Joint Source/Channel Coding Based on Two-Dimensional Optimization for Scalable H.264/AVC Video

Xiao-Feng Li, Ning Zhou, and Hong-Sheng Liu

The scalable extension of the H.264/AVC video coding standard (SVC) demonstrates superb adaptability in video communications. Joint source and channel coding (JSCC) has been shown to be very effective for such scalable video consisting of parts of different significance. In this paper, a new JSCC scheme for SVC transmission over packet loss channels is proposed which performs two-dimensional optimization on the quality layers of each frame in a rate-distortion (R-D) sense as well as on the temporal hierarchical structure of frames under dependency constraints. To compute the end-to-end R-D points of a frame, a novel reduced trellis algorithm is developed with a significant reduction of complexity from the existing Viterbi-based algorithm. The R-D points of frames are sorted under the hierarchical dependency constraints and optimal JSCC solution is obtained in terms of the best R-D performance. Experimental results show that our scheme outperforms the existing scheme of [13] with average quality gains of 0.26 dB and 0.22 dB for progressive and non-progressive modes respectively.

Keywords: Scalable video coding (SVC), joint source/channel coding, unequal error protection, H.264/AVC.

I. Introduction

With the rapid development of network infrastructure and wireless communication, various multimedia applications with digital video have become popular. To transmit and store video efficiently, advanced source compression techniques are used to remove as much redundancy as possible. Compressed video is typically sensitive to bit errors, therefore error protection mechanisms have to be adopted for a video transmission system to combat the channel errors. According to Shannon's principle, the source and channel coding can be performed separately, with the assumption that infinite length of data blocks and unlimited coding complexity are acceptable. In practical applications, joint source and channel coding (JSCC) has shown better results by optimally allocating the available resources between source and channel coders.

The scalable extension to H.264/AVC is a new video coding standard, known as Scalable Video Coding (SVC) [1]. It was developed by the Joint Video Team (JVT) of the ISO/IEC Moving Pictures Experts Group (MPEG) and the ITU-T Video Coding Experts Group (VCEG) [2]. In addition to AVC [3], SVC provides spatial, temporal, and quality scalabilities. With an SVC encoder, a video signal is encoded into a high-quality bit stream that contains a base layer and one or more enhancement layers. The enhancement layers improve the resolution (either spatial or temporal) or quality of the video signal. Therefore the SVC is very suitable for applications in a wide range of varying network conditions and different terminal capabilities. However, due to the extensive use of content-based entropy coding, an SVC stream is very sensitive

Manuscript received Apr. 13, 2010; revised Aug. 2, 2010; accepted Aug. 17, 2010.

This work was supported by the National 863 hi-tech project, China (2009AA01Z215, Study on the architecture of multi-services oriented general burst optical network).

Xiao-Feng Li (phone: +86 28 61830690, email: xfli@uestc.edu.cn), Ning Zhou (email: zhouning@uestc.edu.cn), and Hong-Sheng Liu (email: liuhongsheng_66@163.com) are with the School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, China.

doi:10.4218/etrij.11.0110.0220

to bit errors which may cause desynchronization of decoder and eventually lead to failure of video recovery. The application of JSCC to SVC has been an active area of research recently.

Error protection in many transmission systems is typically enabled by means of forward error correction (FEC) codes, which add controlled redundancy to the transmission data for reliable error detection and correction. Since an SVC stream consists of layers of different importance, unequal error protection (UEP) is often adopted in which a channel coder of variable code rate or a set of coders with different code rates are used to protect different sections of the stream.

An early JSCC scheme in the area of scalable video was presented in [4] for transmission of 3D subband video over noisy channels. Distortion functions are derived for each subband, and a joint bit allocation is performed by employing a Lagrangian multiplier method. Many JSCC methods have been studied for scalable video and image coding based on discrete cosine transform and wavelet transform [5]-[9]. One of the first JSCC algorithms for the new SVC standard proposes a rate allocation mechanism for the transmission of SVC stream over wireless MIMO systems [10]. The end-to-end distortion is computed exhaustively for each combination of many source and channel parameters, and the optimal rate allocation is derived based on Lagrangian optimization. A low complex UEP method using Reed-Solomon codes for SVC is proposed in [11]. In [12], a UEP method using low-density parity check codes is proposed with an optimal bit allocation algorithm based on dynamic programming. Recently, a new JSCC scheme has been proposed in [13], which minimizes the end-to-end distortion for the transmission of SVC streams over packet loss channels. Employing a Viterbi-based search algorithm and Lagrangian optimization, the scheme achieves competitive results against state-of-the-art Lagrangian-based algorithms with significantly reduced complexity. As the scheme ignores the dependency constraints in temporal structure, it results in performance degradation. Two-dimensional channel coding was proposed in [14] which relies on accurate estimation of overall distortion of the reconstructed frames and requires much computation for optimization.

In this paper, we propose a new efficient JSCC scheme performing two-dimensional optimization on the quality layers of each frame in a rate-distortion (R-D) sense as well as on the temporal hierarchical structure of frames under dependency constraints. The optimal JSCC solution is obtained in terms of best R-D performance. Experimental results demonstrate performance gains of the proposed scheme for both progressive and non-progressive cases.

The remainder of the paper is organized as follows. In section II, we present the general description of the problem considered, the end-to-end distortion, and the framework of our scheme. In section III, we present a detailed description of the reduced trellis algorithm. Then, a sorting algorithm with dependency constraints is given and the JSCC solution is obtained in section IV. The experimental results are provided in section V. Finally, conclusions are drawn in section VI.

II. Joint Source and Channel Coding

1. Problem Formulation

The problem of JSCC for SVC can be described as follows: Given a scalable coded stream, channel coders, and channel constraints (such as target bit rate, packet loss rate, probability of error, and noise level), how can we select appropriate portions of an SVC stream and corresponding channel coding to achieve the best reconstructed video performance?

Considering a sequence of N frames, a substream is a partial SVC stream representing a version of the video sequence. Let $\pi(n,d,q)$ denote an NAL unit, that is, a layer in the SVC stream associated with frame n at spatial resolution d and quality level q (q=0 refers to the base quality). A substream is a set of NAL units and can be represented by $\boldsymbol{\pi} = (\boldsymbol{\pi}_n)_{n=1,2,\ldots,N}$, where $\boldsymbol{\pi}_n = \{\pi(n,d,q)\}$, containing the NAL units associated with frame n. Example of a substream is illustrated in Fig. 1. In this study, we focus on single resolution SVC stream and replace $\pi(n,d,q)$ with $\pi(n,q)$. For the multiresolution case, our discussion can be applied with the constraint that all lower resolution NAL units are included before a higher resolution one. For a given SVC stream, since $\boldsymbol{\pi}_n = \{\pi(n,q)\}$ is completely determined by l_n, its total length in bytes, a substream $\boldsymbol{\pi}$ can be uniquely identified by a vector defined by $\boldsymbol{l} = (l_n)_{n=1,2,\ldots,N}$.

For ideal channels, a substream can be simply transmitted without error, and the best reconstructed performance problem is to minimize the sum of distortion contributed from all frames

$$\min_{(l,\ldots l_N)\in\mathbf{L}} \sum_{n=1}^{N} D(l_n), \qquad \text{s.t.} \sum_{n=1}^{N} l_n \le L_T, \tag{1}$$

where $D(l_n)$ is the global distortion contributed by $\boldsymbol{\pi}_n$ when the substream $\boldsymbol{l}$ is decoded, $\mathbf{L}$ is the set of all possible substreams,

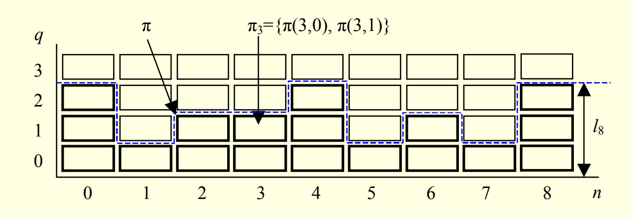


Fig. 1. Example of substream.

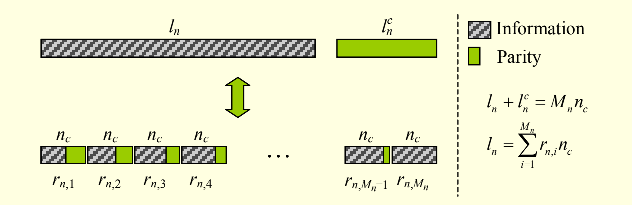


Fig. 2. Codewords associated with frame.

and L_T is the total bytes corresponding to the target bit rate.

For error-prone channels, some error-control measures have to be taken for the transmission of the coded streams. The expected distortion of reconstructed video at the receiver-end from the original video, called end-to-end distortion, is a function of both quantization noise and channel errors. The best reconstruction performance problem becomes

$$\min_{\substack{(l,\ldots,l_N)\in\mathbf{L} \\ (c,\ldots,c_N)}} \sum_{n=1}^{N} \bar{D}(l_n, c_n), \qquad \text{s.t.} \sum_{n=1}^{N} (l_n + l_n^c) \le L_T, \tag{2}$$

where c_n is the channel coding for $\boldsymbol{\pi}_n$, $\bar{D}(l_n, c_n)$ is the global distortion of reconstructed video, and l_n^c is the equivalent parity length in bytes introduced by c_n.

Suppose that the channel coders output codewords of a fixed-length denoted by n_c at rates of $r_{n,1}$, $r_{n,2}$, ... , respectively. As illustrated in Fig. 2, we have the equivalence

$$(l_n, c_n) \Leftrightarrow \mathbf{r}_n = (r_{n,1}, r_{n,2}, \cdots, r_{n,M_n}), \tag{3}$$

where M_n is the codeword number for $\boldsymbol{\pi}_n$. It can be deduced that $l_n = \sum_{i=1}^{M_n} r_{n,i} n_c$ and $l_n + l_n^c = M_n n_c$. Therefore, the JSCC problem can be stated as

$$\min_{\substack{\mathbf{r}_1,\ldots,\mathbf{r}_N \\ M_1,\ldots,M_N}} \sum_{n=1}^{N} \bar{D}(\mathbf{r}_n), \qquad \text{s.t.} \sum_{n=1}^{N} M_n n_c \le L_T. \tag{4}$$

2. Expected End-to-End Distortion

In the decoding of an SVC stream, missed refinement NAL units cause propagation of error to all frames in the predication hierarchy. The distortion of a frame depends on both the NAL units of itself and the refinement units of its references.

When the references are available, the SVC decoder can decode all NAL units prior to the one containing uncorrectable errors, and the expected end-to-end distortion for frame n of M_n codewords at rates $r_{n,1}, r_{n,2}, \cdots, r_{n,M_n}$, respectively is given by

$$\begin{aligned} \bar{D}(\mathbf{r}_n) &= \bar{D}(r_{n,1}, r_{n,2}, \cdots, r_{n,M_n}) \\ &= \sum_{i=1}^{M_n-1} \left\{ D(l_{n,i}) \prod_{j=1}^{i} p_c(\varepsilon, r_{n,j}) \left[1 - p_c(\varepsilon, r_{n,i+1})\right] \right\} \\ &\quad + D(l_{n,M_n}) \prod_{j=1}^{M_n} p_c(\varepsilon, r_{n,j}), \end{aligned} \tag{5}$$

where, $l_{n,i} = \sum_{j=1}^{i} r_{n,j} n_c$, and $p_c(\varepsilon, r)$ denote the probability that there is no uncorrected error after channel-decoding in a codeword at rate r under channel condition ε.

By defining the distortion changes $\Delta D_{n,1} = D(l_{n,1})$ and $\Delta D_{n,i} = D(l_{n,i}) - D(l_{n,i-1})$ for $1 < i \le M_n$, (5) can be calculated recursively as follows [13]:

1) Initialization: $P_c^0 = 1$, $\bar{D}(\varnothing) = 0$, where $\varnothing$ is the notion of void argument.

2) For $1 \le i \le M_n$: $P_c^i = P_c^{i-1} p_c(\varepsilon, r_{n,i})$, and
$\bar{D}(r_{n,1}, \cdots, r_{n,i}) = \bar{D}(r_{n,1}, \cdots, r_{n,i-1}) + \Delta D_n^i P_c^i$.

For each refinement layer q, the source rate distortion information $\left(\Delta l_n^q, \Delta D(l_n^q)\right)$ can be found as proposed in [15], where Δl_n^q and $\Delta D(l_n^q)$ represents the length of layer q and its corresponding distortion change. Then, $\left(\Delta l_{n,i}, \Delta D(l_{n,i})\right)$ for $1 \le i \le M_n$ can be derived from $\left(\Delta l_n^q, \Delta D(l_n^q)\right)$ by interpolation techniques [13].

3. Solution

In principle, a solution to (4) can be found using a nonlinear optimization scheme. In order to converge to a solution, evaluation of the cost function on various combinations of parameters is necessary. The computational burden of this optimization is unmanageable in practice.

To solve (4) efficiently, [13] proposed a scheme employing a Lagrangian multiplier method,

$$\min_{\substack{\mathbf{r}_1,\ldots,\mathbf{r}_N \\ M_1,\ldots,M_N}} \left\{ \sum_{n=1}^{N} \bar{D}(\mathbf{r}_n) + \lambda n_c \sum_{n=1}^{N} M_n \right\}, \tag{6}$$

where $\bar{D}(\mathbf{r}_n)$ is computed by (5). Given a positive λ, $\mathbf{r}_n$ and M_n are obtained by minimizing the term $\left\{\bar{D}(\mathbf{r}_n) + \lambda n_c M_n\right\}$ of frame n independently. If the value of $\sum_{n=1}^{N} M_n n_c$ happens to equal L_T, a desired solution is found or a bisection is applied to determine the λ which makes $\sum_{n=1}^{N} M_n n_c$ approach L_T.

By processing each frame independently, this scheme ignores the dependency between frames and leads to distortion degradation. Proper measurement has to be taken into account for the dependency constraint. As a substitution, we propose a new scheme using a joint two-dimensional optimization which proceeds in three steps.

First, for a given channel condition ε, the optimal end-to-end R-D points are computed for each frame with a new algorithm which is much more efficient than the one from [13]. Here, the end-to-end R-D values refer to the joint source/channel rate and the expected reconstructed distortion at the receiver end.

Next, the end-to-end R-D points for all frames are sorted to construct an R-D curve of the transmission system under the dependency constraint, which guarantees that any corresponding substream contains frames with available references.

Finally, the R-D curve of the system is truncated to meet the designated target rate. The associated NAL units with their channel coding in the truncated portion form the JSCC solution. In the progressive refinement mode, fine grain scalability (FGS), the NAL unit on the truncating point may be truncated accordingly.

The first step is performed on scalable quality layers in each frame in a rate-distortion sense while the second step is performed on the temporal hierarchical structure among frames. The final JSCC solution is the result of joint optimization in two dimensions.

III. Computation of End-to-End R-D Points

One possible way to compute the end-to-end R-D points of a frame is the exhaustive search by enumerating all possible $\mathbf{r}_n$ and M_n, and find out the optimal $\mathbf{r}_n^*$ and minima $\bar{D}_n(\mathbf{r}_n^*)$. Generally, the search space is too large even with some practical constraints. Based on the Markovian condition in the scalable stream data, [13] proposed a simplified Viterbi-based algorithm (VA) and reduced much of the computation complexity.

From [7], we can see the optimal protection level decreases along the embedded stream of a frame. Three protection levels are usually enough to provide the most protection gains for typical channel conditions. These conditions imply that in our case the components of any $\mathbf{r}_n$ satisfy that $r_{n,1} \le r_{n,2} \le \cdots \le r_{n,M_n}$, and the rate sequence $\{r_{n,1}, \cdots, r_{n,M_n}\}$ appears in a few flat segments.

With the above, we impose a reasonable constraint that the same protection level is used for all codewords in an NAL unit, that is, a quality layer. Our experimental results indicate that this constraint causes negligible loss to the JSCC optimization.

1. Description of Trellis Structure

Let R be the set of all possible code rates and d be the cardinality of R. The codeword count is used as the state variable for the trellis structure. Figure 3(a) illustrates the relations among the state arrays, nodes, and branches. Given ΔD_n^q and Δl_n^q as the distortion changes and length increments of K progressive refinement layers of frame n, the related terms are described as follows.

Node. A node $n_i^q(\mathbf{r}_n^q, d_n^q)$ represents a valid point at state i in stage q, where $\mathbf{r}_n^q$ is a path towards stage q, and $d_n^q = \bar{D}(\mathbf{r}_n^q)$ is the end-to-end distortion through path $\mathbf{r}^q$.

Branch. A branch $b_{i,i+j}^q(r_{n,q})$ stems from node $n_i^{q-1}(...)$, ends at node $n_{i+j}^q(...)$, and produces j codewords by using rate $r_{n,q} \in R$ for the (q–1)th layer, where $j = \lceil \Delta l_n^{q-1} / (r_{n,q} n_c) \rceil$. The operation $\lceil v \rceil$ returns the least integer not less than v.

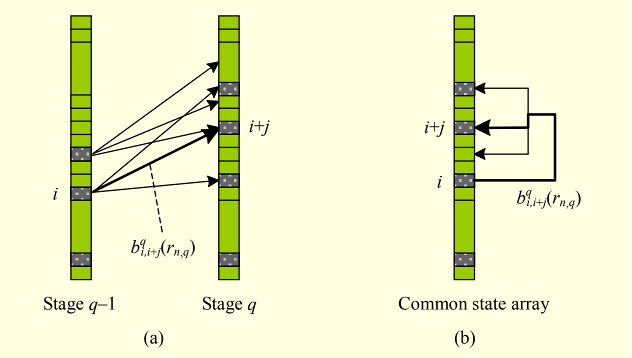


Fig. 3. Trellis structures using codeword count as the state variable: (a) basic trellis structure and (b) reduced trellis structure.

Path. A path $\mathbf{r}_n^q = (r_{n,1}, \cdots, r_{n,q})$ starts from stage 0 and extends to a valid node in stage q, passing a sequence of branches with rates $r_{n,1}, r_{n,2}, \cdots, r_{n,q}$, respectively.

2. Reduced Trellis Algorithm

For the R-D information computation purpose, we further reduce the state arrays of the trellis from K to 1 and make all stages share a common state array in the iteration. Denote J as the common set of all valid nodes. The proposed algorithm is called the reduced trellis algorithm (RTA) and is described as follows.

```
Reduced Trellis Algorithm (RTA)
Initialization: ($q$=0)
    $J = \{n_0^0(\varnothing, 0)\}$, $P_c^0 = 1$   ($\varnothing$ denotes an empty path).
For $q$ from 1 to $K$:
    Step 1. For each $n_i^{q-1}(\mathbf{r}_n^{q-1}, d_n^{q-1}) \in J$ and $r_{n,q} \in R$, build a
        branch $b_{i,i+j}^q(r_{n,q})$ with $j = \lceil \Delta l_n^{q-1} / (r_{n,q} n_c) \rceil$, extend the
        path to $\mathbf{r}_n^q = (\mathbf{r}_n^{q-1}, r_{n,q})$, and calculate
            $P_c^q = P_c^{q-1} [p_c(\varepsilon, r_{n,q})]^j$,
            $d_n^q = d_n^{q-1} + \Delta D_n^{q-1} P_c^q$.
    Step 2. If the destination node was valid and the new $d_n^q$
        is greater than the old, ignore the new one or select the
        new path as a survivor and store (or renew) the
        destination node.
Post processing:
    Check with the valid nodes in $J$ along their state values in
    increasing order. If the distortion of a node is greater than
    or equal to the distortion of a node at lower state, delete
    the node at higher state from $J$.
Results:
    The nodes in $J$ are the end-to-end R-D points.
```

3. Analysis of Complexity

Now, we analyze the complexity of our proposed algorithm. Under the condition of $r_{n,1} \le r_{n,2} \cdots \le r_{n,K}$ and d possible rates, the numbers of branches that need to be built for stage 1,2,…,K, are at most given by d, (1+2+⋯+d),…,(1+2+⋯+d). So the total number of branches to compute for a frame is

$$\begin{aligned} C_b &\le d + (K-1)(1+2+\cdots+d) \\ &\le (K-1)(d+1)d/2+d, \end{aligned} \tag{7}$$

which is globally of order $O(d^2K/2)$. Due to the reduction of state arrays in our algorithm, some nodes may be overwritten because all stages share a common state array. So the right hand of (7) gives the figure of the worst case.

For the Viterbi algorithm from [13], the total number of branches to compute is given by

$$C_b = (M_n - 1)(d+1)d/2+d, \tag{8}$$

where M_n is the total number of codewords used to transmit a frame. As M_n is always much bigger than K, our algorithm is much more efficient. The experimental results indicate that an improvement of 20 to 70 times over the algorithm from [13] is typically achieved.

On the other hand, the reduction of state arrays in our algorithm saves memory space too. The number of states in our algorithm is M_n and the number of valid nodes to store is often less than M_n, while in the algorithm of [13] the number of states to store is d×M_n.

IV. Two-Dimensional Optimization

The distortion of a frame depends on the distortion of its parents, that is, the frames from which it is predicted. In the particular case of scalable H.264/AVC, the temporal structure is a hierarchical decomposition and represents the temporal dependency constraints. Let $(f_0, f_1, \cdots, f_{N_{\mathrm{GOP}}})$ represent the N_{GOP} frames in a group of pictures (GOP) and the proceeding key frame, as shown in Fig. 4. Being bidirectionally predicated by f_0 and f_4, f_2 depends on f_0 and f_4.

In order to take into account the dependency-constraints among frames, we associate each frame $f_n(q)$ (at quality-level q) with a parent-set $\Lambda_n(q) = \{f_{n_1}(q), f_{n_2}(q)\}$, where f_{n_1} and f_{n_2} are parents of f_n. The parent-set of a key frame is an empty set.

It is possible to sort the end-to-end R-D points of all frames under the dependency constraints according to their R-D slopes. Let Ω represent the set of end-to-end R-D points of all frames, and Π represent the sorted queue of Ω. The optimal sorting can be accomplished by the algorithm described as follows, where function *Location*(x) returns the location of item x in Π or 0 if x is invalid or not found in Π.

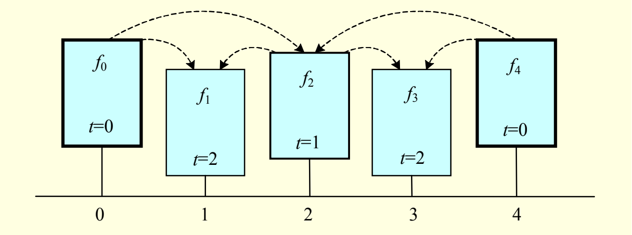


Fig. 4. Dependency in temporal hierarchical structure (N_{GOP}=4).

```
Sorting Algorithm
Initialization: Setup an empty queue, $\Pi$.
Sorting:
  For quality-level $q$ from 0 to $K$–1,
    For temporal structure level from 0 to the deepest one,
      Step 1. Find the points with proper ($n$, $q$) in $\Omega$, and
        denote it by $f_n(q)$.
      Step 2. Calculate $i_0 = \max(i_1, i_2, i_3) + 1$, where
          $i_1$= Location($f_{n_1}(q)$),
          $i_2$= Location($f_{n_2}(q)$),
          $i_3$= Location($f_n(q-1)$).
      Step 3. Calculate the R-D slope of $f_n(q)$ and denote it
        by $k_n(q)$.
      Step 4. Search $\Pi$ from location $i_0$ to the end for the
        first item with R-D slope smaller than $k_n(q)$. If
        found, insert $f_n(q)$ before the item or append it at
        the end of $\Pi$.
```

Once R-D points are sorted into a queue, an R-D curve of the transmission system is constructed, and any portion of the queue (or curve) starting from the beginning gives a substream of the video sequence with a best R-D result. Let $(\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N)$ be the JSCC solution for an optimal substream which meets the target rate. The JSCC solution can be obtained by an algorithm described as follows.

```
Optimal Extraction Algorithm
Initialization:
  $L_{\mathrm{used}}$=0 and $(\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N) = (\varnothing, \varnothing, \cdots, \varnothing)$
Extraction:
  For $i$ from 1 to the end,
    Step 1. Take the point in $\Pi$ at location $i$.
    Step 2. $L_{\mathrm{used}} = L_{\mathrm{used}}$ +length_of_the_point.
    Step 3. If( $L_{\mathrm{used}} \le L_T$ ), then
              $n$ = frame number of the point
              $\mathbf{r}_n$= the path of the point
          else exit.
Results: $(\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N)$
```

Obtained from the R-D curve of the transmission system, the above JSCC solution is optimal in terms of overall R-D performance.

V. Experimental Results

In this section, we first evaluate the performance of our proposed two-dimensional optimized JSCC scheme for scalable H.264/AVC in progressive refinement mode (FSG) and compare it to the existing scheme proposed in [13].

For comparison, the experiments are performed as in [13] on test sequences of "Bus," "Football," "Mobile," and "Foreman" using version 7.10 of the JSVM. Each sequence is encoded into a base layer and three FGS quality enhancement layers with the same encoder settings. To verify the consistency, the encoded streams are extracted at various target rates with the bit stream extractor provided by JSVM. All statistics for the extraction of streams exactly match the data given by [13]. The rate-distortion points of each SVC stream are computed with the method proposed by [15] and tabulated for use in our algorithm.

Transmission over packet loss channels is considered with parameters (packet loss rate) $\varepsilon = 5\%$ and $\varepsilon = 10\%$. We employ the same channel coding/decoding structure as in [13]. Specifically, punctured regular (3, 6) low-density parity-check (LDPC) codes producing codewords of fixed-length of 256 bytes are used. The decoders adopt an iterative algorithm based on belief propagation with maximum 100 iterations. Error performance for these LDPC codes are simulated over two packet loss channels and the statistics are similar to the results of [13]. For fair comparison purpose, the data provided by [13] are used.

To measure the performance, the mean squared error (MSE) between reconstructed and original sequences is computed over whole sequence in Y, U, and V components. Then, the peak signal to noise ratio (PSNR) is obtained as

$$PSNR_{\mathrm{X}} = 10\log_{10}[255^2/MSE_{\mathrm{X}}], \tag{9}$$

where X is Y, U, or V, respectively. The averaged PSNR of all components is computed as

$$PSNR_A = \frac{1}{6}\left(PSNR_Y \times 4 + PSNR_U + PSNR_V\right). \tag{10}$$

For each optimized substream, the transmission is repeated 300 times over the probabilistic packet loss channels. Then, the PSNR values are averaged over all transmissions resulting in $\overline{PSNR_Y}$, $\overline{PSNR_U}$, $\overline{PSNR_V}$, and $\overline{PSNR_A}$, respectively.

Table 1 shows our proposed JSCC scheme compared to the scheme from [13] with the same five code rates over packet loss channels at rates of 5% and 10%. The target rates are basically 500 kbps, 1000 kbps, and 1500 kbps with some exceptions to avoid low PSNR results. As demonstrated, the proposed scheme outperforms the scheme from [13] by a maximum of 0.58 dB and an average of 0.26 dB. The gain of the proposed scheme is mainly due to the temporal dependency constraints. On the other hand, the scheme from [13] may violate the constraints and cause unexpected distortion propagation.

Table 1. Performance comparison of our proposed JSCC scheme to scheme from [13] for transmission over 5% and 10% packet loss channels (progressive refinement mode).

Sequences	Packet loss rate	Target rate (kbps)	Proposed scheme		Scheme from [13]	
			Rate met (kbps)	$\overline{PSNR_A}$ (dB)	Rate met (kbps)	$\overline{PSNR_A}$ (dB)
Bus	5%	500	500.1	32.16	500.1	32.02
		1,000	1,000.2	34.73	1,000.2	34.45
		1,500	1,500.4	36.40	1,500.4	35.95
	10%	500	500.1	32.00	500.1	31.85
		1,000	1,000.2	34.50	1,000.2	34.33
		1,500	1,500.4	36.13	1,499.1	35.64
Football	5%	1,000	1,000.2	32.55	1,000.2	32.30
		1,500	1,500.4	34.13	1,500.4	33.85
		2,000	2,000.5	35.45	2,000.5	35.09
	10%	1,000	1,000.2	32.30	1,000.2	32.05
		1,500	1,500.4	33.89	1,500.4	33.66
		2,000	2,000.5	35.13	2,000.5	34.76
Foreman	5%	500	500.1	37.61	500.1	37.39
		1,000	1,000.2	39.73	1,000.2	39.54
		1,500	1,500.4	41.18	1,500.4	40.71
	10%	500	500.1	37.43	500.1	37.29
		1,000	1,000.2	39.53	1,000.2	39.44
		1,500	1,500.4	40.97	1,500.4	40.46
Mobile	5%	500	500.1	29.53	498.9	29.52
		1,000	1,000.2	32.16	1,000.2	32.11
		1,500	1,500.4	33.75	1,500.4	33.17
	10%	700	700.4	30.52	699.2	30.43
		1,000	1,000.2	31.95	999.0	31.92
		1,500	1,500.4	33.47	1,500.4	33.01

In the following experiments, we evaluate the performance of our proposed JSCC scheme for scalable H.264/AVC in non-progressive refinement mode, medium grain scailability (MSG). The sequences are encoded into a base layer and a quality enhancement layer with basis quantization parameters *QP*=36 and *QP*=24, respectively. The quality enhancement layer is further divided into 5 MGS layers. We compare our scheme with the scheme from [13] over packet loss channels again. The experimental data is shown in Table 2. Target rates of 1,000 kbps, 1500 kbps, and 2,000 kbps are used for three

Table 2. Performance comparison of our proposed JSCC scheme to scheme from [13] for transmission over 5% and 10% packet loss channels (non-progressive refinement mode).

Sequences	Packet loss rate	Target rate (kbps)	Proposed scheme		Scheme from [13]	
			Rate met (kbps)	$\overline{PSNR}_A$ (dB)	Rate met (kbps)	$\overline{PSNR}_A$ (dB)
Bus	5%	1,000	994.1	35.13	1,007.6	34.97
		1,500	1,496.7	36.55	1,505.6	36.34
		2,000	1,998.0	37.92	2,002.3	37.62
	10%	1,000	999.0	34.91	1,006.3	34.77
		1,500	1,499.1	36.30	1,497.8	36.11
		2,000	2,000.5	37.59	2,001.0	37.30
Football	5%	1,000	995.3	33.03	1,019.3	32.98
		1,500	1,493.0	34.16	1,499.1	34.03
		2,000	2,000.5	35.37	1,999.7	35.07
	10%	1,000	1,015.0	32.98	1,068.4	32.98
		1,500	1,488.1	33.98	1,494.0	33.87
		2,000	2,000.5	35.06	2,002.3	34.83
Foreman	5%	500	500.1	37.92	500.6	37.74
		1,000	999.0	40.12	1,001.1	39.98
		1,200	1,200.5	41.50	1,200.3	41.07
	10%	500	500.1	37.78	507.0	37.64
		1,000	1,000.2	39.99	1,001.1	39.66
		1,200	1,200.5	41.09	1,200.3	40.76
Mobile	5%	1,000	996.6	32.32	1,008.9	31.77
		1,500	1,497.9	33.79	1,503.0	33.62
		2,000	1,995.6	35.02	2,006.2	34.67
	10%	1,000	1,000.2	32.18	1,012.8	32.11
		1,500	1,496.7	33.53	1,500.4	33.37
		2,000	1,998.0	34.75	2,001.0	34.41

sequences, while 500 kbps, 1,000 kbps, and 1,200 kbps are used for the foreman sequence. Similar to the results in the FGS mode, our proposed scheme provides gains of 0.55 dB maximum and 0.22 dB on average.

Table 3 summarizes the number of branches to be computed in our RTA and VA from [13]. The number for our RTA is counted during the execution of the algorithm and is 50 given by (7), while the number for VA is given by (8). The factors of improvement are also listed in the table, which shows a significant reduction of complexity with typical factors of 20 to 70.

VI. Conclusion

Table 3. Number of branches to compute in our RTA and VA from [13] and factors of improvement in complexity.

Sequences	Number of branches		Factors of improvement
	Ours	[13]	
Bus	18	986	54.8
Football	18	1,217	67.6
Foreman	19	501	26.4
Mobile	18	1,213	67.4

JSCC for SVC over error-prone channels is very complex due to the complicated hierarchical predication structure and other new features of SVC. To deal with this difficulty, a new two-dimensional optimization JSCC scheme is proposed which employs a novel RTA to compute the end-to-end R-D points efficiently and sorts the R-D points of frames to construct an R-D curve for the transmission system. The proposed RTA adopts a simplified calculation and a reduced trellis structure so that it achieves a significant improvement of 20 to 70 times over the existing VA approach from [13]. As the JSCC solution is obtained in an R-D sense and under the dependency constraints, it is optimal in terms of overall R-D performance. Experiments are carried out for the SVC transmission over packet loss channels with various target rates. Both progressive (FGS) and non-progressive (MGS) modes are considered. The results demonstrate the advantage of our proposed scheme over the existing schemes [13]. An average quality gain of up to 0.26 dB and 0.22 dB are achieved for FGS and MGS modes, respectively.

References

[1] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, 2007, pp. 1103-1120.

[2] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, "Scalable Video Coding," Joint Draft ITU-T Rec. H.264/ISO/IEC 14496-1/ Amd.3 2007.

[3] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, "Advanced Videon Coding," Draft ITU-T Rec. H.264/ISO/IEC 14496-10, 2005.

[4] G. Cheung and A. Zakhor. "Joint Source/Channel Coding of Scalable Video Over Noisy Channels," *Proc. IEEE Int. Conf. Image Process.*, 1996, p. 767.

[5] L. Kondi, F. Ishtiaq, and A. Katsaggelos. "Joint Source-Channel Coding for Motion-Compensated DCT-Based SNR Scalable Video," *IEEE Trans. Image Process.*, vol. 11, no. 9, 2002, pp. 1043-1052.

[6] S. Dumitrescu, X. Wu, and Z. Wang, “Globally Optimal Uneven Error Protected Packetization of Scalable Code Streams,” *IEEE Trans. Multimedia*, vol. 6, no. 2, 2004, pp. 230-239.

[7] Z. Wu, A. Bilgini, and M.W. Marcellin, “Joint Source/Channel Coding for Image Transmission with JPEG2000 Over Memoryless Channels,” *IEEE Trans. Image Process.*, vol. 14, no. 8, 2005, pp. 1020-1032.

[8] M. Bansal and L.P. Kondi, “Scalable Video Transmission Over Rayleigh Fading Channels Using LDPC Codes,” *Image Video Commun. Process. Conf.*, 2005, pp. 390-401.

[9] T. Fang and L.P. Chau, “GOP Based Channel Rate Allocation Using Genetic Algorithm for Scalable Video Streaming over Error-prone Networks,” *IEEE Trans. Image Process.*, vol. 15, no. 6, 2006, pp. 1323-1330.

[10] M.K. Jubran et al, “Optimal Bandwidth Allocation for Scalable H.264 Video Transmission over MIMO Systems,” *Proc. MILCOM,* 2006, pp. 1-7.

[11] A. Naghdinezhad, M.R. Hashemi, and O. Fatemi, “A Novel Adaptive Unequal Error Protection Method for Scalable Video over Wireless Networks,” *IEEE Int. Symp. Consumer Electron.*, 2007, pp. 1-6.

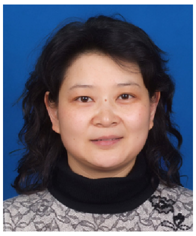
[12] Y. Liu et al, “On Unequal Error Protection with Low Density Parity Check Codes in Scalable Video Coding,” *CISS,* 2009, pp. 793-798.

[13] M. Stouf et al., “Scalable Joint Source-Channel Coding for the Scalable Extension of H.264/AVC,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 12, 2008, pp. 1657-1670.

[14] E. Maani and A.K. Katsaggelos, “Two-Dimensional Channel Coding for Scalable H.264/AVC Video,” *Picture Coding Symp.*, 2008, pp. 1-4.

[15] I. Amonou et al., “Optimized Rate-Distortion Extraction with Quality Layers in the Scalable Extension of H.264/AVC,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, 2007, pp. 1186-1193.

Xiao-Feng Li received the BS in information theory from Xidian University, Xian, China, in 1984, and the MS in communication engineering from University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 1987. Since 1987, he has been with the UESTC, where he is a professor at the School of Communication and Information Engineering. His research interests are in the areas of wireless communication systems, multimedia communications, image and video coding, signal processing, and DSP applications.



Ning Zhou received the BS in radio techniques from UESTC, in 1988, and the MS and PhD in communication engineering from UESTC in 1991 and 2009, respectively. Since 1991, she has been with the UESTC, where she is an associate professor at the School of Communication and Information Engineering. Her research interests include image processing, video coding, and multimedia communications.

Hong-Sheng Liu received the BS in radio techniques from Chongqing University, in 1989, and the MS and PhD in communication engineering from UESTC in 1995 and 2009, respectively. Since 1995, he has been with the UESTC, where he is an associate professor at the School of Communication and Information Engineering. His research interests include wireless video communication, space-time signal processing, and array signal processing.