s_5,t) \leq \cdots \leq \text{loss}(s_{100},t)$.

5) Temporal Collapsing Function: It is computed as the mean of the $\text{gain}(t)$ and $\text{loss}(t)$ over the entire CU period (1 second). It can be mathematically written as

$$\text{gain} = \sum_{t=1}^{5}\frac{\text{gain}(t)}{5} \quad \text{loss} = \sum_{t=1}^{5}\frac{\text{loss}(t)}{5}.$$

6) Spatial-Temporal Distortion: It is computed by combining the loss and gain of f_1 and f_2 as

$$D^{S-T} = 0.38 * f_{1\_\text{loss}} + 0.39 * f_{2\_\text{loss}} - 0.23 * f_{2\_\text{gain}}.$$

Furthermore, the selective enhancement and the background composition functionality provide a tradeoff between the video qualities of the background region and the object region under the same transmission bit rate. Thus, the total distortion of the entire video sequence should consist of the distortions of the selected object region and the background region, which is defined as

$$D^{S-T} = \lambda \cdot D_{fg}^{S-T} + (1-\lambda)\cdot D_{bg}^{S-T}$$

where the weighting coefficient λ denotes the visual importance of the selected object in the video sequence. λ is defined as shown in the equation, at the bottom of the page, where the value of λ is clipped at 0.85 (i.e., if $\lambda > 0.85$, then $\lambda = 0.85$).

The perceived distortion metric of the FGS video sequences under different frame skipping rates is called the *temporal distortion*, whereas the perceived distortion metric of the FGS video sequences under the same frame rate but different total bit rates is named the *spatial distortion*. From our experiments, we find that different types of video sequences suffer different amount of perceptual distortion under the same kind of video impairment. Thus, the distortion metric provides a pertinent measurement of the perceptual video quality.

C. Feature Selection

The video impairment mainly consists of the spatial and the temporal distortion. The high motion sequences are sensitive to the temporal distortion; whereas the high texture complexity sequences are sensitive to the spatial distortion. Thus, to characterize the video sequence, the selected features must characterize the motion activity, as well as the texture complexity of the video sequence.

Besides, since the weighting coefficient is λ (it is a function of object's size), and the total bit budget must also be taken into consideration for bit rate allocation, the object size and transmission bandwidth are also essential features for the system to make the coding mode selection. The feature vector is defined as

$$X = [MA_{fg}, TC_{fg}, MA_{bg}, TC_{fg}, Size_{fg}, BW]^T$$

where MA_{fg} and MA_{bg} represent the *temporal (or motion) activity* of foreground and background, respectively; TC_{fg} and TC_{bg} represent the *texture complexity* of foreground and background, respectively; $Size_{fg}$ is the *size* of foreground object; BW is the transmission *bandwidth*.

Since the transmission bit rate is unknown during the video streaming, we assume that the classifier's decision is insensitive to the small variation of transmission rate, and an encoder designed for certain bit rate range (not exactly at a specific bit rate) is reasonable. In this paper, we consider three bit rate ranges as $0 \sim 100$ kb/s, $100 \sim 200$ kb/s, and $200 \sim 300$ kb/s.

To characterize different video sequences, we need to select the features highly correlated with the spatial-temporal distortion of video sequences. We test several features which may represent the *temporal activity* (or the *texture complexity*) of video sequences, and then choose the one with the highest correlation with the spatial-temporal distortion. To perform the frame-dropping (the frame rate of the source video sequence is 5 frames/s), we need to compute the correlation ρ_t between the temporal distortion and the *temporal activity* which is the motion vector magnitude or the frame difference. The correlations of the temporal distortion and either one of the two features representing the *temporal activity* of the video sequences is defined as shown

$$\lambda = \begin{cases} \dfrac{\text{Area of foreground object}}{\text{Entire frame area}} + 0.5, & \text{if foreground object exist} \\ 0, & \text{otherwise} \end{cases}$$

TABLE I
CORRELATION BETWEEN THE TEMPORAL ACTIVITY AND THE TEMPORAL DISTORTION

	Motion Vector Mag.	Frame Diff.
ρ_t	-0.8466	-0.8039

TABLE II
CORRELATION BETWEEN THE SPATIAL COMPLEXITY AND THE SPATIAL DISTORTION

	High Freq. Energy	*Xi* in TM5
ρ_s	-0.8987	-0.8738

in the first equation, at the bottom of the page, where $\mathrm{feature}$ indicates the *temporal activity* (i.e., the motion vector magnitude or the frame difference). The value of correlation ρ_t is shown in Table I.

By encoding source video sequences into 200 kb/s video stream at 30 frames/s, we compute the correlation ρ_s between the spatial distortion and the *texture complexity* (i.e., the high frequency energy or the X_i as defined in TM5). The high frequency energy can be obtained by taking a frame, down-sampling it by a factor of 2 horizontally and vertically, then up-sampling it back to the original size, and finding the energy in the difference between this and originally frame. The correlation of the spatial distortion and either one of the two features representing the texture complexity of the sequences is defined as shown in the second equation, at the bottom of the page, where $\mathrm{feature}$ indicates the *texture complexity* (i.e., high frequency energy or X_i defined in TM5). The value of correlation ρ_s is shown in Table II.

From the experiment results (Tables I and II), we choose *motion vector magnitude* and *high frequency energy* to represent the temporal activity and the texture complexity of the video sequences, because they demonstrate higher ρ_t and ρ_s.

D. The Training Process

We have collected 22 standard MPEG test sequences in QCIF (176×144 format at a frame rate of 30 Hz for the training set and divide them into 500 CUs (30 frames per CU), of which 270 CUs are selected for training and 230 CUs are used for testing. Each CU is down-sampled to meet six different frame rates for six different coding schemes. We encode all CUs in training set for all six coding modes at three different bit-rates (100, 200, 300 kb/s) and extract the feature vectors for each CU at the same time. We assign 5 frames/s for the base layer, so that the bit rate for base layer is fixed at 30 kb/s. To train the classifier, we then decode all CUs in the training set and compute the distortion metric corresponding to each of them. Then, we can group all CUs in the training set into six hypotheses as

$$H_j : C_j = \min\{C_1, C_2, \ldots, C_6\}, \quad j = 1 \ldots 6.$$

To model the mode decision problem as the hypothesis testing problem, we count the appearance of the feature vector, X_i, (with the minimal cost $c_{i,j}$, if it assigned to region R_j) a number of $|c_{i,\mathrm{worst}} - c_{i,j}|$ times, and then compute the pdf of the feature vector X in each hypothesis, $\{p(X \mid H_j),\ j = 1, \ldots 6\}$. Finally, we model the pdfs of all the feature vectors in each hypothesis (a discrete pdf) as a mixture of Gaussians and train this model by using GEM algorithm [9] to obtain the continuous pdfs of the feature vectors of each hypothesis. Thus, given a set of training CUs, we can find all the feature vectors $\{X\}$, compute the conditional probability of every CU for each hypothesis $p(X \mid H_j)$, and the probability of each hypothesis $P(H_j)$. After the training process, for an input CU, we may select the best coding mode by finding the corresponding hypothesis with the highest probability.

V. EXPERIMENTAL RESULTS

We have tested 230 CUs to illustrate the performance of the coding mode selection method. As shown in Table III(a), the success rate of making the best decision is only 63.5%, however, if we allow more additional distortion compared with the best one, then the success rate will be increased. As shown in Table III(b), if we allow more additional distortion, then the

$$\rho_t = \frac{E\left[(\mathrm{feature} - \mu_{\mathrm{feature}})\,(\mathrm{Temporal\_Dist} - \mu_{\mathrm{Temporal\_Dist}})\right]}{\sqrt{\mathrm{var}(\mathrm{feature})\mathrm{var}(\mathrm{Temporal\_Dist})}} = \frac{\sum_{i=1}^{L}\left[(\mathrm{feature}_i - \mu_{\mathrm{feature}}) * (\mathrm{Temporal\_Dist}_i - \mu_{\mathrm{Temporal\_Dist}})\right]/L}{\sqrt{\left[\sum_{i=1}^{L}(\mathrm{feature}_i - \mu_{\mathrm{feature}})^2/L\right] * \left[\sum_{i=1}^{L}(\mathrm{Temporal\_Dist}_i - \mu_{\mathrm{Temporal\_Dist}})^2/L\right]}}.$$

$$\rho_s = \frac{E\left[(\mathrm{feature}_i - \mu_{\mathrm{feature}})\,(\mathrm{Spatial\_Dist} - \mu_{\mathrm{Spatial\_Dist}})\right]}{\sqrt{\mathrm{var}(\mathrm{feature})\mathrm{var}(\mathrm{Spatial\_Dist})}} = \frac{\sum_{i=1}^{L}\left[(\mathrm{feature}_i - \mu_{\mathrm{feature}}) * (\mathrm{Spatial\_Dist}_i - \mu_{\mathrm{Spatial\_Dist}})\right]/L}{\sqrt{\left[\sum_{i=1}^{L}(\mathrm{feature}_i - \mu_{\mathrm{feature}})^2/L\right] * \left[\sum_{i=1}^{L}(\mathrm{Spatial\_Dist}_i - \mu_{\mathrm{Spatial\_Dist}})^2/L\right]}}$$

TABLE III
PERFORMANCE OF THE CLASSIFIER

Mode Selection	Success Rate
1'st	63.48%
2'nd	75.8 %
3'rd	85.65%
4'th	92.61%
5'th	97.25%
6'th	100 %

(a)

Allowable Add. Dist.	Success Rate
0 %	63.48%
< 5 %	80.14%
< 10%	87.83%
< 20%	93.33%

(b)

TABLE IV
STREAMING "HALL MONITOR" AT 100 KB/S

CU	1	2	3	4	5	6	7	8	9	10
Optimal Mode	3	6	6	6	6	6	6	6	1	6
Selected Mode	1	6	6	6	6	6	6	6	6	6
AC (%)	2	0	0	0	0	0	0	0	5	0

chance of choosing the correct mode (i.e., the allowable additional distortion within 10%) will increase to 88%.

To illustrate the impact of mode selection failure, we define the additional cost (AC) as shown at the bottom of the page.

The classifier's performance for six MPEG-4 video sequences (e.g., Hall Monitor, Salesman, Stefan, Suzie, Mom, and Akiyo) using six coding modes (e.g., FGS, FGST9, FGST13, FGS-SE, FGST9-BC, and FGST13-BC) based on the perceptual distortion metrics are shown in Tables IV–IX. We compare the additional code of the selected mode for each CU. From the experiments, we have demonstrated that our method is efficient and the AC of misclassified modes will be less than 10% for most of the cases, as shown in Tables IV and V. Furthermore, we observe the cases of which the AC is larger than 10% and conclude that there are two reasons: 1) insufficient feature vectors in the training set and 2) video sequences with low motion activity and low texture complexity.

The probability of the feature vector belonging to certain hypothesis is related to the number of appearance of such feature vector in the training set. Thus, the selected mode for the CUs, of which the number of corresponding feature vectors in the training set is not sufficient, may not be the best one. For example, the cases of the tenth CU of Stefan and the second CU of Suzie are rare in our training set. The performance of these two video sequences is shown in Tables VI and VII. This problem can be improved by collecting a larger training set.

Tables VIII and IX are examples of the cases of slow motion and low texture complexity video sequences. From these figures, we can find that the cost difference between two different coding modes is quite small because the perceived quality of Mom and Akiyo with low texture complexity and low motion activity is insensitive to the distortion. Thus, there is no guarantee that the best coding mode will be selected for this kind of video sequences. The cost difference between two different coding modes is too little for our algorithm to select an effective coding mode. However, we may improve the performance by: 1) choosing better features to characterize the video sequences; 2) using more complex Gaussians to model the distribution of feature vectors; or 3) making decision by user preference for this kind of video sequences.

Our classifier has been trained at three different transmission bandwidths, 100, 200, and 300 kb/s, however, we then model the originally discrete pdf of the feature vectors by mixture of Gaussians to obtain the continuous pdf. Therefore, we may test our algorithm at the other transmission bandwidths, e.g., $50 \sim 300$ kb/s, and show the outcomes of the classifier. The comparisons between the optimal modes and the selected modes

TABLE V
STREAMING "SALESMAN" AT 200 KB/S

CU	1	2	3	4	5	6	7	8	9	10
Optimal Mode	1	2	1	2	1	1	2	2	1	1
Selected Mode	1	1	1	1	1	1	1	1	1	1
AC(%)	0	2	0	0	0	0	1	5	0	0

TABLE VI
STREAMING "STEFAN" AT 300 KB/S

CU	1	2	3	4	5	6	7	8	9	10
Optimal Mode	3	6	3	3	3	6	3	3	3	3
Selected Mode	3	3	3	3	3	3	3	3	3	4
AC(%)	0	2	0	0	0	5	0	0	0	36

TABLE VII
STREAMING "SUZIE" AT 100 KB/S

CU	1	2	3	4	5
Optimal Mode	6	1	6	6	6
Selected Mode	3	3	3	3	6
AC(%)	7	29	4	4	0

TABLE VIII
STREAMING "MOM" AT 300 KB/S

CU	1	2	3	4	5	6	7	8	9	10
Optimal Mode	3	3	2	4	1	2	1	1	3	1
Selected Mode	3	3	3	3	3	3	3	3	3	3
AC(%)	0	0	1	5	10	10	10	16	0	26

TABLE IX
STREAMING "AKIYO" AT 300 KB/S

CU	1	2	3	4	5	6	7	8	9	10
Optimal Mode	1	1	3	1	1	1	1	3	1	1
Selected Mode	3	3	3	3	3	3	3	3	3	3
AC(%)	12	32	0	31	26	44	24	0	31	20

$$\mathrm{AC} = \left| \frac{\text{Dist. of the best mode-Dist. of the chosen mode}}{\text{Dist. of the best mode}} \right| \times 100\%$$

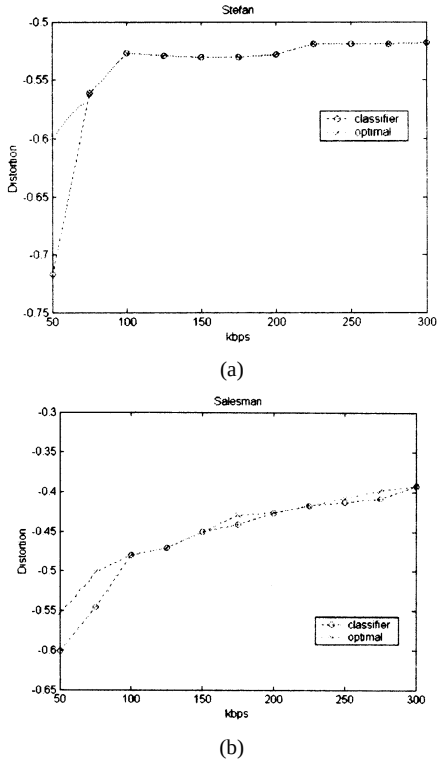


Fig. 7. Optimal mode versus selected mode at different transmission rates.

at different transmission bit rates for the fifth CU of Stefan and the fourth CU of Salesman are shown in Fig. 7. In Fig. 7(a), our classifier provides the optimal selection mode, since the selected mode and the optimal mode induce the same distortion. However, in Fig. 7(b), the difference between the optimal mode and the selected mode is also very small.

VI. Conclusion

This paper has proposed a mode determination scheme to decide which FGS coding mode in MPEG-4 is the most suitable one for the input CU. The mode determination scheme converts the problem of minimization of the total cost into a standard maximum likelihood problem. The experimental results illustrate that the mode determination scheme provides an efficient strategy to encode the video sequences under a given bit rate range. Since the priori probabilities and the conditional probability density functions are available after the off-line training process, the mode determination can be an on-line operation.

References

[1] "Coding of moving pictures and audio," ISO/IEC, JTC 1/SC 29/WG 11 N4350, 2001.

[2] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 301–317, Mar. 2001.

[3] H. M. Radha, M. van der Schaar, and Y. Chen, "The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP," *IEEE Trans. Multimedia*, vol. 3, pp. 53–68, Mar. 2001.

[4] M. van der Schaar and H. Radha, "A hybrid temporal-SNR fine-granular scalability for internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 318–331, Mar. 2001.

[5] M. van der Schaar and Y.-T. Lin, "Content-based selective enhancement for streaming video," in *Proc. IEEE ICIP*, vol. 2, 2001, pp. 977–980.

[6] H. Katata, N. Ito, and H. Kusao, "Temporal-scalable coding based on image content," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 52–59, Feb. 1997.

[7] S. Wolf and M. H. Pinson, "Spatial-temporal distortion metrics of in-service quality monitoring of any digital video system," presented at the SPIE Int. Symp. Voice, Video, and Data Communications, Boston, MA, Sept. 11–22, 1999.

[8] D. S. Turaga and T. Chen, "Classification based mode decisions for video over networks," *IEEE Trans. Multimedia*, vol. 3, pp. 41–52, Mar. 2002.

[9] N. Vlassis and A. Likas, "A Greedy EM algorithm for Gaussian mixture learning," IAS , Tech. Rep., 2000.

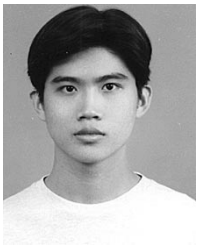
[10] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Mag.*, vol. 13, pp. 47–60, Nov. 1996.

[11] W. Li, "Bit-plane coding of DCT coefficients for fine granularity scalability," ISE/IEC, JTC1/SC29/WG11, MPEG98/M3989, 1998.

[12] B. Schuster, "Fine granular scalability with wavelets coding," ISE/IEC, JTC1/SC29/WG11, MPEG98/M4021, 1998.

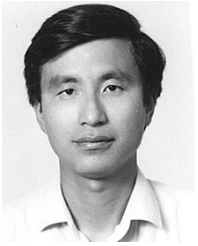
[13] S. Cheung and A. Zakhor, "Matching pursuit coding for fine granular scalability," ISE/IEC, JTC1/SC29/WG11, MPEG98/M3991, 1998.

[14] H. M. Radha, Y. Chen, K. Parthasarathy, and R. Cohen, "Scalable Internet video using MPEG-4," *Signal Processing: Image Commun.*, pp. 95–126, 1999.



Bin-Feng Hung received the B.S. and M.S. degrees in electrical engineering from the National Tsing-Hua University, HsinChu, Taiwan, R.O.C., in 2000 and 2002 respectively. Currently he is working for Etron Technology, Inc., HsinChu, Taiwan, R.O.C.

His research interests are image processing and multimedia communication.



Chung-Lin Huang (S'84–M'86) received the B.S. degree in nuclear engineering from the National Tsing-Hua University, HsinChu, Taiwan, R.O.C., in 1977, the M.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1979, and the Ph.D. degree in electrical engineering from the University of Florida, Gainesville, FL, in 1987.

From 1987 to 1988, he worked for the Unisys Company, Orange County, CA, as a Project Engineer. Since August 1988, he has been with the Electrical Engineering Department, National Tsing-Hua University. Currently, he is a Professor in the same Department. His research interests are signal processing, image/video processing, and multimedia communication.