Dynamic Resource Allocation Via Video Content and Short-term Traffic Statistics

Min Wu, *Student Member, IEEE,* Robert A. Joyce, *Student Member, IEEE,* Hau-San Wong, Ling Guan, *Senior Member, IEEE,* and S.-Y. Kung, *Fellow, IEEE*

Abstract— **The reliable and efficient transmission of high quality VBR video through the Internet generally requires network resources be allocated in a dynamic fashion. This includes the determination of when to re-negotiate for network resources as well as how much to request at a given time. The accuracy of any resource request method depends critically on its prediction of future traffic patterns. Such prediction can be performed using the content and traffic information of short video segments. This paper presents a systematic approach to select the best features for prediction, indicating that while content is important in predicting the bandwidth of a video bitstream, the use of both content and available short-term bandwidth statistics can yield significant improvements. A new framework for traffic prediction is proposed in this paper; experimental results show a smaller mean-square resource prediction error and higher overall link utilization.**

Keywords— **Bandwidth prediction, dynamic resource allocation, multimedia over IP, neural network (NN), VBR video.**

I. Introduction

TRANSMISSION of digital video and other multimedia information over the Internet is becoming increasingly important. Applications such as video conferencing and multimedia streaming have the potential to transform both learning and entertainment worldwide. The high bandwidth of digital video, one of the key components of most multimedia sources, requires careful management of network resources in order to keep utilization high while preserving any quality of service (QoS) requirements. Variable bit rate (VBR) video, which in general offers improved perceptual quality for a given average bitrate, presents a particular challenge in time-varying resource allocation.

Bandwidth allocation and management for individual streams generally must be done at the "edges" of the network, in order to conserve computational resources on network switches, as illustrated in Figure 1. Such systems will likely not have complete knowledge of the network state, and must therefore make their use of network resources as minimal as possible to maintain a given QoS. If a source requests more bandwidth than it actually uses, the overall network utilization drops. Conversely, if the source exceeds its bandwidth request, packet loss and delay will become significant. While offline systems could compute the exact dynamic bandwidth requirements for a stream before transmitting it, on-line processing is desirable in many applications. Systems such as video conferencing and live news-on-demand absolutely require on-line processing. In addition, on-line processing is needed in any system that dynamically transcodes video, or that splices and combines segments in an interactive manner. To keep delay and computational requirements low, the information used to make bandwidth decisions should be directly available from the compressed video stream. The overall goal is to have a resource management system that can accurately estimate the required bandwidth in real-time.

No one-time bitrate allocation will provide loss-free VBR video transmission with high utilization and low delay. For MPEG-1 and MPEG-2 streams, the bitrate variations can be up to an order of magnitude and occur on two different time scales. The shorter time scale corresponds to the duration of the GoP (group of pictures); the variation is due to the fact that intracoded (I) frames generally require more bits than forward predicted (P) frames, which in turn require more bits than bidirectionally predicted (B) frames. The brief spikes in traffic caused by I frames are generally not a problem for networks; as most MPEG compressors produce only about two or three I frames per second, a small buffer can adequately smooth the traffic if some delay is tolerable. The long-term variation is brought about by the changes in semantic content of different shots and scenes. Such bandwidth changes cannot be easily absorbed by reasonable-capacity network buffers. This long-term bitrate variation is one of the biggest challenges in VBR video transmission.

Traditional IP traffic, such as that generated by file transfers and email communication, is supported by best-effort service which does not have guaranteed delay, transfer rate, or other QoS characteristics. To facilitate transmission of real-time multimedia content across the Internet for future integrated services, several new protocols have been proposed in recent years. Among them, the Resource Reservation Protocol (RSVP) is a network-control protocol that allows Internet applications to obtain a certain QoS for the corresponding data flow [1][2]. Within this protocol, a

Manuscript received May 31, 2000; revised January 31, 2001. This work was supported in part by a New Jersey State R&D Excellence Award, by the Australian Research Council, and by Mitsubishi Electric Research Labs, Murray Hill, NJ. The associate editor coordinating the review of this paper and approving it for publication was Dr. John Aa. Sorensen.

M. Wu, R. A. Joyce, and S.-Y. Kung are with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544, USA (e-mail: minwu@ee.princeton.edu; robjoyce@ee.princeton.edu; kung@ee.princeton.edu).

H.-S. Wong was with the School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia. He is now with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong.

L. Guan was with the School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia. He is now with the Department of Electrical Engineering, Ryerson Polytechnic University, Toronto, ON, Canada.

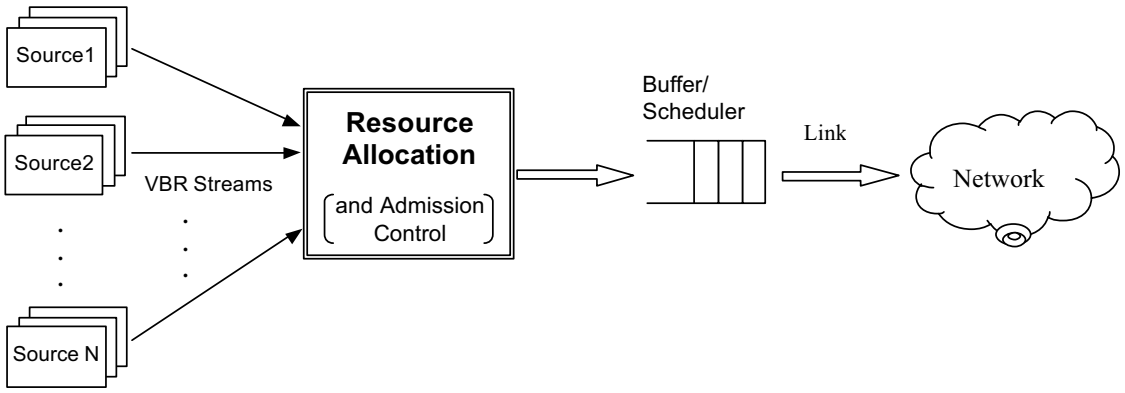


Fig. 1. Illustration of dynamic resource allocation for multiplexing VBR streams.

route reservation is created and periodically updated in two stages: the sender multicasts PATH messages containing traffic characteristics, then RESV messages containing resource reservation requests are forwarded from the receiver along a reverse path. In the context of QoS-guaranteed network communication, it is crucial to quantify the video traffic characteristics as precisely as possible (no matter what protocol is being used). Such quantification generally involves prediction of future and/or long-term traffic patterns because the frequency of reservation adjustment is limited in practice.

This paper proposes a new resource allocation framework, making use of content as well as short-term traffic statistics and achieving better prediction accuracy than content- and traffic-only approaches. Its overall structure is shown in Figure 2, which also serves as a road map of the paper. In Section II, we shall discuss two major issues of dynamic resource renegotiation and several candidate features that can be used for predicting video traffic. Approaches for the selection of relevant features are presented in Section III. Based on the feature selection, we propose a new neural network traffic predictor in Section IV, and demonstrate its effectiveness in improving network utilization. Conclusions and possible research directions are described in Section V.

II. Dynamic Resource Renegotiation for VBR Video

As mentioned above, the hallmark of VBR video is that its bandwidth undergoes both short- and long-term changes, in reaction to the complexity—and therefore, compressibility—of the underlying video. Allocating a constant amount of bandwidth for a VBR stream will lead to one of the following

1) inefficient use of network resources, due to over-allocated bandwidth, OR

2) a requirement of large endpoint (and possibly internetwork) buffers, causing variable delay proportional to the complexity of the most recent frames.

In order to obtain high network utilization and low delay, the bandwidth requests made by the VBR source must be periodically renegotiated. Conventional approaches renegotiate resources according to changes in bitstream level statistics [3]. In these approaches, it is common to use a parametric model to predict future traffic, as described in work such as [4][5], and references therein. Content-based approaches have been introduced, motivated by the high correlation between long-term traffic characteristics and video content [6][7]. Our study shows that while content is a major factor controlling the bandwidth, content alone may not be sufficient in predicting future traffic and in determining resource requests.

In studying the dynamic nature of resource requirements in VBR video, we shall look at the two issues illustrated in Figure 3:

1) at what points the bandwidth resource should be renegotiated, and

2) how much bandwidth resource to ask for at any given point.

A. Bandwidth Renegotiation Points

Whenever the traffic characteristics of the transmitted VBR stream change dramatically, the requested bandwidth should be renegotiated. A tradeoff in overhead must be considered, however: if the renegotiation is too often (say, every frame), the request and negotiation packets themselves will be a significant source of traffic. In addition, the renegotiation process likely involves delay itself, and is limited by the available computational power. Renegotiating too infrequently leads to dropped packets or frames, poorer overall network utilization, and possibly wasted expense, if bandwidth is not a free commodity.

The on-line determination of bandwidth renegotiation points in VBR video generally falls into one of three categories: deterministic, traffic-based, and content-based. Deterministically setting the renegotiation points is the simplest method: bandwidth requests are made every n frames, where n is an empirically determined balance between request overhead and correlation of frame bitrates. Traffic-based renegotiation, mentioned above, occurs when the stream violates a previously negotiated bandwidth request, or when utilization drops below some level. Although traffic-based renegotiation tracks the real bandwidth more closely, a single complex frame can cause the re-

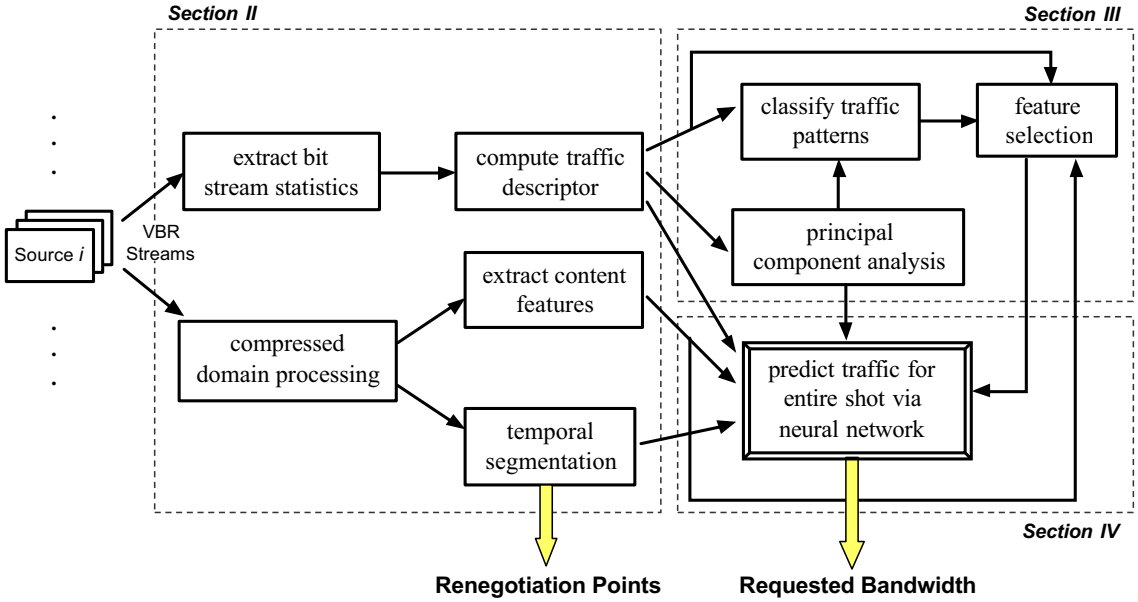


Fig. 2. Proposed resource allocation structure, containing two subsystems: determination of renegotiation points (Sect. II) and bandwidth requests (Sects. III & IV). Sect. II describes the feature extraction and temporal segmentation, Sect. III details the feature selection and evaluation methods used in training, and Sect. IV describes our NN-based per-interval traffic predictor.

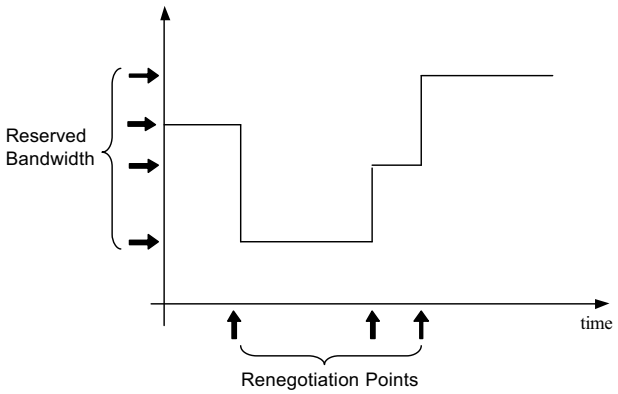


Fig. 3. An illustration of the two issues in dynamic resource allocation: choice of renegotiation points and how much bandwidth to request at each point, in order to track the source's requirements as closely as possible.

quested bandwidth to remain elevated for some time, even if successive frames require few bits. A more "natural" set of renegotiation points is the set of shot boundaries in the video stream. By studying the bits used per frame in VBR video, one sees that the most dramatic changes occur at the beginning of new camera shots [7]. Within a single shot, the traffic characteristics are relatively constant[1].

There exist many approaches to finding shot boundaries in the compressed domain [8]. For simplicity, we consider only abrupt transitions in this paper and adopt the compressed-domain cut detector described in [9]. This method uses a windowed relative threshold on the sum of absolute pixel differences,

$$d_k = \sum_{i,j} |x_k(i,j) - x_{k-1}(i,j)| \tag{1}$$

where the $x_k(i,j)$ is the first-order estimate of the DC pixel (i,j) in frame k. As discussed in [9], DC frame estimations can be easily computed from the compressed data. This cut detection scheme allows for fast, on-line computation of renegotiation points.

B. Bandwidth Requests Per Interval

After selecting renegotiation points, the next step is to determine how much resource to request for each interval without introducing significant delay. For natural renegotiation points such as shot boundaries, past shots' traffic generally cannot help in determining how much resource to request as the traffic pattern has changed. Exact traffic information for the new video shot can be obtained by measuring every frame between the current and the next renegotiation point, and using this as a basis for the resource request. This is possible in an offline situation where the future video is available, but is not suitable for real-time applications because significant delay will be introduced (particularly if the shots are long). With the requirement of online processing in mind, one can predict the traffic for the entire shot based on an observation of the first few frames. This is illustrated in Figure 4, where the shaded areas indicate observation periods. Renegotiation is performed after the short-term observation, and if granted, the video will be transmitted using the newly reserved bandwidth. (If the request is not granted, the source could attempt to transcode the video in order to fit into a smaller bandwidth for the single shot.) Note that the observation will inevitably introduce some delay in renegotiation,

[1]If a shot has a sudden change in content features, the change can be considered a boundary as far as renegotiation is concerned. For simplicity, we will ignore such intrashot "boundaries".

but the video itself may be transmitted without delay, as in Fig. 4(a). With this approach, unexpected bursty traffic during the shaded periods can only be accommodated by adding extra capacity to network buffers. For short-delay tolerable applications, the video may be transmitted with a t-second delay as in Fig. 4(b), so that the video traffic is always within the bounds of the negotiated agreement. While our approach can be applied to both delayed and non-delayed transmission, the performance of delayed transmission is better. In this paper, we shall focus on the delayed transmission case.

A content-based approach to per-interval prediction has been proposed by Bocheck *et al.*, consisting of training and testing stages [7]. In the training stage, content features are quantized into a small number of levels (e.g., slow / medium / fast motion), and every possible combination of significant features is labeled as one content class for which the typical traffic pattern is computed. After training, the content class of each shot in the test video is identified by extracting the same features, and the typical traffic pattern of that class is taken as the predicted traffic for the shot. We notice some potential weaknesses of this approach. First, the specific prediction structure via classification can only feasibly incorporate a limited number of coarsely-quantized features. Also, prediction based only on content may not be applicable for video streams produced with different encoding algorithms or parameters. Finally, some useful and readily available information, such as the exact bandwidth statistics of the video in the observation period, are not incorporated.

In order to alleviate these weaknesses, we take a more general approach by using a neural network with non-quantized features to predict long-term traffic. Both short-term traffic and content are considered as candidate inputs to be used for prediction. Before we address how to evaluate the relevance of each candidate to traffic prediction, we shall discuss the traffic descriptors that we use to quantify both the long-term traffic to be predicted and the observed short-term traffic to be used as the candidate predictor inputs. We also describe fourteen compressed-domain content features, each of which has the potential to influence traffic and will be used as a candidate prediction input.

B.1 Video Traffic Descriptors

Many traffic descriptors have been proposed in literature. Among them, peak rate and average rate are two very simple ones, but they do not capture the traffic pattern over different time scales. To overcome this problem, Knightly *et al.* proposed the D-BIND descriptor for deterministic service, providing a performance guarantee for the worst case [10]. The D-BIND (*deterministic bounding interval dependent*) model is essentially a vector containing the maximum allowed arrival rate for intervals of various lengths. It is defined as follows: let $A[\tau, \tau+t]$ be the cumulative number of bits arriving during the t-length interval beginning at time τ. The tightest bound over all time,

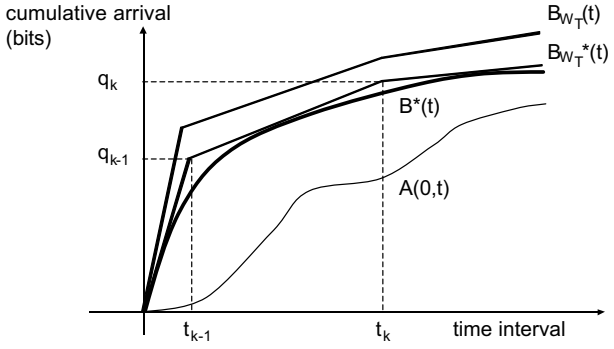


Fig. 5. Illustration of the D-BIND traffic descriptor: $A(0,t)$ is the cumulative arrival in the interval $[0,t]$, $B^*(t)$ is the empirical envelope of all $A[\tau,\tau+t]$. $B_{W_T}(t)$ is any piecewise-linear bounding function of $B^*(t)$, and $B^*_{W_T}(t)$ is the tightest such function.

called the *empirical envelope*, is

$$B^*(t) = \sup_{\tau} A[\tau, \tau+t] \tag{2}$$

A piecewise-linear bounding function B_{W_T} is constructed, where $W_T = \{(q_k, t_k)|k = 1, 2, ..., p\}$ is the vector of [bit arrival, interval] pairs. Given a set of t_k, the tightest bounding function is denoted $B^*_{W_T}$, as shown in Figure 5. The D-BIND descriptor is usually expressed in terms of arrival rates, i.e., $R_T = \{(r_k, t_k)|k = 1, 2, ..., p\}$, where $r_k = q_k/t_k$. This descriptor captures both the short-term burstiness and the long-term traffic characteristics of a video segment, while being relatively simple to implement in admission control and policing.

We use the D-BIND descriptor and deterministic service in our experiments, though the proposed framework is applicable to other descriptors and policies. Fixing $[t_1, ..., t_p]$, the D-BIND descriptor is simply a vector $[r_1, ..., r_p]$. When a video segment is short, only the first several D-BIND elements can be reliably computed. Therefore, we choose r_1 through r_4 of the short observed traffic as candidate inputs for traffic prediction. When describing the entire shot, the dimensionality of D-BIND is large and the prediction complexity goes up. Such an increase is rather wasteful as there is redundancy in the D-BIND vector. For example, r_k is close to the average bitrate for all large k. In order to remove the redundancy and to reduce prediction complexity, we apply principal component analysis (PCA) [11] on D-BIND and use the first N principal components as the desired predictor output. More specifically, we estimate the covariance matrix $\hat{\Sigma}_r$ of the D-BIND vectors $[r_1, \ldots, r_p]$ from training video shots. The eigenvalues $\{\lambda_i\}$ and the corresponding eigenvectors $\{\underline{\phi_i}\}$ of the estimated covariance matrix are computed and sorted, i.e., $\lambda_1 \geq \lambda_2 \geq \ldots$, and $\hat{\Sigma}_r \underline{\phi_i} = \lambda_i \underline{\phi_i}$ for $i = 1, \ldots, p$. The j^{th} principal component of a given D-BIND vector $[r_1, \ldots, r_p]$ is the projection of the D-BIND vector onto the j^{th} eigenvector:

$$a_j = [r_1, \ldots, r_p]\underline{\phi_j}. \tag{3}$$

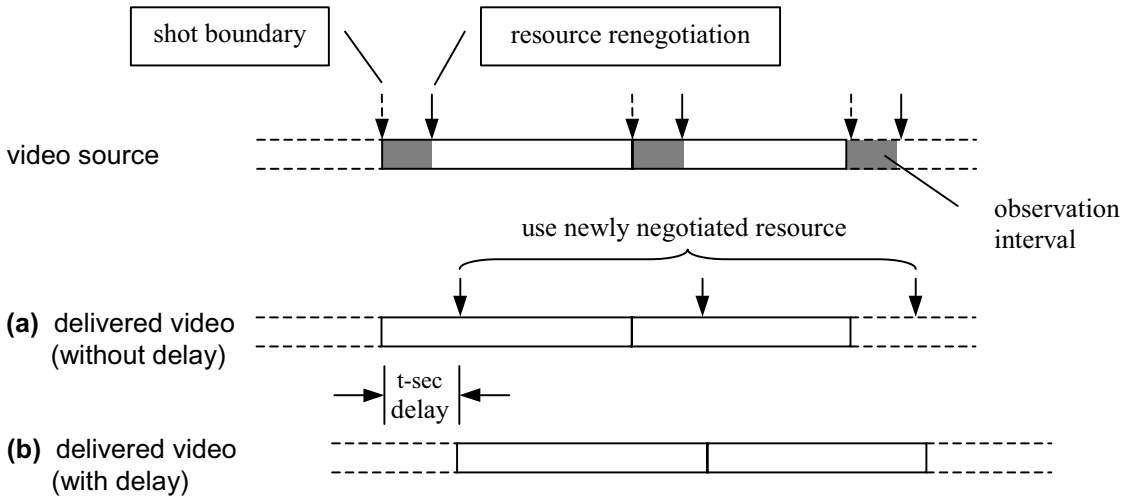


Fig. 4. Two methods of handling the delay caused by short-term content and traffic observations: (a) delivery without delay but before renegotiation takes place, and (b) delivery with t-second delay. The t-second delay includes the time spent on observation (shaded interval) and on renegotiation. Under scheme (a), large network buffers may be needed to smooth any unexpected traffic bursts between the interval boundary and successful bandwidth renegotiation t-seconds later. For short-delay tolerable applications, scheme (b) achieves better performance because the video traffic is always within the bounds of the negotiated agreement.

B.2 Content Features

Image complexity and motion have been suggested by Bocheck *et al.* as significant features related to video traffic [7]. Keeping in mind the requirement of efficient online processing, we extract fourteen features related to complexity and motion by processing the video in the compressed domain. This set of content features is likely more than necessary, but we will rely on the selection methods in the next section to weed out redundant features. Other features could be incorporated as well, if they have a high relevance to traffic.

The spatial "complexity" of the intracoded (I) frames is intuitively the dominant factor determining a stream's resource requirements, because the number of bits required to encode the frame is directly dependent on the energy compaction provided by the DCT and the compaction is less dense in blocks with edges or complex textures. In order to estimate complexity, we compute the weighted sum of the magnitudes of AC coefficients in the frame (DC coefficients are differentially encoded, so high DC magnitudes do not exact much penalty in traffic). Any weighting pattern giving more weight to higher-frequency DCT coefficients could be used; we chose to weight coefficients according to the sum of their frequencies in each dimension (the L_1 "distance" from the DC coefficient).

Motion vector magnitudes can dramatically effect the resources required by predicted (P and B) frames; for simplicity we shall consider only the forward predicted frames. Higher magnitudes mean more intense motion, and consequently more correction will likely be needed in the residue frames after motion compensation. Motion direction, for the most part, is irrelevant to traffic. We compute the mean motion vector magnitude, for the whole frame, as follows:

$$\overline{\|\mathrm{mv}\|} = \frac{1}{M} \sum_{i,j} \|\mathbf{m}_k(i,j)\|_2 \tag{4}$$

where M is the number of macroblocks in the video frame and $\mathbf{m}_k(i,j)$ is frame k's forward motion vector for the macroblock (i,j). In order to identify segments with strong motion in part of the frame, but not the entire frame, we also compute the value of (4) for each of four spatial quadrants.

The coding efficiency of predicted frames can also be measured by counting the number of intracoded blocks in the frame; areas that could not be adequately predicted from previous frames must be encoded again, at some expense in bandwidth. The fraction of P frame macroblocks that must be intracoded, instead of intercoded, therefore is another candidate feature.

Motion compensation is less efficient if the object or frame motion is not "simple", meaning that more correction must be applied in the residue frames if different macroblocks' motion vectors point in radically different directions. We measure the motion complexity in a number of ways, and rely on the feature selection process to find the ones most important to traffic prediction. First, we form a simple directional histogram of the motion vectors, in which each intercoded macroblock's motion vector is classified into five bins: up, down, left, right, or "zero", according to the dominant axis of the vector. Complex motion corresponds to having roughly equal values in each bin, so we use the variance over these five bins as a candidate feature. An alternative way of measuring the coverage of the motion prediction over the new frame is to compute the

spatial variance of the motion vector magnitudes:

$$\mathrm{var}\left(\|\mathbf{m}_k\|\right) = \frac{1}{M}\sum_{i,j}\|\mathbf{m}_k(i,j)\|_2^2 - \left(\frac{1}{M}\sum_{i,j}\|\mathbf{m}_k(i,j)\|_2\right)^2 \quad (5)$$

In addition, the spatial variances of the x and y motion vector components, as well as their cross covariance, are calculated:

$$\mathrm{var}\left(m_{k_x}\right) = \frac{1}{M}\sum_{i,j} m_{k_x}^2(i,j) - \left(\frac{1}{M}\sum_{i,j} m_{k_x}(i,j)\right)^2 \quad (6)$$

$$\mathrm{var}\left(m_{k_y}\right) = \frac{1}{M}\sum_{i,j} m_{k_y}^2(i,j) - \left(\frac{1}{M}\sum_{i,j} m_{k_y}(i,j)\right)^2 \quad (7)$$

$$\begin{aligned}\mathrm{cov}\left(m_{k_x}, m_{k_y}\right) &= \frac{1}{M}\sum_{i,j} m_{k_x}(i,j)m_{k_y}(i,j) \\ &\quad -\frac{1}{M^2}\sum_{i,j} m_{k_x}(i,j)\sum_{i,j} m_{k_y}(i,j) \quad (8)\end{aligned}$$

Finally, as we are only able to observe the very beginning of each new camera shot, the ways in which motion might change throughout the shot are important to estimate. Even if the motion magnitude is small in the first few frames, it can be large later in the shot, requiring more bandwidth to represent. To make this effect more manageable, we measure the object and frame acceleration in two ways. First, motion vectors from adjacent predicted frames are subtracted to form acceleration vectors, of which we take the mean magnitude:

$$\overline{\|\mathrm{accel}\|} = \frac{1}{M}\sum_{i,j}\|\mathbf{m}_k(i,j) - \mathbf{m}_{k-1}(i,j)\| \quad (9)$$

A high value for this mean indicates that the motion in the video is not simple, and that the residue frames will become increasingly complex (thus requiring more bits). The second candidate acceleration feature places greater emphasis on changes in speed, rather than changes in direction:

$$\overline{\Delta\|\mathbf{m}\|} = \frac{1}{M}\sum_{i,j}\left(\|\mathbf{m}_k(i,j)\| - \|\mathbf{m}_{k-1}(i,j)\|\right) \quad (10)$$

The eighteen candidate predictor inputs (fourteen content plus four traffic) are summarized in Table I. None of the fourteen content features requires full decompression of the VBR stream to compute; in MPEG-1 and 2, the amount of computation required is quite low. There is, however, significant redundancy in these features, and not all may be highly relevant to traffic prediction. The importance of selecting the relevant subset from the original feature set is closely related to the "curse of dimensionality" problem in function approximation, where sample data points become increasingly sparse when the dimensionality of the function domain increases, such that the finite set of samples may not be adequate for characterizing the original mapping [12]. In addition, the computational requirement is usually greater for implementing a high-dimensional mapping. To alleviate these problems, we reduce the dimensionality of the input domain in next section by choosing a relevant subset of features from the original set.

III. Feature Selection for Traffic Prediction

There exist several popular feature selection algorithms, which can roughly be grouped into linear methods and nonlinear methods. Linear methods are normally mathematically tractable and efficient; for example, principal component analysis is one of the best choices in transform-domain feature selection in the linear or minimum mean squared error sense. However, the nonlinearity inherent in content's effects on traffic prompts us to consider methods which select features in a nonlinear fashion in order to achieve improved performance. In this section, we first propose the adoption of an efficient nonlinear one-pass selection procedure, the sequential forward selection (SFS) method [13], and a specialized neural network model, the general regression neural network (GRNN) [14][15], for the purpose of selecting the relevant features for traffic prediction. We then discuss some limitations of the GRNN as the number of selected features grows, and adopt a consistency-based selection as a complementary approach.

Formally, we denote the original set with N features as the index set $F = \{1, \ldots, N\}$ and let S be the number of predicted quantities. Our purpose is to approximate the original mapping $f : E_F \subset \mathbf{R}^N \longrightarrow \mathbf{R}^S$, where $\mathbf{x}_F = [x_1, \ldots, x_N]^T \in E_F$ are the vectors in the input domain, using the alternative mapping $g : E_{F'} \subset \mathbf{R}^M \longrightarrow \mathbf{R}^S$, where $F' = \{i_1, \ldots, i_M\} \subset F$ is the relevant feature subset with $M \leq N$, and $\mathbf{x}_{F'} = [x_{i_1}, \ldots, x_{i_M}]^T \in E_{F'}$ are the vectors in the associated function domain. To achieve this, the subset F' has to be chosen in such a way that $f(\mathbf{x}_F) \approx g(\mathbf{x}_{F'})$ for every $\mathbf{x}_F \in E_F$ and the associated $\mathbf{x}_{F'} \in E_{F'}$.

The simplest way to construct the subset is to select all the possible combinations of features from the original set F, reconstruct the mapping g for each of these combinations, and then evaluate the approximation accuracy using a set of sample points. However, this approach is usually not feasible due to the large number of possible feature combinations, which amounts to $\sum_{m=1}^{N}\binom{N}{m} = 2^N - 1$ for the N features in F. Previous attempts to perform efficient sampling of this large combination include the adoption of genetic algorithms [16] where a population of subsets F' are generated and evaluated on the basis of the proximity of their associated functions g to the original mapping f on a set of sample points. Those subsets resulting in a good approximation are retained in the population and allowed to proceed into the next generation, while unsatisfactory subsets are removed from the population. New subsets in the population are then generated by slightly perturbing the successful subsets in a random way. Although this approach does not require the evaluation of all the possible combinations, it may still take many generations before a

TABLE I
CANDIDATE CONTENT AND TRAFFIC FEATURES TO USE IN PER-INTERVAL TRAFFIC PREDICTION.

Feature	Description	Feature	Description
1	I frame complexity	10	Mean MV magnitude, lower-right
2	Mean MV magnitude	11	Var. of MV x components
3	Var. of MV directional histogram	12	Var. of MV y components
4	Fraction of intracoded MB's	13	Cov. of MV x and y comp.
5	Mean magnitude of accel vectors	14	Var. of MV magnitudes
6	Mean change in MV magnitudes	15	Short-term D-BIND r_1
7	Mean MV magnitude, upper-left	16	Short-term D-BIND r_2
8	Mean MV magnitude, upper-right	17	Short-term D-BIND r_3
9	Mean MV magnitude, lower-left	18	Short-term D-BIND r_4

truly relevant feature subset emerges from the population.

To further complicate this problem, the mapping g associated with each subset is usually not available in an analytical form, and numerical approximations in the form of iterative algorithms are required to reconstruct the mapping. For example, a common approach for function approximation is to adopt an artificial neural network model to represent the mapping, where the sample points from the original function are considered as training examples for the network. In particular, the multilayer perceptron with the associated back-propagation (BP) training algorithm is often used for such purpose [12]. Due to the large number of training iterations required to determine the network weights and the generally slow convergence rate, BP will further accentuate the difficulties in evaluating the relevancy of a potentially large number of candidate subsets.

In summary, to allow effective identification of the relevant feature subsets, a potential algorithm should satisfy the following criteria to represent

1) an efficient but possibly sub-optimal approach to sample from the space of possible feature subsets such that candidate subsets which characterize the original mapping with reasonably good accuracy can be quickly identified.
2) an efficient but possibly approximate approach for evaluating the relevancy of individual candidate subsets without requiring iterative identification of the underlying mapping.

Due to the difficulties of the previously described approaches in fulfilling these two conditions, we propose the adoption of alternative approaches which specifically address these two criteria. For the first criterion, we propose the adoption of the sequential forward selection method [13] for incrementally constructing the relevant subset starting from a single feature. For this approach, a candidate subset with reasonably good relevancy to the current problem can be constructed rapidly without requiring the observation of a large number of possible subsets. For the second criterion, we follow a two-step process: we first adopt the general regression neural network [14] for evaluating the relevancy of the set of candidate subsets generated by SFS. Unlike the alternative multilayer perceptron model which requires iterative BP training, the parameters of the GRNN model can be directly determined in a single pass of training, which allows rapid evaluation of the individual feature subsets in terms of their relevancies. However, when the number of selected features is large, the GRNN approximation error in the high-dimensional mapping becomes significant; this is evident in Figure 7, where the error increases after the sixth feature is added. Since the confidence in the SFS/GRNN approach to feature selection diminishes around and beyond the minimum MSE point, we adopt a complementary follow-up step, discussed in Section III-C. As we will see in Section IV-A, the traffic prediction error is small when using the features selected by this two-step process as inputs to the prediction mechanism.

A. Sequential Forward Selection (SFS)

The sequential forward selection procedure [13] allows construction of a suitable feature subset starting from a single feature. Specifically, given the original feature set F, the SFS algorithm generates a sequence of subsets F'_m, $m = 0, \ldots, N$ with associated cardinalities $|F'_m| = m$. In other words, the original subset is incrementally expanded to accommodate new features. To begin with, we require a measure to evaluate the relevancies of the candidate subsets F'_m. For this purpose, we are usually given a set of training data $(\mathbf{x}_{F,p}, \mathbf{y}_p)$, $p = 1, \ldots, P$ for the desired mapping, where $\mathbf{x}_{F,p} \in E_F \subset \mathbf{R}^N$ denotes each sample vector in the input domain incorporating the full set of features. To evaluate the relevancy of a particular feature subset F'_m, we construct a mapping $g_{F'_m}$ which minimizes a particular discrepancy measure $D_{F'_m} = \sum_{p=1}^{P} d(\mathbf{y}_p, g_{F'_m}(\mathbf{x}_{F'_m,p}))$ over the set of reduced dimension vectors $\mathbf{x}_{F'_m,p} \in E_{F'_m} \subset \mathbf{R}^m$ corresponding to each of the full dimension vectors $\mathbf{x}_{F,p}$. We can then regard the value $D_{F'_m}$ as a measure of the relevancy of F'_m. A common candidate for $D_{F'_m}$ is the mean square error measure

$$D_{F'_m} = \frac{1}{P} \sum_{p=1}^{P} \|\mathbf{y}_p - g(\mathbf{x}_{F'_m,p})\|^2 \tag{11}$$

Given the discrepancy measure $D_{F'_m}$ and with the empty set ϕ assigned as the initial subset F'_0, we generate the $(m+1)$-th relevant subset from the m-th subset by individually evaluating the suitability of each remaining feature

in the complement set $\overline{F'}_m = F - F'_m$. Suppose we denote $F'_m = \{i_1, \ldots, i_m\}$ and $\overline{F'}_m = \{i_{m+1}, \ldots, i_N\}$. For each remaining feature $i_j, j = m+1, \ldots, N$, we form the new subset $G'_{m+1,j}$ as follows:

$$G'_{m+1,j} = F'_m \cup \{i_j\}, j = m+1, \ldots, N \tag{12}$$

The $(m+1)$-th relevant subset is then chosen from these candidate subsets using the following criterion:

$$F'_{m+1} = G'_{m+1,j^*} \tag{13}$$

where

$$\text{where } j^* = \arg\min_j D_{G'_{m+1,j}}, j = m+1, \ldots, N \tag{14}$$

In this way, a nested sequence of feature subsets $F'_1 \subset \ldots \subset F'_N$ can be constructed, and the associated performance measure values $D_{F'_1}, \ldots, D_{F'_N}$ indicate the relevancy of the corresponding subsets. As a result, we can select the subset containing the minimum number of features and with its associated discrepancy measure lower than a prescribed threshold.

As mentioned previously, one of the problems with the feature selection process is the necessity to reconstruct $g_{F'_m}$, usually by an iterative function approximation process, for the evaluation of $D_{F'_m}$ in Eq (11). In the next section, we propose the adoption of the general regression neural network for representing $g_{F'_m}$. As a result, only a single iteration is required to reconstruct $g_{F'_m}$ for each subset F'_m, allowing rapid evaluation of their relevancies.

B. SFS Implemented with General Regression Neural Network (GRNN)

The general regression neural network (GRNN) [14][15] can be considered as a special example of a radial basis function (RBF) network [17], where the first layer units adopt the Gaussian kernel as the nonlinear transfer function while the second layer consists of linear summation units (Fig. 6). However, unlike conventional RBF network where the centers and widths of the Gaussian kernels are determined by iterative clustering procedures, the corresponding parameters in GRNN are represented as deterministic functions of the training data. In other words, no iterative training procedures are required to reconstruct a mapping g using GRNN, thus allowing rapid evaluation of the relevancy of different feature subsets for our current problem.

To carry out function approximation using GRNN, we are given a set of sample observations $(\mathbf{x}_p, \mathbf{y}_p), p = 1, \ldots, P$ from the original function. We then associate each sample point with a single Gaussian kernel in the first network layer, with the input vector $\mathbf{x}_p$ assigned as the center of the kernel. In other words, there are P RBF units in the first network layer. For an arbitrary input vector $\mathbf{x}$ to the network, the output of the p-th RBF unit is given by

$$\beta_p = \exp\left[-\frac{(\mathbf{x}-\mathbf{x}_p)^T(\mathbf{x}-\mathbf{x}_p)}{2\sigma^2}\right] \tag{15}$$

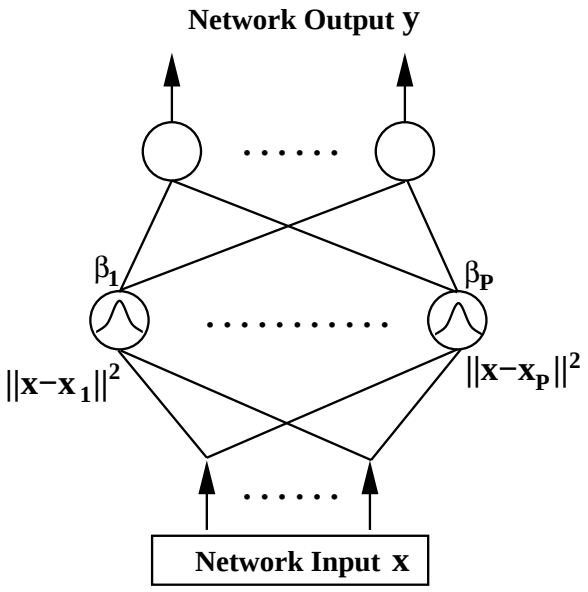


Fig. 6. The General Regression Neural Network (a special case of the RBF network) which is used in the implementation of the Sequential Forward Selection. The centers and widths of the Gaussian kernels are deterministic functions of the training data; iterative training is not needed.

where σ is a user-specified smoothing parameter. The GRNN output which represents the estimated function value for $\mathbf{x}$ is given by the following convex combination,

$$\mathbf{y} = \sum_{p=1}^{P} \alpha_p \mathbf{y}_p, \quad 0 \le \alpha_p \le 1, \quad \sum_{p=1}^{P} \alpha_p = 1 \tag{16}$$

where the coefficients α_p are defined as follows

$$\alpha_p = \frac{\beta_p}{\sum_{p=1}^{P} \beta_p}, \quad p = 1, \ldots, P \tag{17}$$

Intuitively, the GRNN performs interpolation by linearly combining the given training outputs $\mathbf{y}_p$ using a set of adaptively determined coefficients. If the current input vector $\mathbf{x}$ is close to one of the training inputs $\mathbf{x}_p$ in a Euclidean sense, the corresponding coefficient α_p given by Eq (16) will also become large, and the estimated output $\mathbf{y}$ will be close to the associated function value $\mathbf{y}_p$ for $\mathbf{x}_p$, which is a reasonable construction. On the other hand, those sample points which are far away from the current input vector do not appreciably contribute to the summation due to the exponentially decaying weighting function α_p.

B.1 Experimental Results Using SFS and GRNN

In this section, we apply the SFS technique to select feature subsets from the original set of four short-term D-BIND features and fourteen content features in Table I, and use the GRNN to evaluate the relevancies of each feature subset. Thus, corresponding to the original set F, we have an associated collection of training samples $\{\mathbf{x}, \mathbf{y}\}$ where the input vector $\mathbf{x}$ is of length 18 with each component being either a D-BIND feature or a content feature, and, for each shot, the output vector $\mathbf{y}$'s two compo-

nents are the principal components of the long-term D-BIND traffic descriptor. A total of P training samples $\{\mathbf{x}_p, \mathbf{y}_p\}, p = 1, ..., P$ are used for the feature selection process. Using SFS, we construct the nested sequence of subsets $F_1' \subset \ldots \subset F_{18}'$ and evaluate their relevancies via GRNN. In general, the evaluation of a particular GRNN model requires an additional test data set, since by construction the approximation error at each of the training samples is negligibly small. In view of this, we adopt the *leave-one-out* method, which is a special case of the cross validation approach [12], for evaluating the approximation error. For the cross validation approach, the training set with P samples is divided into an estimation subset with P' samples for determining the model, and a validation subset with $P - P'$ samples for validating the model. The leave-one-out method represents a special example of the above approach where $P' = P - 1$, such that the validation set consists of only one sample. This is necessary when the size of the training set is small to allow enough samples for model construction. This process is repeated by successively leaving out each of the P samples for validation and then averaging the associated error values.

Our experiments are performed on a 13175-frame video (about 7 minutes) digitized from cable television at 30 frames per second. The video consists of a fast-action documentary segment from "The Oprah Winfrey Show" and clips of the ABC series "The Practice." It is encoded via an MPEG-1 VBR coder with fixed quantization step size, and the average encoding rate is 2.1Mbps. Using the automatic shot boundary detection algorithm reviewed in Section II-A and feature extraction approaches discussed in Section II-B, 177 shots are identified and features are obtained. We plot the error values for each subset F_m' in Figure 7. The numbers in the horizontal axis indicate the total number of features selected after each SFS trial. The accompanying table shows which features were included in each subset F_m' (cf. Table I for feature definitions). For example, the fifth feature to be added is number 1, the I frame complexity; the subset F_5' consists of the features $\{1, 15, 16, 17, 18\}$. It is seen that the error curve exhibits a distinct minimum point at the feature subset F_6', which in our case corresponds to the index subset $\{1, 6, 15, 16, 17, 18\}$, beyond which the error starts to increase again. A possible interpretation of this minimum is that, due to the approximative nature of the GRNN, it will be increasingly difficult for the neural network to characterize the underlying mapping beyond a certain maximum number of features. With the limited number of training samples and their increasing sparseness in high-dimensional spaces, the error starts to rise beyond this point. In other words, although the sequence of indices in Fig. 7 is supposed to indicate the importance of the individual features in characterizing the long-term network traffic, this may not be the case for those features around and beyond the minimum point. As a result, it is natural to adopt F_6' as a first approximation to our relevant feature subset due to the smallness and comparative reliability of its associated error value. Among the features in F_6', it is observed that all four short-term D-BIND statistics are included in the subset, which implies that the short-term traffic statistics are essential for predicting the long-term traffic patterns. It is next observed that the complexity feature associated with the I-frames (feature 1) is more important for prediction than other content features.

We have already pointed out that the order of features beyond the minimum point may not necessarily reflect their actual importance ranking because of the increasing difficulties of characterizing mapping in high dimensional space using a finite training data set for the simple GRNN model, as indicated by the rising error values. The approximation nature of the GRNN model implies that the minimum itself is not exact; we therefore need to consider more carefully the relevance of those features around and beyond the minimum. More specifically, we use an alternative approach in next section to review the potential usefulness of features 1–14, i.e., the entire set of candidate content features.

C. Consistency-Based Feature Selection

In this section, we describe a consistency-based approach as a complementary selection mechanism to evaluate the relevance of content features with respect to video traffic. Consistency measures were originally used to formulate class separability and to select features which are most effective for preserving class separability [18]. For the problem of traffic prediction, this measure was used by Bocheck *et al.* to evaluate the relevancy of content features to video traffic [7]. The traffic classes generally indicate the average bitrate and typical bitrate pattern of video segments, for example, classes of low, medium, and high average rate, as well as of constant, semi-constant, and oscillating bitrate. The class information can provide insights on how much network resource should be allocated. However, as reviewed in Section II-B, there are some weaknesses and limitations in the particular way of determining traffic classes suggested by Bocheck *et al.*

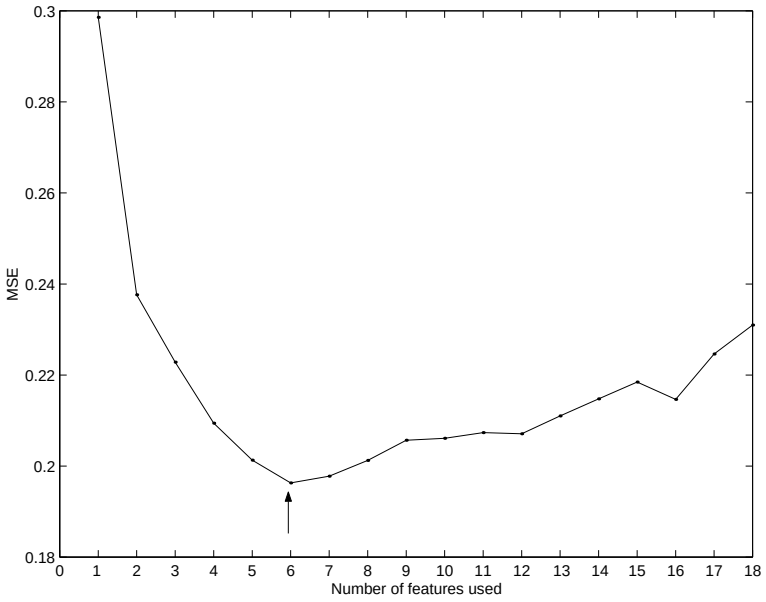
To overcome these weaknesses, we propose the following evaluation procedure, using a different way to determine traffic classes. In the first step, video shots are classified into k traffic clusters based on a specific traffic descriptor. Classification can be done by k-mean, E-M, or other algorithms. In the second step, a consistency measure $\mathcal{C}$ of each feature is computed [7]:

$$\mathcal{C} = \frac{\text{mean inter-class distance}}{\text{mean intra-class distance}} \tag{18}$$

where the distances are in the space of the features under consideration. A greater value of the consistency measure implies a better feature, because the feature has a small intra-class distance and large inter-class distance.

C.1 Experimental Results Using Consistency-Based Selection

We apply k-mean clustering to classify video shots' traffic into 4 clusters. Using the first two principal components of the D-BIND descriptor of each shot, the classification result is presented in Figure 8. Each cluster reflects a different level of complexity and action. For example, the



Feat. Set	Feat.	Feat. Set	Feat.
F'_1	15	F'_{10}	$8 \cup F'_9$
F'_2	$18 \cup F'_1$	F'_{11}	$12 \cup F'_{10}$
F'_3	$17 \cup F'_2$	F'_{12}	$13 \cup F'_{11}$
F'_4	$16 \cup F'_3$	F'_{13}	$4 \cup F'_{12}$
F'_5	$1 \cup F'_4$	F'_{14}	$2 \cup F'_{13}$
F'_6	$6 \cup F'_5$	F'_{15}	$10 \cup F'_{14}$
F'_7	$3 \cup F'_6$	F'_{16}	$9 \cup F'_{15}$
F'_8	$5 \cup F'_7$	F'_{17}	$11 \cup F'_{16}$
F'_9	$7 \cup F'_8$	F'_{18}	$14 \cup F'_{17}$

Fig. 7. Cumulative error plot for SFS/GRNN feature selection; the table shows which features are included after each SFS step. Minimum MSE is achieved after selecting six features, as indicated by an arrow. Our simulation study indicates that with feature selection order different from that listed in this figure, the dropping of MSE is less sharp. However, because of the increasing difficulties of characterizing mapping in high dimensional space, the feature order around and beyond the minimum may not reflect their actual importance, prompting the investigation of the alternative selection scheme for these features, discussed in Sect. III-C.

rightmost cluster involves fast motion along with considerable complexity. We then compute the consistency measure according to (18), with the results shown in Figure 9. We can see that I frame complexity (feature 1) has the highest consistency among all content features, which is the same result as achieved by SFS/GRNN approach in Section III-B.1. Similarly, we find that the average magnitude of P frame motion acceleration (feature 5) is with the second highest consistency. We also notice that features 7–10, the regional motion magnitudes, have high correlation with feature 2, the global motion magnitude, and both of them have similar consistency. To reduce the redundancy in the selected feature set and the prediction complexity, we prune the regional motion features and get four highly consistent features, namely, $\{1, 5, 2, 13\}$.

It should be pointed out that the consistency-based approach assumes features are uncorrelated and only considers features that are related with the traffic descriptor in a monotonic way as beneficial. For this class of features, a large distance in traffic space implies a large distance in feature values. Although these assumptions simplify the problem and provide a feasible way to evaluate a certain kind of relevancy of features, more complicated relations between features and traffic are not captured by this approach. How to feasibly and reliably capture more complex relations for feature selection will be studied in our future work.

IV. NN Traffic Predictor for Improvement of Network Utilization

In this section, we present a neural network (NN) traffic predictor utilizing the features selected in the last section. That is, our prediction takes into account both the significant content features and the bandwidth statistics of the video in the observation periods. In the context of dynamic resource allocation, the prediction results determine how much bandwidth to request. We shall first discuss the architecture of the proposed traffic predictor and present quantitative prediction results which demonstrate the performance of our framework as well as verify the contribution of various inputs suggested by our feature selection scheme. We then perform trace-driven simulation and show the enhancement of network link utilization when incorporating the proposed traffic predictor into the dynamic resource negotiation mechanism.

A. NN Predictor Architecture and Prediction MSE

Although the problem of predicting long-term or future traffic based on short-term traffic may be handled via parametric modeling, it is not easy to come up with a simple and effective parametric model when incorporating content features. For this reason, we use a neural network (NN) to accomplish the prediction task, as shown in Figure 10. The input to the neural network consists of the selected content features and traffic descriptors from the observation period. The outputs are the principal components of the D-BIND traffic descriptor for the entire shot, as discussed in Section II-B.1. We adopt a multilayer-perceptron network with a single hidden layer and apply the back-propagation (BP) approach to determine the weights and biases of the network in supervised training [11]. Recall, the overall system structure is summarized in Figure 2.

We shall demonstrate the performance of our proposed framework by evaluating the prediction mean squared er-

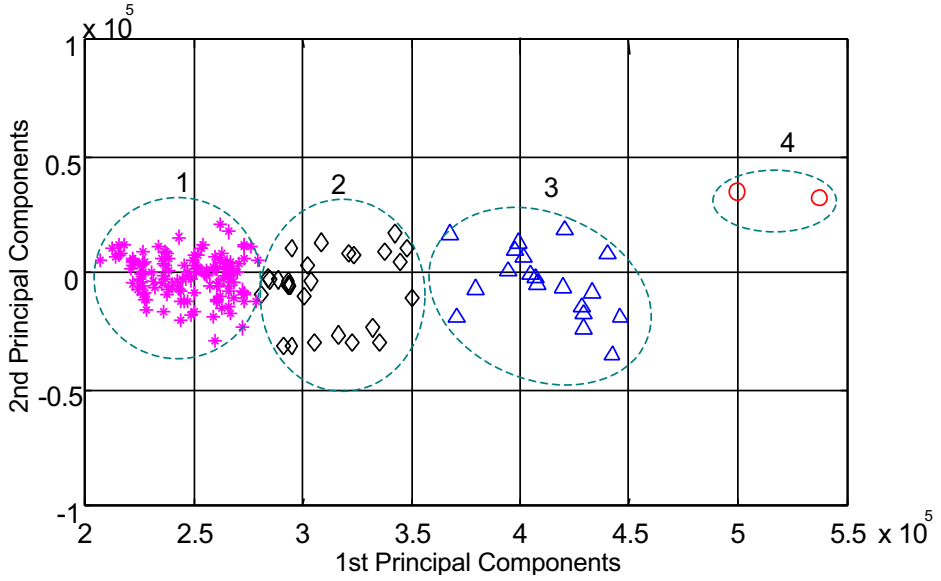


Fig. 8. Four traffic classes derived by K-mean clustering on the two principal components of D-BIND, the first step in consistency-based feature selection.

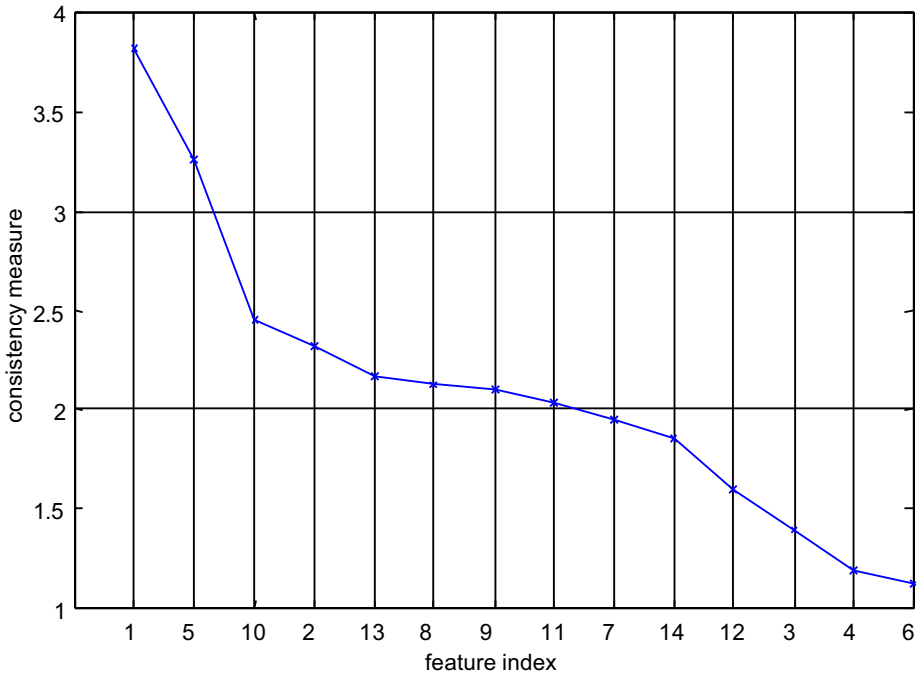


Fig. 9. Sorted consistency measures for each candidate content feature, when used individually.

ror (MSE), a commonly used criterion. For the traffic prediction problem, the overestimation of shot D-BIND descriptors could lower network utilization, and the underestimation could degrade QoS or even cause network buffer overflow. As mentioned before, our experiments are performed on a 13175-frame MPEG-1 VBR video consisting of segments from a fast-action documentary and a television drama.

To verify the selection results of the SFS/GRNN approach, the feature subset $F_6' = \{1, 6, 4\text{-dim D-BIND}\} = \{1, 6, 15, 16, 17, 18\}$ (Section III-B.1) is used for training a multilayer perceptron to predict the long-term traffic statistics. Among the 177 shots extracted from the video sequences, the first 50 shots are used as training samples for the network, and the next 127 shots are used as test data. We have listed the prediction mean square error in normalized units for different numbers of hidden nodes in Table II [2]. For the purpose of comparison, we have also included the prediction results by randomly choosing 2 sets

[2]Note that the D-BIND principal values are on the order of 10^5 bits per frame, and the prediction MSE of these principal values is on the order of 10^{10}.

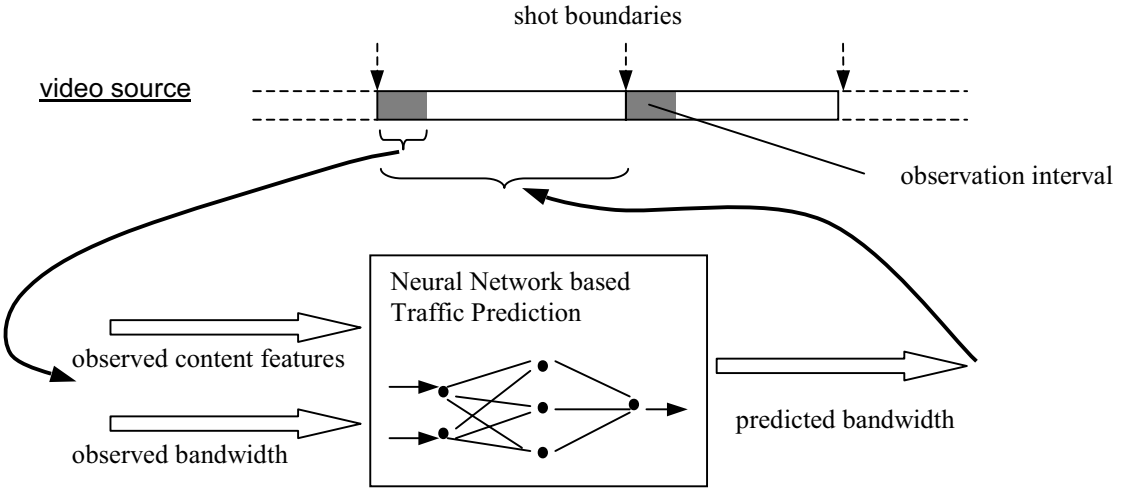


Fig. 10. Neural network based traffic prediction using both content and traffic features from the first few frames of a shot to predict the entire shot's traffic.

TABLE II

MSE TRAFFIC PREDICTION RESULTS USING CONTENT/TRAFFIC FEATURES SELECTED BY SFS/GRNN, TRAFFIC FEATURES ONLY, AND TWO RANDOM FEATURE SETS.

Feature Subset	number of hidden units	MSE (1st PCA)	MSE (1st and 2nd PCA)
F_6'	10	0.0238	0.0277
D-BIND	10	0.0247	0.0281
Random Set 1	10	0.0559	0.0695
Random Set 2	10	0.0426	0.0545
F_6'	20	0.0232	0.0268
D-BIND	20	0.0244	0.0279
Random Set 1	20	0.0579	0.0719
Random Set 2	20	0.0488	0.0617

TABLE III

MSE TRAFFIC PREDICTION RESULTS, COMPARING FEATURES SELECTED BY SFS/GRNN AND BY THE COMBINED SFS/GRNN/CONSISTENCY APPROACH.

Feature Subset	MSE (1st PCA)	MSE (1st and 2nd PCA)
F_6'	0.0232	0.0268
F_8'	0.0215	0.0257

of 6 features from the original 18. We can observe that the 6 features selected by SFS and GRNN achieve the smallest error in each case. In addition, we also notice that increasing the number of hidden nodes from 10 to 20 does not significantly improve the prediction results, and for some particular feature combinations the prediction error even increases for a large hidden layer, indicating the possibility of overfitting. As all the D-BIND features rank close to the top of the feature list, it is reasonable to suggest that most of the useful information for predicting the future traffic is already embedded in these short-term statistics. To confirm this, we have also included the prediction results using the 4 short-term D-BIND features only. We can observe that the resulting errors are only slightly greater than those of the original selected subset F_6', indicating that these short-term features are the most essential for predicting the long term network traffic.

From these results, we can conclude that the SFS/GRNN selection mechanism is capable of identifying the most important features—the short-term D-BIND statistics—for the current prediction problem. On the other hand, we can observe that the addition of content features to the D-BIND subset serves to improve the prediction result. That only two of the 14 content features are included in the selected subset is due to our previous decision not to adopt those content features beyond the GRNN minimum error point. As explained before, there are increasing difficulties in characterizing a high-dimensional mapping using a finite training data set for the simple GRNN model, and we have employed consistency-based selection to augment the SFS/GRNN process. To demonstrate the improvement, we list the prediction MSEs of the feature sets F_6' of the SFS/GRNN approach and $F_8 = \{1, 2, 5, 13, \text{4-dim D-BIND}\}$ of the combined approach (Section III-C.1) in Table III, where the number of hidden nodes is 20. The prediction MSE using F_8 selected by the combined approach is smaller than those selected by SFS/GRNN alone, especially for predicting the most significant component of D-BIND. This confirms that incorporating the alternative selection approach can enhance prediction performance.

Finally, using feature set F_8 selected by the combined approach, we compared the prediction MSE under four different strategies. With respect to renegotiation points, we consider: (A) using equal-length request intervals (one request every 75 frames, which is the average shot length), and (B) using shot boundaries from temporal segmentation. We also consider three different neural network inputs for traffic prediction: (I) the 4-dimension content feature (feature $\{1, 2, 5, 13\}$) of the observed video, (II) the 4-dimension D-BIND (feature $\{15, 16, 17, 18\}$) of the observed video, and (III) both of the above. Two sets of comparisons are shown in Fig. 11. Comparing the two left-

most columns, (A-III) and (B-III), we observe that (B-III) gives much smaller MSE, meaning that content-based renegotiation points are by far superior to non-content-based ones. Comparing the three rightmost columns, we observe that short-term traffic (B-II) gives better prediction than content features (B-I) alone. In addition, we found again that using both content and short-term bandwidth of observed video (B-III) is only marginally better than using short-term bandwidth alone (B-II). This implies that most of the useful information in content features for predicting traffic is already inherent in the short-term bandwidth statistics.

B. Improvement of Network Link Utilization

We shall compare with a static peak-rate allocation and a bitstream-level dynamic scheme to demonstrate the improvement of network link utilization achievable by our proposed approach. The R-VBR scheme, a heuristic dynamic renegotiation algorithm using D-BIND descriptors, was proposed in [3], claiming significant improvement over static peak rate allocation. It raises the reserved bandwidth (described by D-BIND) by a factor α when the real bandwidth exceeds the reserved resource, and lowers it by a factor β when the real bandwidth remains below the reserved resource for K frames. The average R-VBR renegotiation frequency is determined by the triplet (α, β, K). In contrast, our proposed scheme uses the shot boundaries, obtained from content-based temporal segmentation, as renegotiation points, and an NN traffic predictor to determine how much resource to ask for at each point. For the 177-shot video used in our experiments, the full D-BIND vector of each entire shot is estimated from the two principal components which are the outputs of NN traffic predictor. These predicted D-BIND descriptors are used for determining how much bandwidth to ask for in renegotiation.

Link utilization is obtained by trace-driven simulation, similar to that described in [7]. Multiple video sources, based on the above mentioned sample video but with random starting points, are multiplexed into a T3 line (link speed $c = 45$ Mbps). For simplicity, the current simulation blocks a source when its resource request is rejected, and a new request is generated at the next renegotiation point. More sophisticated admission control is certainly possible, a subject for future research. A network buffer with maximum capacity Q and FCFS queuing policy is used to smooth out the bursty traffic. When a renegotiation request is received from the n^{th} source, the worse case buffer occupancy is computed:

$$Q_1 = \max\left\{0, \max_{1\le k\le p}\left\{t_k \cdot \left[\sum_{i\neq n} a_i \cdot r_k(i) + r_k(n) - c\right]\right\}\right\}, \tag{19}$$

where i is the source index, k is the index of D-BIND components, p is the dimension of D-BIND descriptor, $r_k(i)$ is the k^{th} D-BIND components of the i^{th} source, and a_i is set to 1 if the i^{th} source is admitted and 0 otherwise. The requested resource is granted only if Q_1 is below Q. Given a bound of rejection probability (e.g., 1% in our simulation), link utilization is defined as:

$$u = \frac{\text{max number of admitted sources}}{\text{number of admittable CBR sources with rate } r_{avg}}, \tag{20}$$

where r_{avg} is the average rate of the entire video sequence. The simulation result of utilization versus buffer capacity is shown in Figure 12. With three parameter settings, $(\alpha = 1.3, \beta = 0.7, K = 30)$, $(\alpha = 1.3, \beta = 0.7, K = 60)$, and $(\alpha = 1.4, \beta = 0.7, K = 90)$, the R-VBR scheme generates requests at average rates of 0.81, 1.54, and 2.32 seconds, respectively. The corresponding utilization is shown as the dashed curves. The bottom straight line shows the utilization if the peak bandwidth were allocated to each sequence. The upper solid curve is the utilization of our proposed scheme, which renegotiates once every 2.48 seconds on average. The figure shows that our proposed scheme obtains much higher link utilization compared with peak-rate allocation scheme. Furthermore, our proposal outperforms the R-VBR scheme of similar renegotiation frequency by 18%, and by 9% against the R-VBR with tripled renegotiation frequency.

V. Conclusion and Future Research

We have proposed a new framework for resource allocation of VBR video, and presented systematic ways of evaluating features for video traffic prediction. According to our experiments, we found that

1) sequential feature selection with a general regression neural network is capable of identifying most relevant features, and incorporating alternative selection approaches further enhances prediction performance;
2) in determining optimal renegotiation points, a content-based approach is preferred over non-content-based ones;
3) in traffic prediction, using short-term bandwidth statistics as neural network inputs is more effective than using content; and
4) this approach to dynamic resource allocation significantly outperforms bitstream level approaches.

The proposed traffic prediction framework has the potential to be used with network control protocols such as RSVP to facilitate transmission of QoS-guaranteed multimedia transmission across the Internet and the quickly growing wireless data network. It can also be extended to communication over asynchronous transfer mode (ATM) networks, which make use of virtual circuit routing to reserve resources needed by each connection, achieving tight control over resource allocation and QoS [19].

In addition to the deterministic service and D-BIND traffic descriptor that are used in our experiments as proof-of-concept, our proposal is also applicable to other traffic descriptors and service policies. For example, probabilistic traffic descriptors and non-deterministic services may be pursued in future work as alternatives to deterministic methods. The prediction neural network may be trained beforehand, and be steadily updated online. Furthermore, our work can be applied to other problems related to video

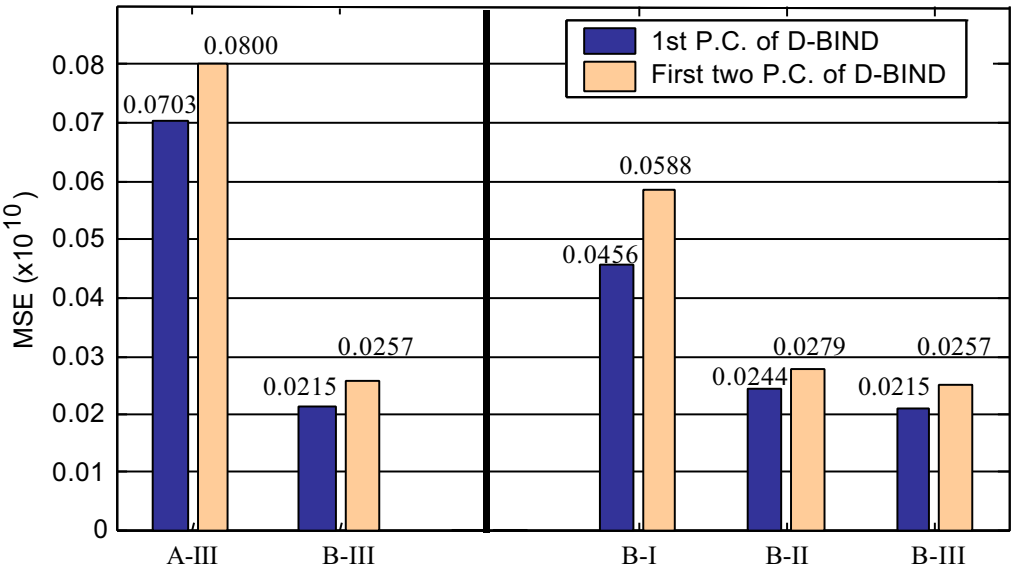


Fig. 11. Traffic prediction MSE with renegotiation points (A) at fixed intervals and (B) determined by shot-boundaries; (I) uses content features only, (II) uses short-term traffic only, while (III) uses both for prediction. For each of the six cases, the dark colored bar indicates the MSE for predicting the first principal component of the D-BIND for the entire shot, and the light colored bar indicates the sum of the MSEs for predicting the first two principal components.

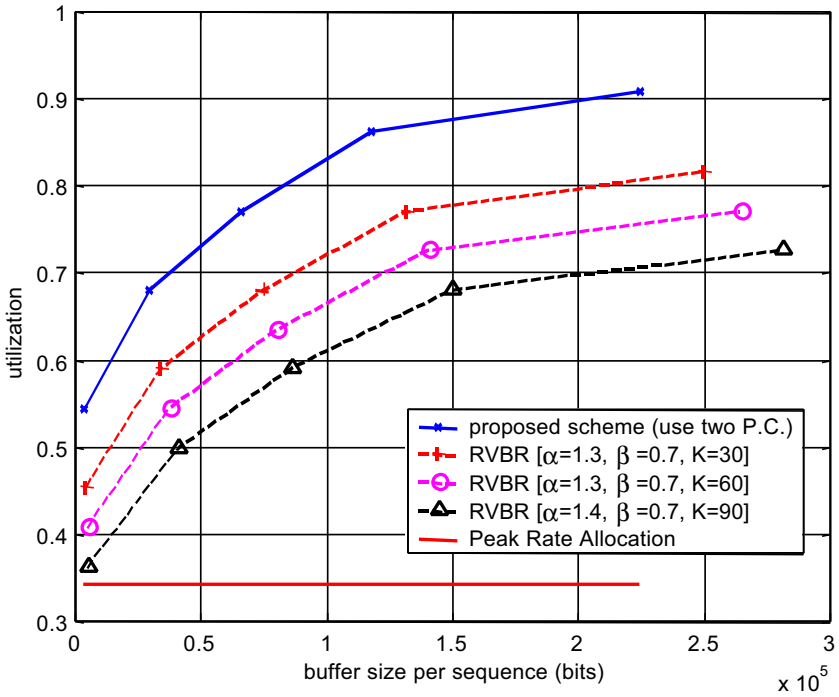


Fig. 12. Network utilization for multiplexed sources using the proposed scheme, as well as the renegotiated VBR and the peak-rate allocation.

traffic modeling, such as utility function estimation [20] and transcoding.

Acknowledgements The authors wish to thank Prof. Bede Liu of Princeton University for his enlightening discussions and support. The authors would also like to thank Mr. Xiang Yu of Princeton University for his help in digitizing VBR video streams, and Mr. Kim-Hui Yap of the University of Sydney for his help in performing initial experiments in feature selection.

References

[1] Resource Reservation Protocol (RSVP), White paper, Cisco Systems Inc., 1999, San Jose, CA. [Online]. Available: <http://www.cisco.com/cpress/cc/td/cpress/fund/ith2nd/it2443.htm>

[2] R. Jain, Recent advances in networking, Course web page, Ohio State Univ., Columbus. [Online]. Available: <http://www.cis.ohio-state.edu/ ∼jain/cis788-99/>

[3] H. Zhang and E. W. Knightly, "RED-VBR: A new approach to support delay-sensitive VBR video in packet-switched networks," in *Proc. NOSSDAV*, 1995, pp. 258–272.

[4] S. Chong, S. Li, and J. Ghosh, "Predictive dynamic bandwidth allocation for efficient transport of real-time VBR video over ATM," *IEEE Journal Sel. Areas of Comm.*, vol. 13, no. 1, pp. 12–23, 1995.

[5] M. R. Izquierdo and D. S. Reeves, "A survey of statistical source models for variable bit-rate compressed video," *Multimedia Systems*, vol. 7, no. 3, pp. 199–213, 1999.

[6] A. M. Dawood and M. Ghanbari, "MPEG video modelling based on scene description," in *Proc. IEEE Int'l Conf. on Image Processing*, 1998, vol. 2, pp. 351–355.

[7] P. Bocheck and S.-F. Chang, "Content-based VBR traffic modelling and its application to dynamic network resource allocation," Research Report 48c-98-20, Columbia University, 1998.

[8] J. S. Boreczky and L. A. Rowe, "Comparison of video shot boundary detection techniques," in *Storage and Retrieval for Still Image and Video Databases IV*. Proc. SPIE, 1996, vol. 2670, pp. 170–179.

[9] B.-L. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 5, no. 6, pp. 533–544, 1995.

[10] E. W. Knightly and H. Zhang, "D-BIND: An accurate traffic model for providing QoS guarantees to VBR traffic," *IEEE Trans. on Networking*, vol. 5, no. 2, pp. 219–231, 1997.

[11] S.-Y. Kung, *Digital Neural Networks*, Englewood Cliffs, NJ: Prentice-Hall, 1993.

[12] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Englewood Cliffs, NJ: Prentice-Hall, 1999.

[13] J. Kittler, "Feature set search algorithms," in *Pattern Recognition and Signal Processing*, C. H. Chen, Ed. Amsterdam: Sijthoff & Noordhoff, 1978.

[14] D. F. Specht, "A general regression neural network," *IEEE Trans. Neural Networks*, vol. 2, no. 6, pp. 568–576, 1991.

[15] H. Schioler and U. Hartmann, "Mapping neural network derived from the Parzen window estimator," *Neural Networks*, vol. 5, no. 6, pp. 903–909, 1992.

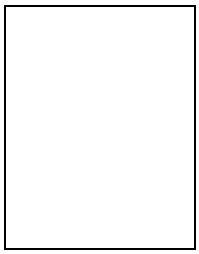
[16] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, MA: Addison–Wesley, 1989.

[17] J. Moody and C. J. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Computation*, vol. 1, no. 2, pp. 281–294, 1989.

[18] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, New York: Academic Press, 2nd ed., 1990.

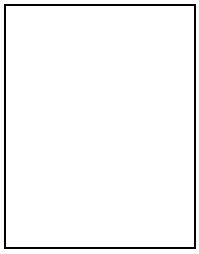
[19] J. Walrand, *Communication Networks: A First Course*, New York: McGraw–Hill, 2nd ed., 1998.

[20] S.-F. Chang and P. Bocheck, "Principles and applications of content-aware video communication," in *Proc. IEEE Int'l Symp. on Circuits and Systems*, 2000.



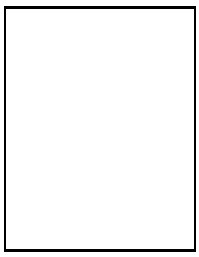
Min Wu (S'95) received the B.E. degree in electrical engineering and B.A. degree in economics from Tsinghua University, Beijing, China, in 1996 (both with the highest honors), and the M.A. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, in 1998 and 2001, respectively.

She was with NEC Research Institute and Signafy, Inc., Princeton, NJ, in 1998, and with the Media Security Group, Panasonic Information & Networking Laboratories, Princeton, NJ, in 1999. Her research interests include digital watermarking and information security, video communication, multimedia signal processing and content analysis.

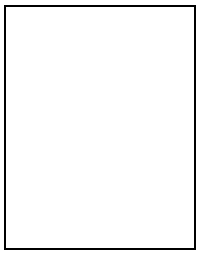


Robert A. Joyce (S'95) received the B.Sc. degree in electrical engineering (with honors) from Cornell University, Ithaca, NY, in 1997, and the M.A. degree from Princeton University, Princeton, NJ, in 1999. He is currently pursuing the Ph.D. degree in the Department of Electrical Engineering at Princeton University as well.

His research interests include image and video processing, video compression and transmission, visualization and design, and issues relating to multimedia databases and libraries.



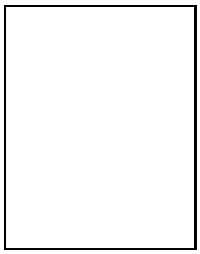
Hau-San Wong received the B.Sc. degree and the M.Phil. degree in Electronic Engineering from the Chinese University of Hong Kong, in 1991 and 1993, respectively. He was a research assistant in the Department of Mechanical and Automation Engineering of the same university in 1994. In 1999, Dr. Wong received his Ph.D. degree from the Department of Electrical Engineering at the University of Sydney, Australia. He is currently a postdoctoral fellow with the Department of Computer Science at Hong Kong Baptist University, Kowloon Tong, Hong Kong. His research interests include neural networks, computer vision and image restoration.



Ling Guan (S'88–M'90–SM'96) received his Ph.D. Degree in Electrical Engineering from University of British Columbia, Canada in 1989. In 1989-92, he was an Research Engineer of Array Systems Computing Inc, Toronto, Canada in machine vision and signal processing. From October 1992 to April 2001, he was a faculty member at University of Sydney, Australia. In May 2001, he joined the Ryerson Polytechnic University in Toronto, Canada, where he is currently a professor in Department of Electrical and Computer Engineering. In 1994, he was a visiting fellow at British Telecom. In 1999, he was a visiting professorial fellow at Tokyo Institute of Technology. In 2000, he was on sabbatical leave at Princeton University.

His research interests include multimedia processing and systems, optimal information search engines, signal processing for wireless multimedia communications, computational intelligence and machine learning, adaptive image and signal processing. He has published more than 130 technical articles, and is the editor/author of two books, *Multimedia Image and Video Processing* (Boca Raton, FL: CRC Press, 2000) and *Adaptive Image Processing: A Computational Intelligence Perspective* (Boca Raton, FL: CRC Press, 2001). He is an Associate Editor for the *IS&T/SPIE Journal of Electronic Imaging*, and the *Journal of Real-Time Imaging*.

Dr. Guan is an Associate Editor of IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION. In 1999, he co-guest-edited the special issues on Computational Intelligence for PROCEEDINGS OF THE IEEE. He also serves on the editorial board of CRC Press' Book Series on Image Processing. He has been or is currently involved in organizing numerous international conferences. He played the leading role in the inauguration of the First IEEE Pacific-Rim Conference on Multimedia in Sydney, 2000, and served as the General Co-Chair. Dr. Guan is a member of IAPR and SPIE. He is currently serving on IEEE Signal Processing Society Technical Committee on Multimedia Signal Processing.



Sun-Yuan Kung received his Ph.D. Degree in Electrical Engineering from Stanford University, Stanford, CA.

In 1974, he was an Associate Engineer of Amdahl Corporation, Sunnyvale, CA. From 1977 to 1987, he was a Professor of Electrical Engineering-Systems of the University of Southern California. Since 1987, he has been a Professor of Electrical Engineering at Princeton University, Princeton, NJ. He has authored more than 300 technical publications, including three books, *VLSI Array Processors* (Englewood Cliffs, NJ: Prentice-Hall, 1988) (with Russian and Chinese translations), *Digital Neural Networks* (Englewood Cliffs, NJ: Prentice-Hall, 1993), and *Principal Component Neural Networks* (New York: Wiley, 1996).

Dr. Kung has served as an Editor-In-Chief of *Journal of VLSI Signal Processing Systems* since 1990. He served as a Founding Member and General Chairman of various international conferences, including IEEE Workshops on VLSI SIgnal Processing in 1982 and 1986, International Conference on Application Specific Array Processors in 1990 and 1991, IEEE Workshops on Neural Networks and Signal Processing in 1991, 1992, and 1998, the first IEEE Workshop on Multimedia Signal Processing in 1997, and International Computer Symposium in 1998. He is a Fellow of the IEEE. He was the recipient of 1992 IEEE Signal Processing Society's Technical Achievement Award for his contributions on "parallel processing and neural network algorithms for signal processing." He was appointed as an IEEE-SP Distinguished Lecturer in 1994. He received the 1996 IEEE Signal Processing Society's Best Paper Award. He was a recipient of the IEEE Third Millennium Medal in 2000.