Utility-based Video Adaptation for UMA and Content-based Utility Function Prediction for Real Time Video Transcoding

Yong Wang*, Jae-Gon Kim, Shih-Fu Chang and Hyung-Myung Kim

Yong Wang* and Shih-Fu Chang are affiliated with the Department of Electrical Engineering, Columbia University, New York, NY 10027.

Jae-Gon Kim is affiliated with the Department of Broadcasting Media Technology, Electronics and Telecommunications Research Institute (ETRI), Daejeon, 305-350, Korea.

Hyung-Myung Kim is affiliated with the Department of Electrical Engineering & Computer Science, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 305-701, Korea.

Yong Wang* (Contact person)
1312 S.W.Mudd, 500 West 120th Street
New York, NY 10027
Phone: +1-212-854-7473
Fax: +1-212-932-9421
E-mail: ywang@ee.columbia.edu

EDICS: 1-CONV, 8 STDS (5-STRM), Media Type: video

ABSTRACT

Many techniques exist for adapting videos to satisfy heterogeneous resource conditions or user preferences. However, selection of the best adaptation operation among various choices usually is either ad hoc or inefficient. To provide a systematic solution, we present a conceptual framework based on utility function (UF), which models video entity, adaptation, resource, utility, and the relations among them. In order to support real-time video adaptation, we present a content-based statistical paradigm to facilitate the prediction of UF for real-time transcoding of live videos. Instead of modeling the UF through analytical models, as in the conventional rate-distortion framework, the proposed approach formulates the prediction as a classification and regression problem. Each video clip is classified into one of distinctive categories and then local regression is used to accurately predict the utility value. Our extensive experiment results based on MPEG-4 transcoding demonstrate that the proposed method achieves very promising performance — up to 89% accuracy in choosing the optimal transcoding operation (in both spatial and temporal dimensions) with the highest quality over a diverse range of target bitrates.

***Index Terms*: video adaptation, universal media access, utility function, content based prediction**

I. INTRODUCTION

An emerging multimedia framework, in which multimedia content is accessed from heterogeneous networks and terminals in a seamless way, is often referred to as *universal multimedia access* (UMA)[1]. In UMA, media content adaptation is considered to be a core technology for coping with variations of environment resources and user preferences. Media adaptation is a process that transcodes the original encoded media into a new version in order to match resource constraint (e.g., bandwidth and resolution) or user preference. Many adaptation methods exist for adjusting the bit rate of compressed video streams. For example, requantization of transform coefficients [2], frame dropping (FD) [3], DCT coefficients dropping (CD) [7] and resolution reduction [5] are commonly used. More discussion involving transcoding for UMA can be found in [4]. To address heterogeneous resources and user conditions, some recent developments in scalable video coding have been made with greater flexibility and improved video quality [17].

Nevertheless, most existing adaptation techniques have a common problem – they concentrate on optimization of pre-selected adaptation operations, rather than systematically choosing the optimal adaptation operation from multiple options. The issue becomes more prominent when the number of adaptation dimension increases, including spatial, temporal and SNR. In the literature there are a few efforts to address this issue. In [6] a rate-distortion (R-D) optimization method was proposed by modeling the Mean Squared Error (MSE) distortions caused by quantization and frame skipping. In [10], a dynamic programming scheme was used to achieve optimal rate control where frame rate, spatial resolution and quantization step size were jointly considered in modeling the distortion. A distortion measurement was used to estimate the video quality in the full resolution, while some weights were assigned to address the perceptual effects of spatio-temporal scale variation. In [11], variable frame rate coding was realized, where the quantization step size was

determined by an analytical distortion model for each frame, and the frame with quantization step size exceeding some threshold was skipped. However, all of these approaches rely on the availability of some analytical models. Construction of adequate analytical models of R-D relationship is known to be nontrivial. It is even more difficult when video undergoes multi-dimensional adaptation.

In this paper we present a general framework, called utility-based video adaptation, as a systematic solution for the issue of spatio-temporal combined adaptation. Specifically three key aspects involved in adaptation problems – adaptation (A), resource (R), and utility (U) are modeled and represented using a utility function (UF), which describes the tradeoff relationship between resources and utilities along each adaptation dimension. UF plays a key role in choosing the optimal adaptation among multiple options that meet resource constraints or user preferences. This approach represents a simple extension of conventional R-D framework to allow incorporation of diverse types of resources (e.g., complexity and bandwidth) and adaptation operations.

In the utility-based framework, one key issue is how to generate UF in real time to accommodate live videos. For stored videos in on-demand applications, UF can be generated by exhaustive off-line simulations. Such an approach may require significant computational complexity, which is unacceptable for live videos. In this paper we present a novel real-time UF prediction method that utilizes the strong correlations between content features and the UF characteristics of a video. The prediction method, first described in [21], combines real-time compressed-domain feature extraction, pattern discovery, classification, and statistical regression. We formulate the problem as a pattern classification and prediction question, taking the automatically extracted content features as input and then making predictions about the UF. The only on-line computation required is for content feature extraction and pattern classification. Therefore, the proposed approach is fully automatic and can be done in real-time. Our extensive MPEG-4 transcoding experiment results

show a very promising accuracy (up to 89%) in choosing the optimal operation from several competing options.

The main contributions of our work include the formulation of UF for the joint spatio-temporal adaptation and the novel algorithms for predicting the optimal adaptation operation based on the content feature extracted from the compressed streams. The rest of this paper is organized as follows. The framework of utility-based transcoding is introduced in Section II. In Section III, the statistical approach to UF prediction is described, including feature extraction, unsupervised and supervised learning methods and statistical local regression. The experiment setup and results are presented in Section IV. The conclusion and future work are given in Section V.

II. UTILITY-BASED TRANSCODING

The UF-based adaptation approach mentioned above fits very well a popular three-tier server-proxy-client architecture, shown in Figure 1. The adaptation engine deployed in the proxy adapts incoming videos to satisfy dynamic resource constraints that are not known a priori. The role of the UF is to describe the relationship between required resources and resulting video utilities when the video is subject to various adaptation operations in multiple dimensions. For stored videos, UF can be generated offline at the server and sent to the adaptation engine. The engine will then select the optimal adaptation operation based on the information in the UF. For live videos, UF needs to be obtained on the fly through some estimation and update processes. In this paper, we specifically propose a content-based prediction method that estimates the UF according to the content features and statistical classification tools. Such real-time prediction methods can be implemented at either the server or the proxy.

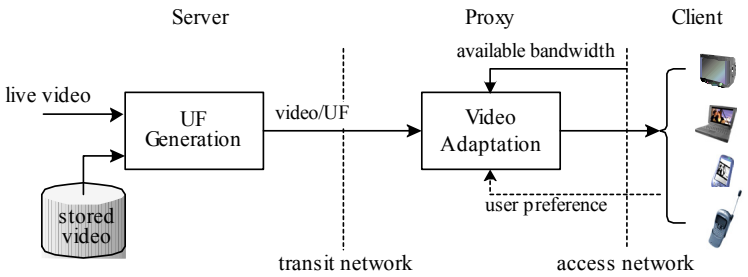


Figure 1. A three-tier adaptation architecture using the utility-based framework.

A. DEFINITION OF ADAPTATION, RESOURCE, UTILITY AND THEIR RELATIONS

UF is defined in the Adaptation-Resource-Utility (ARU) space, where relationships among diverse types of adaptations, resources (e.g., bandwidth, power, and display) and utilities (e.g., objective or subjective quality) are modeled. We use the term "space" in a loose sense here to indicate the multiple dimensionalities involved. Figure 2 depicts the notions of ARU involved in a video adaptation problem. The entity, e, refers to the basic unit of video data that undergoes the adaptation process. Adaptation operators are the methods to reshape the video entities, such as requantization and frame dropping. All permissible adaptations for a given video entity constitute the adaptation space. Resources are constraints from terminals or networks, including bandwidth, display resolution, power, etc. Utility represents the quality of an entity when it is rendered on an end device after adaptation, such as PSNR, perceptual quality, or even high-level user satisfaction. The mapping relationship among ARU spaces is illustrated in Figure 3. Typically, there exist multiple adaptation solutions that satisfy the same resource constraints, while yielding different utilities. In Figure 3, the points in the oval shaped region in the adaptation space indicate such a constant-resource region. Likewise, different points in the adaptation space (the shaded rectangle) may lead to the same utility value. It is such a multi-option situation that makes the adaptation problem interesting – we want to choose the optimal one with the highest utility or minimal

resource.

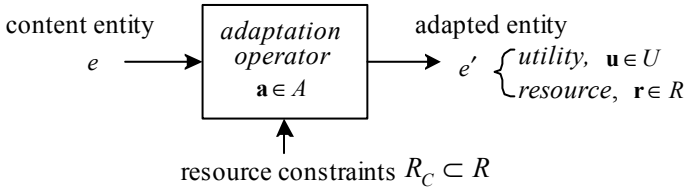


Figure 2. Definition of adaptation, resource, and utility spaces involved in video adaptation problems in the utility-based framework.

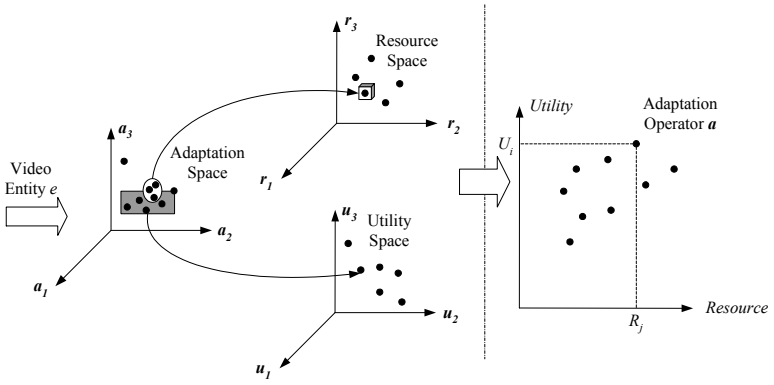


Figure 3. Use of UF to describe relations among adaptation, resource and utility

We are interested in describing the relationship between rate and utility associated with each adaptation operation. We represent such relationship by using UF. The right figure in Figure 3 shows a simple example of UF, in which only one dimension is shown in both resource and utility. This is equivalent to the known R-D curve when R is bitrate and D is related to video quality. Each point in UF is associated with one specific adaptation operator, which may include combinations of multiple operations (such as frame dropping and coefficient dropping). More detailed discussion of the UF function can be found in [12].

B. FD-CD ADAPTATION

To illustrate our method of utility based adaptation without losing generality, in this paper we consider a specific case involving two types of adaptations — Frame Dropping (FD) and AC DCT

Coefficient Dropping (CD) and their combinations (FD-CD). FD adapts the source stream by skipping frames, while CD transcodes the source stream by truncating some high frequency DCT coefficients. For CD there is more than one choice during coefficient dropping. Therefore in order to eliminate the ambiguity and obtain the optimal CD performance, the Lagrange optimization method is employed. Typically, suitable rate control techniques are needed to meet specific bandwidth constraint after FD-CD adaptation. Due to the space limitation details about the FD-CD algorithm and its implementations can be found in [16].

The advantage of FD-CD adaptation firstly lies in its simplicity allowing real time implementation. Also FD can meet a coarse level of the target rate since its processing data unit is a frame. CD is able to meet the target rate with a finer granularity by adjusting the amount of dropped coefficients. The combination of FD-CD accommodates a wide range of bit rate constraints. Furthermore, FD-CD provides adequate flexibility in balancing the trade-off between spatial and temporal quality.

For simplification, we assume the entity undergoing adaptation is a group of pictures (GOP) in the MPEG-4 sequence. Namely the same FD-CD operation parameters will be applied to the whole GOP.

C. FD-CD REPRESENTATION USING UTILITY FUNCTION

Using the utility-based adaptation framework, a two-dimensional adaptation space can be constituted for FD-CD, in which both FD and CD entail a finite set of adaptation operations. Specifically, an FD-CD adaptation method can be expressed as $\mathbf{a} = (f, c)$, where f and c represent a specific frame dropping method and a coefficient dropping method respectively. For instance, $\mathbf{a} =$ (all B-frames dropped, 10%) means all of the B frames in a GOP are dropped and 10 percent of the bits from each remaining frame will be reduced by coefficient dropping. A typical UF

is shown in Figure 4. For a specific video clip, given an adaptation operator $\mathbf{a}$, its corresponding resource and utility value are denoted as r and u. In the case of FD-CD, we have coarse discrete values of FD (i.e., "no frame dropping", "drop all B and P frames", "drop all B frames", and "drop 1 B frame only"), and finer discrete values of CD (i.e., drop c% of DCT coefficients"). Thus in Figure 4, points with the same FD are connected to a curve, and the adaptation operations between two anchor nodes are obtained through linear interpolation. The whole set of the curves define the UF, which represents the utility-resource relation associated with the given video in response to the available adaptation operations (FD-CD). Given a resource constraint r_0, all of the possible operators meeting the same resource constraint, such as $\mathbf{a}_A$ and $\mathbf{a}_B$ in Figure 4 can be found from the UF. If an operation is selected using the actual UF, it will achieve the target resource and the utility when it is applied to the video. If the operation is selected based on predicted UF, the actual resource and utility resulting from the adaptation may be slightly different from the target values due to prediction errors. Such utility based adaptation mechanism was also accepted as a part of MPEG-21 Digital Item Adaptation (DIA) [19]. More information about the DIA utility-based description tool can be found in [22].

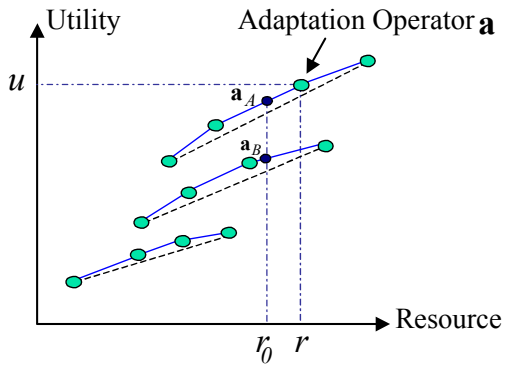


Figure 4. Definition and representation of utility function

To obtain a more efficient representation, we further simplify the representation of the UF by using the linear approximation of each curve as shown in Figure 4. The approximation is defined by

two end nodes of each curve. Therefore the UF can be denoted as:

$$\mathbf{F}^{UF} = \left(f_1^{UF}, f_2^{UF}, ..., f_n^{UF} \right) = \left(r_1, r_2, ..., r_{2n}, u_1, u_2, ..., u_{2n} \right)$$

where each curve $\left(f_i^{UF}, i = 1, 2, ..., n \right)$ in Figure 4 is associated with 2 end points. Such approximation representation is very beneficial in reducing the dimensionality of the representation and improving the efficiency of the statistical prediction method described later. Our experiment demonstrates that such linear approximation provides a very satisfactory result in UF prediction (shown in Section IV.B). The ordering of the nodes does not matter, as long as a consistent scheme is maintained.

D. ISSUES IN COMPUTING THE UTILITY FUNCTION

In practice the generation of the UF is a nontrivial process. It may be done by exhaustive computation of all of the adaptation points, each of which requires transcoding of the video, decoding the transcoded bit stream, and computing the distortion. This process is very time consuming and typically cannot be done efficiently. To avoid exhaustive computation, there are two possible solutions: analytical modeling or empirical estimation.

Approximate Analytical Modeling

In [11], some analytical source models were developed by extending the theoretical R-D curves derived from ideal statistical distributions to approximate models using empirical data. Certain statistical models (e.g., Gaussian, Laplacian or variations) were assumed for video signals and parameters of the models were computed and updated from input video. Given the approximate R-D information, some recent methods have been developed to automatically adjust the frame rate and quantization step size under low-bit-rate conditions [10]. However, such analytical models may not be valid in general due to several reasons.

First, the adopted signal models like Gaussian or Laplacian may not be valid for realistic signals. Second, the R-D relationship is greatly affected by the specific coding algorithm, which has become increasingly complex in recent video coding technologies. Simple statistical signal models may not be valid for such complex coding methods. Lastly, it is difficult to extend the analytical models in order to take into account different coding structures, utilities (e.g., subjective measures), and resources (e.g., power).

Empirical Estimation and Content-Based Prediction

Another approach to R-D estimation is based on empirical learning – namely, learning from the training data. Such an approach does not use explicit statistical models for the video signals to derive the R-D curves. Instead, collection of training video clips are used to generate samples of video content features and the resulting UFs, represented by some efficient schemes described in the previous section. Machine learning techniques are then applied to develop mapping functions from the content features to the UFs. We refer to the aforementioned approach as *content-based utility function prediction*.

The above prediction methods explore the potential correlation between content features and the R-D characteristics of a video. Such correlation has been observed in our experimental observations (Section IV). Here for video content, we refer to low-level features such as motion, spatial complexity, and characteristics of the coded stream (e.g., number of inter-frame coded macroblocks, motion vector statistics, etc.). Such features can be efficiently computed from the compressed streams. In our prior work [8], we have explored such an approach in which visual features from the video objects are used to predict the subjective quality of the objects after undergoing MPEG-4 transcoding. However, the work in [8] did not explore systematic representations of the UFs for FD-CD adaptation and did not address issues related to prediction of the optimal spatio-temporal

adaptation operation.

III. UTILITY FUNCTION PREDICTION

A. PROBLEM DESCRIPTION

The issue of UF prediction can be formalized as follows: given the content feature $\mathbf{F}^{CF}$ of one video clip, develop a suitable mapping from the content feature space into the UF space, i.e.,

$$\mathbf{F}^{UF} = G\left(\mathbf{F}^{CF}\right) \tag{1}$$

where $\mathbf{F}^{UF} = \left(f_1^{UF}, f_2^{UF}, ..., f_N^{UF}\right)$ is a *N*-dimension UF row vector and f_i^{UF} is i^{th} component of $\mathbf{F}^{UF}$, and similarly $\mathbf{F}^{CF} = \left(f_1^{CF}, f_2^{CF}, ..., f_M^{CF}\right)$ is the *M*-dimension content feature row vector and f_j^{CF} is j^{th} component of $\mathbf{F}^{CF}$. Equation (1) is a typical multivariate regression problem. For each f_i^{UF} in $\mathbf{F}^{UF}$, we want to find a mapping g_i, such that

$$\begin{aligned} &f_i^{UF} = g_i\left(\mathbf{F}^{CF}\right) = g_i\left(f_1^{CF}, f_2^{CF}, ..., f_M^{CF}\right) \\ &G = \left(g_1, g_1, ..., g_N\right) \end{aligned} \tag{2}$$

By using Taylor expansion, this mapping can be written as:

$$f_i^{UF} = g_i\left(\mathbf{F}_0^{CF}\right) + \nabla \mathbf{g}_i\left(\mathbf{F}_0^{CF}\right) \cdot \left(\mathbf{F}^{CF} - \mathbf{F}_0^{CF}\right) + O\left(\left|\mathbf{F}^{CF} - \mathbf{F}_0^{CF}\right|^2\right). \tag{3}$$

where $(\cdot)$ is the dot product of two vectors, and $\nabla \mathbf{g}_i\left(\mathbf{F}_0^{CF}\right)$ is the *M*-dimension partial differential row vector. By keeping the components in (3) up to first order and ignore the higher orders, this mapping can be considered as a classic linear regression problem. Based on Equation (3), we can derive the following:

$$\mathbf{F}^{UF} = \left(f_1^{UF}, f_2^{UF}, ..., f_N^{UF}\right) = \left[\mathbf{F}^{CF} \quad 1\right]\begin{bmatrix} \nabla \mathbf{g}_1^T & \nabla \mathbf{g}_2^T & \cdots & \nabla \mathbf{g}_N^T \\ c_1 & c_2 & \cdots & c_N \end{bmatrix}$$
$$c_i = g_i\left(\mathbf{F}_0^{CF}\right) - \nabla \mathbf{g}_i(\mathbf{F}_0^{CF}) \cdot \mathbf{F}_0^{CF} \tag{4}$$
$$\nabla \mathbf{g}_i^T = \nabla \mathbf{g}_i^T(\mathbf{F}_0^{CF}),\ i = 1, 2, ..., N$$

By applying the standard Least Mean Square Error (LSE) method, the optimal estimation of g_i, indicated as $\hat{g}_i$, can be found to be:

$$\hat{\mathbf{g}}_i = \left(\left(\tilde{\mathbf{F}}^{CF}\right)^T \tilde{\mathbf{F}}^{CF}\right)^{-1}\left(\tilde{\mathbf{F}}^{CF}\right)^T \tilde{\mathbf{F}}_i^{UF} \tag{5}$$

where $\tilde{\mathbf{F}}^{CF}$ is the set of observed content feature vectors in the training data with each row corresponding to a training sample, and $\tilde{\mathbf{F}}_i^{UF}$ is the corresponding i^{th} component of observed UF. Moreover, the Taylor expansion works only in a small neighbor of the center $\mathbf{F}_0{}^{CF}$. Thus the first-order approximation is effective if the content feature space can be divided into some small areas, and the regression procedure is applied for each area separately. Specifically, this can be done by forming K such subareas S_k, k=1,2…K, and conducting the above approximate estimation method for points within each subarea. Therefore, the problem can be modeled as a K-segment piecewise linear regression problem and the parameters (c_i and $\mathbf{g}_i$) can be obtained for each subset. In forming the partitions of the space, we can consider clustering in the CF space, clustering in the UF space, or a hybrid one that partition the CF space subject to some compactness constraints of corresponding UF values. In this work, in order to apply the local regression method discussed above we chose clustering in the CF space, plus combinations of classification techniques mapping content features to the CF clusters. Our experiment results presented later indeed confirm the superiority of this choice.

Figure 5 shows the overall architecture of the proposed framework. The top diagram shows the procedures of extracting content features, classifying the video to one of the UF

classes, and local regression for predicting the UF. In the bottom diagram, an offline mechanism is shown to illustrate the use of a training pool in developing the UF clusters, the classifier for mapping future clips, and the local regression method for each cluster. Each training clip is associated with the content features and the actual UFs that are obtained in advance through exhaustive computation of all the adaptation operations. Details of each component mentioned above will be described in the following subsections.

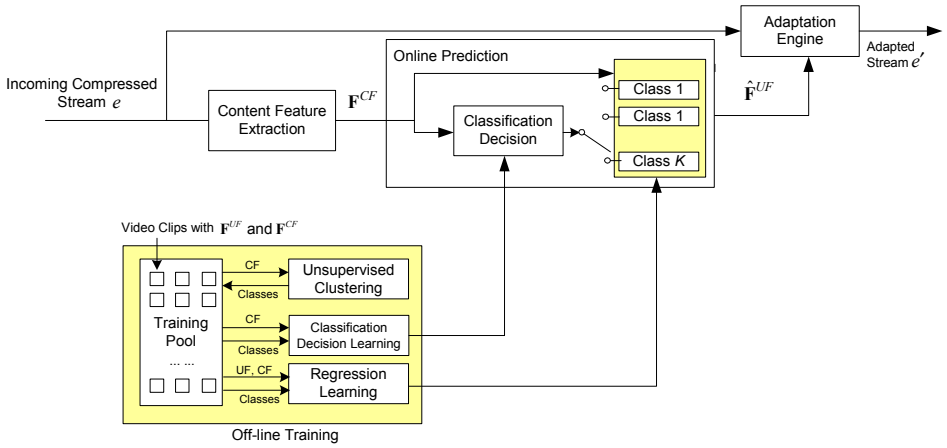


Figure 5. Overall architecture of the proposed framework.

B. CONTENT FEATURE EXTRACTION

We adopt the content features based on the set adopted in our prior work [8] with minor modification. Three groups of features are considered: motion intensity, AC DCT energy, and quantization parameters. The first two groups of features embody the spatial texture complexity and temporal motion intensity information. The third group also indirectly reflects the scene complexity subject to the specific rate control algorithm used. They are extracted directly from the encoded stream or the stream metadata without decoding the video to the pixel domain. Our experimental results show that the performance of prediction can be improved if we also include the PSNR information from the metadata associated with the original encoded stream.

Content features are extracted from each local video segment that is one second long. The length of the local segment is currently empirically determined, to keep an adequate balance between efficiency and accuracy. Note to ensure the video content in each segment is more or less consistent, we avoid shot boundaries within a segment by running automatic shot boundary detection and keeping the shot boundaries aligned with the segment boundaries. Although the shot boundary detection tool is not perfect, performance of the existing detection tools is quite high (precision up to 97% and recall up to 98% in [20]).

Specifically, the following features are used in our system:

1) Average motion intensity approximated by computing motion vector magnitude;
2) Motion variance within the adaptation unit;
3) Average percentage of macroblocks which have non-zero motion vector;
4) Average I frame AC DCT coefficient energy;
5) Average P frame AC DCT coefficient energy;
6) Average quantization step size;
7) Average PSNR if available in the stream metadata.

The average values are computed over the frames in the one-second segment. To further improve efficiency, we only process the I and P frames. The AC DCT energy of I and P frames are kept separate because our statistical feature analysis (Principal Component Analysis) shows they have distinctive contributions to the final performance. This is reasonable considering the DCT energy in the I frame is more related to the texture complexity due to the use of intra-frame coding, while the DCT energy in the P frames is mainly related to motion compensation residues because of the use of inter-frame coding.

C. UNSUPERVISED CLUSTERING

The purpose of unsupervised clustering is to partition the content feature space into

separate subspaces so that the local regression technique described in Section III can be applied in each local area. We adopt the K-Harmonic Mean (KHM) [13] clustering method, which in principle is related to the popular K-mean method. The main improvement of KHM over K-mean is by using the p^{th}-order harmonic distance, rather than the Euclidian distance. It was shown in [13] that KHM outperform K-mean in reducing the sensitivity to initialization and avoiding local optimal points.

Note the above clustering process is performed in the CF space, instead of the UF space. As shown in Figure 6, clusters formed in the CF space will ensure points in the same cluster have similar CF values. This is important for keeping a subarea of small variation of CF values and thus the first-order approximation by Taylor expansion described in Section III.A remains valid. Although the alternative of doing clustering in the UF space can achieve compact data sets with similar UF values. The corresponding values in the CF space may be spread over a large range, and thus violate the assumption of proximity of the local regression method mentioned above. We will present performance comparison of these competing options later.

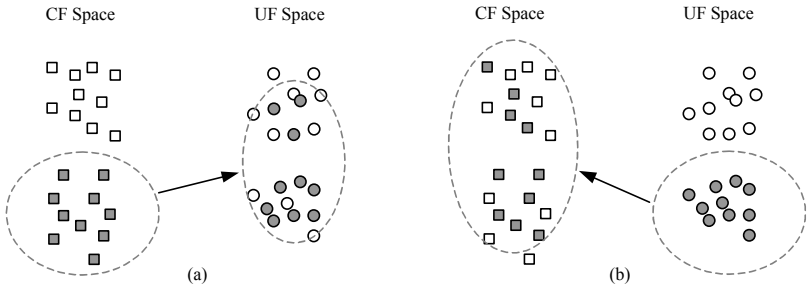


Figure 6. Difference between CF clustering and UF clustering. Shaded points show a cluster formed in one space and the corresponding values in the other space.

Another general problem of unsupervised clustering is determining the number of cluster,

K. A K value that is too large will lose generality and result in overfitting, while a K value that is too small will result in significant bias. In our experiment, we determine the number of cluster through empirical trials and find $K = 16$ yields satisfactory performance. We expect the adequate choice of K depends on the characteristics of the video content and dynamic variations of the video over time. It is conceivable to propose some prediction schemes to determine the cluster number based on computable content features. Study of such methods and analysis of the effect on the UF prediction performance is beyond the scope of the current work.

D. SUPERVISED CLASSIFICATION BY SVM

The purpose of classification is to categorize an incoming video clip into one of the classes and then apply the corresponding regression model to predict the UF. Note if the classes are formed by clustering in the CF space, the same clustering method can be used for such a classification purpose. But if the clusters are formed in the UF space, the corresponding points in the CF space may not be compact and therefore we need a separate process for classification.

We employ Support Vector Machine (SVM) for the classification task. Basic SVM classifiers are for two-class discrimination. There are several ways to extend a binary classifier to support multiple-class separation, such as classifiers for one against others [15], or ones that fuse a set of two-class classifiers by methods like the Max Wins algorithm in [9]. We adopt the directed acyclic graph SVM (DAGSVM) algorithm presented in [14] with minor modification to resolve the ambiguous region issue. In DAGSVM, the multiple-class classifier is constructed by using a Decision Directed Acyclic Graph. The classifier starts with separation between two most distinguishable classes using a regular two-class

SVM. The negative class is excluded and the same two-class discrimination procedure is repeated for the remaining classes. It has been known to be a fast multi-class classifier with satisfactory performance [14].

IV. EXPERIMENT RESULTS

A. EXPERIMENT SETUP

In our experiment, we selected video from three movies to form the training and testing pool. The details of the video pool are summarized in Table 1. There were totally 2066 clips, each of which was one second long. The clips were carefully selected to cover a wide range of content features. Every clip was extracted from within a shot and thus no abrupt transitions like shot changes occured within a clip. The proposed algorithm was tested using a standard cross validation procedure in which training and testing was done with random partitions of the pool (70% for training and 30% for testing) over multiple runs.

First, we need to compute UFs and extract content features for each set of training clips. In computing UFs, we defined an adaptation space of FD-CD similar to that described in Section II.C (see Figure 4). Based on the given GOP structure $(N=15, M=3)$, we adopt four FD operators: "no frame dropped", "the first B frame dropped in each sub GOP", "all B frames dropped", and "all B and P frames dropped". In the CD dimension, we adopted six CD levels: from 0% to 50% with 10% increment. As a result, there were totally 24 anchor nodes and four operation curves in each UF. Further details of the implementation are described in [16].

Table 1: Summary of data set

Video source	1. *A Beautiful Mind* (736 clips) 2. *Crouch Tiger Hidden Dr*agon (589 clips) 3. *Taxi II* (741 clips)
Clip length	1 second (2 GOPs)

Image format	352 x 240 pixels
Video compression	MPEG-4 with 30 fps and TM5 rate control
GOP structure	GOP size N=15, sub-GOP size M=3

Evaluation of the proposed prediction method can be based on various performance metrics. For example, errors in predicting the UF can be defined based on the L_2 metric as follows

$$D = \frac{1}{L}\sum_{l=1}^{L}\left\|\mathbf{F}_l^{UF} - \hat{\mathbf{F}}_l^{UF}\right\|^2 \tag{6}$$

where $\mathbf{F}_l^{UF}$ is the actual UF and $\hat{\mathbf{F}}_l^{UF}$ is the predicted one. L is the number of the test clips. Alternatively, the utility ranking of permissible operators at fixed bitrates can be evaluated, comparing results using the predicted UF verse the ground truth UF.

Table 2 is the specification of the algorithms employed in the experiment.

Table 2: Algorithm specification

Unsupervised Clustering	*K*-Harmonic Mean (*KHM*) using *p*-th order harmonic distance. $p = 0.5$, Number of clusters *K*=16
Classification	DAGSVM multiple-class classification: *C*=100, kernel=RBF with $\gamma = 0.5$
Linear Regression	Trained by LSE algorithm. See Equation (5).

B. PERFORMANCE

Figure 7 shows the prediction errors from four methods: our proposed method (Content Feature Clustering based Regression, CFCR); our proposed method but without local regression (Content Feature Clustering based Classification, CFCC), an alternative approach using clustering in the UF space instead of the CF space (UF-Clustering based Classification, UFCC), which is adopted in [8], and UF-Clustering based Regression (UFCR). The prediction error is measured by the L2 distance between the true UF and the predicted UF (see Equation 6). The experiments were run for 10 times and the average

performance was computed. The proposed method (CFCR) achieves the best result. That is to say when classification is combined with regression, clustering in the CF space is the best. This validates our decision in adopting the CF-space clustering method. However, it is interesting to note that without regression, techniques using clustering alone (UFCC and CFCC) favors clustering in the UF space. This is consistent with the UF-space clustering techniques used in our prior work [8]. Note in the pure clustering approach, the representative UF of each cluster as used as the predicted UF for all the points mapped to the same cluster.

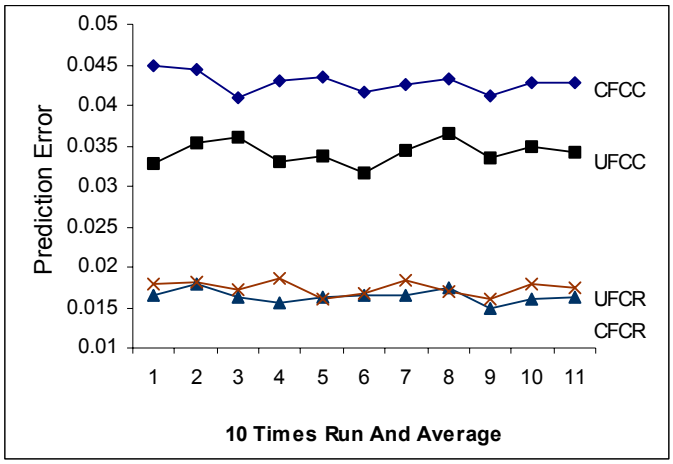
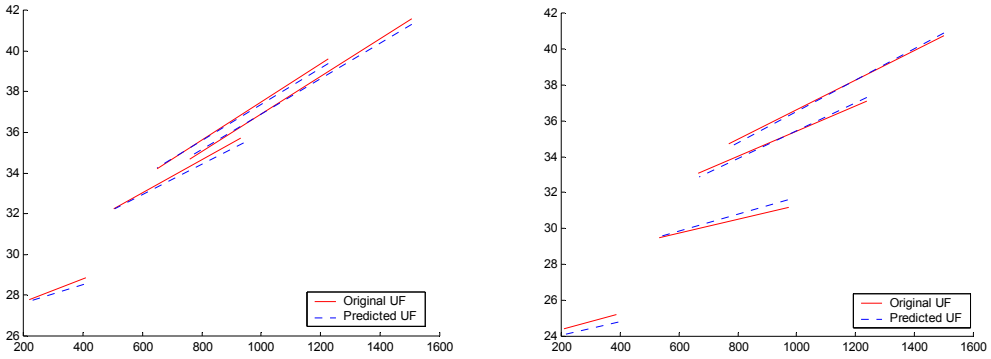


Figure 7. Comparison of the prediction performance in terms of prediction error.

Figure 8 shows comparison between some predicted UFs and the corresponding ground truth. The predicted UFs indeed match the true values very well. Typically, the prediction of the utility value (y axis) is not as good as the prediction of the resource value (x axis). However, the ranking of utility values among different transcoding options are quite consistent. Such ranking information provides the most important input to our adaptation system for selecting the optimal transcoding option meeting a given target resource constraint.



(a) (b)

Figure 8. Matching predicted UF to the ground truths.

In addition, we also measured the accuracy in selecting the optimal operator given various target bit rates. Five typical bandwidths were used as the test target rates: 1.2 M, 1.0 M, 800 K, 480 K and 320 K bps. The original input video rate before transcoding was 1.5 Mbps. Our proposed method (CFCR) was compared with two alternatives: CFCC and the most frequent adaptation method. The latter did not take into account content features in each video, and simply selected the operation that achieves the highest quality for the most number of video clips in the training pool. Figure 9 shows our method outperforms the other two and exhibits significantly higher accuracy (up to 89%).

From both the above evaluation criteria, the results are quite encouraging – the proposed content-based prediction method achieves very good accuracy in predicting the UF values as well as the ranking among competing adaptation operations.

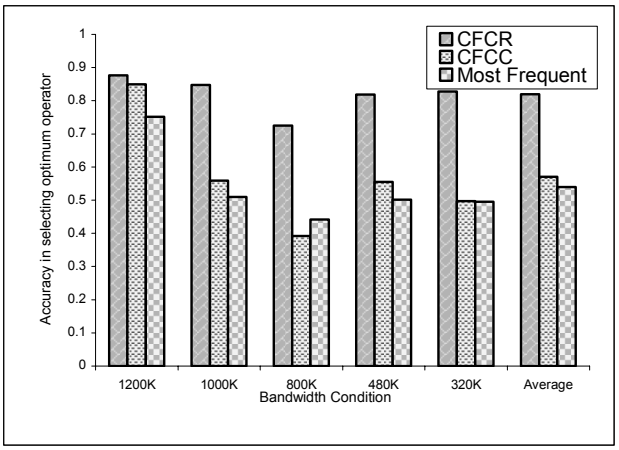


Figure 9. Performance in terms of prediction accuracy in choosing the optimal operator.

Besides prediction performance, computational complexity is another important factor for a real time application scenario. Because the MPEG-4 codec we used was not a real-time implementation, we were not able to provide the real time benchmark data. However, all of the computation processes in our system are light-weight. As shown in Figure 5 the main costs in our system include feature extraction and online prediction. The online prediction process, including classification and regression, can be implemented efficiently. Specifically, SVM classification only needs to calculate the kernel function and dot product between the content features and a sparse set of support vectors; linear regression involves only a multiplication between the model matrix and content feature vector. For feature extraction, partial bit stream decoding is necessary in order to obtain the content features, plus some minor extra calculation such as computing averages. The combination of all these computation steps is still much lighter than the complexity of a regular decoder (because the most complex component, motion compensation, is not needed). Considering video decoders can be implemented on most platforms with a real-time performance, it is reasonable to conjecture that our system can be implemented in a real-time fashion as well.

V. Conclusions And Future Work

In this paper we present a utility-based video adaptation framework as a systematic methodology to meet diverse resource conditions and user preferences in UMA environments. The framework explicitly models the major concepts involved in adaptation processes – adaptation, resource, and utility using a UF. In order to address the computational complexity issue in UF generation and support the real time adaptation scenario, we further propose a general content-based UF prediction approach using automatic content feature extraction, and regression over clustering and classification. Our experiment results using MPEG-4 FD-CD transcoding demonstrate very promising prediction accuracy over diverse types of video content.

The proposed content-based utility prediction framework is general and can be expanded to handle heterogeneous scenarios where various resources, adaptations, or utilities are employed. Recently we have successfully expanded our framework to scalable video coding and subjective evaluation utility case with satisfactory results [18]. Future work will include extensions that consider multiple utilities and resources at the same time. An example scenario is to find the balance between bandwidth demand and power consumption in selecting appropriate adaptation for handheld devices.

VI. Acknowledgement

This work has been partly supported by Electronics and Telecommunications Research Institute of Korea. We thank Dr. K. Kang and Dr. J. Kim of ETRI for their discussions and suggestions. We also thank the anonymous reviewers for careful reviews and valuable comments.

REFERENCE

[1] R. Mohan, J. R. Smith, and C.-S. Li, "Adapting multimedia Internet content for universal access," IEEE Trans. Multimedia, vol. 1., pp. 104-114, Mar. 1999.

[2] O. Werner. "Requantization for transcoding of MPEG-2 intraframes", *IEEE Trans. Image Processing*, vol. 8, pp. 179 –191, Feb. 1999.

[3] T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding to lower spatio-temporal resolutions and different encoding formats," *IEEE Trans. Multimedia*, vol. 2, pp. 101-110, June 2000.

[4] N. Björk, C. Christopoulos, "Video transcoding for universal multimedia access", in *Proc. ACM Multimedia Workshops 2000*, pp. 75-79.

[5] P. Yin, M. Wu, B. Liu, "Video Transcoding by Reducing Spatial Resolution", In *Proc. IEEE Int. Conf. on Image Processing*, Vancouver, 2000.

[6] A. Vetro, Y. Wang, and H. Sun, "Rate-distortion optimized video coding considering frameskip," In *Proc. IEEE Int. Conf. on Image Processing*, Rochester, New York, Oct. 2002.

[7] A. Eleftheriadis and D. Anastassiou, "Constrained and general dynamic rate shaping of compressed digital video," In *Proc. IEEE Int. Conf. on Image Processing*, Washington, DC, Oct. 1995, pp. 396-399.

[8] P. Bocheck, Y. Nakajima and S.-F. Chang, Real-time estimation of subjective utility functions for MPEG-4 video objects, In *Proc. Packet Video*, New York, USA, Apr., 1999.

[9] T. Berger, *Rate Distortion Theory*. Englewood Cliffs, New York: Prentice Hall, 1971.

[10] E. C. Reed and J. S. Lim, "Optimal multidimensional bit-rate control for video communications," *IEEE Trans. Image Processing*, vol. 11, no. 8, pp. 873-885, Aug. 2002.

[11] H.-M. Hang and J.-J. Chen, "Source model for transform video code and its application, Part I and II" *IEEE Trans. Circuits Syst.. Video Technol.*, vol. 7, pp. 287-311, Apr. 1997.

[12] S.-F. Chang, "Optimal video adaptation and skimming using a utility-based framework," In *Proc. Int. Work. Digital Comm.*, Capri Island, Italy, Sept. 2002.

[13] B. Zhang, "Generalized K-harmonic means-boosting in unsupervised learning," Hewlett-Packard Lab. Technical Report, Oct. 2000.

[14] J. C. Platt, N. Cristianini, and J. Shawe-Taylor, "Large margin DAGs for multiclass classification," in S.A. Solla, T.K. Leen, and K.-R. Mller editors, Advances in Neural Information Processing Systems 12, pp. 547-553, MIT Press, 2000.

[15] V. Vapnik, *Statistical Learning Theory*. Wiley, New York, 1998.

[16] Y. Wang, J.-G. Kim, S.-F. Chang, "MPEG-4 Real Time FD-CD Transcoding," Columbia

University ADVENT Technical report #208-2005-2, 2003.

[17] P. Chen, J.W. Woods, "Bi-directional MC-EZBC with Lifting Implementation", IEEE Trans. Cir. Sys. For Video Tech. No. 10, October 2004, pp. 1183-1194.

[18] Y. Wang, T.-Tsong Ng, Mihaela van der Schaar, S.-F. Chang, “Predicting optimal operation of MC-3DSBC multi-dimensional scalable video coding using subjective quality measurement,” in Proc. SPIE Video Comm. and Image Processing (VCIP), 2004.

[19] D. Mukherjee, E. Delfosse, J.-G. Kim, Y. Wang, Terminal and Network Quality of Service, invited paper, IEEE Trans. On Multimedia Special Issue on MPEG-21. 2004

[20] D. Zhong. Segmentation, Index and Summarization of Digital Video Content. PhD Thesis. Graduate School of Arts and Sciences, Columbia University, 2001.

[21] Y. Wang, J.-G. Kim, S.-F. Chang. Content-based utility function prediction for real-time MPEG-4 transcoding. In IEEE International Conference on Image Processing (ICIP), Barcelona, Spain, September 2003.

[22] J.-G. Kim, Y. Wang, S.F. Chang. Content-Adaptive Utility-Based Video Adaptation. In IEEE International Conference on Multimedia and Expo (ICME), Baltimore, Maryland, July 2003.