H.264/AVCのためのNALにおけるストリーム認証方式

上田 真太郎† 重野 寛‡ 岡田 謙一‡

† 慶應義塾大学大学院理工学研究科 ‡ 慶應義塾大学理工学部

概要: 本稿ではH.264/AVCのためのNALにおけるストリーム認証方式を提案する．H.264/AVCは高圧縮や伝送システムに対して汎用性を持つ動画像符号化方式である．しかし送信者の真正性やデータの完全性を保証する認証機能は定義されていない．既存のストリーム認証方式はパケットレベルで処理を行うためH.264/AVCが提供する汎用性を損なう．提案方式のプロトタイプを実装し，実際に動画再生されるデータの割合を表す実効率とオーバヘッドの2項目において比較評価を行った．実効率において既存方式に比べ約38%向上することを示した．

A Stream Authentication Scheme at the NAL for H.264/AVC

Shintaro UEDA†, Hiroshi SHIGENO‡, and Ken-ichi OKADA‡

† Graduate School of Science and Technology, Keio University

‡ Faculty of Science and Technology, Keio University

{ueda,shigeno,okada}@mos.ics.keio.ac.jp

Abstract: We propose a stream authentication scheme which carries out authentication procedures at the NAL. H.264/AVC offers high compression and flexible mapping to transport layers. However it does not offer security features, such as data integrity and sender authentication. Existing stream authentication schemes carry out authentication at the packet level and take away the flexibility of H.264/AVC. We implemented a prototype of our authentication scheme and carried out comparative evaluations.

1 はじめに

H.264/AVC[1][2]は高品質の動画像を用いた幅広いマルチメディアアプリケーションに使用できる最先端の動画像符号化方式である．H.264/AVCは以前の符号化方式であるMPEG-2やMPEG-4の圧縮率を高めることを目的としている．

H.264/AVCにはセキュリティ機能として，重要データの再送や画素のコンシールメントなどによるエラー耐性機能が規定されている[3][4]．しかしネットワークでストリーミング伝送する際にセキュリティ上必要になる，送信者の真正性やデータの完全性を保証するための認証機能は規定されていない．安全なストリーミング環境設定のためには，H.264/AVCに上記のようなストリーム認証機能を持たせる必要がある．

ストリーム認証ではストリームの伝送の進行に同期して継続的な認証を行う必要がある．パケットネットワークにおいてはパケットロスによる影響を考慮する必要がある．特にリアルタイム転送の際にはUDPのようなベストエフォートサービスが用いられ，パケットロスが頻繁に見られる．そのため，パケットロスの対応のために1パケットごとに認証を行う必要がある．しかし，全てのパケットに対して演算負荷の高いディジタル署名を施すのは効率が悪い．さらにH.264/AVCは伝送フォーマットとしてRTPやMPEG-2システムなど，複数規定している．そのため，パケット単位での認証を考えるのではなく，伝送フォーマットに依存しない認証方式が必要となる．

また，H.264/AVCデータは画像データのみではなく様々なパラメータや冗長データなどから成り立っている．そのため，データ間に依存関係が存在し，データごとの重要度が異なる．よって，重要度の高いデータほどエラー耐性を持たせる必要がある．このような方式として重要度の高いデータを保護するストリーム認証方式SAVe[5]が提案されているが，他の既存のストリーム認証方式同様H.264/AVCデータ構造を考慮しておらず，またパケット単位での認証しか行うことができない．

H.264/AVCデータ構造を考慮し，伝送システムに依存しないストリーム認証方式を提案する．提案方式において，認証処理は既存方式のようなパケットレベルではなく，動画像符号化処理と伝送システムへのマッピングの間にあるNALで行われる．本方式を適用することにより，効率的かつ汎用的なストリーム認証を可能とすることを目指す．

以下，本稿では，第2章においてH.264/AVCについて述べる．次に，第3章において関連研究について解説する．さらに，第4章において提案方式について述べる．第5章において評価について述べ，最後に第6章を結論とする．

2 H.264/AVC

本章では，対象動画像圧縮符号化方式であるH.264/AVCについて概説する．

H.264/AVC とは ITU-T において H.264, ISO/IEC において MPEG-4 part10 AVC として標準化された動画像圧縮符号化方式であり，一般的に H.264/AVC として呼称されており，本稿でもこれにならう．H.264/AVC は放送・蓄積・通信などの幅広いマルチメディアアプリケーション分野で汎用的に用いられることを想定したジェネリックコーディング方式である．また，特に高圧縮化に重点を置いており，従来の動画像符号化方式の 2 倍以上の圧縮率となっている．これは多くの新しい符号化ツールによって実現されている．

2.1 Network Abstraction Layer

H.264/AVC の特徴の 1 つとして，Video Coding Layer(VCL) と Network Abstraction Layer(NAL) に分離されていることが挙げられる．VCL では動画像符号化処理が行われ，NAL では VCL からのデータを伝送システムに対しマッピングが行われる．VCL から NAL にピクチャのスライスやピクチャのヘッダ情報に相当するパラメータセット等のデータが渡され，NAL ユニットという単位で管理される．その NAL ユニットを単位として RTP や MPEG-2 システム等の下位システムへのマッピングを行うことにより，汎用的な伝送・蓄積を実現することが可能となる．

NAL ユニットは 1 バイトの NAL ヘッダと，可変サイズの動画像圧縮された生データである Raw Byte Sequence Payload(RBSP) から構成されている．RBSP にピクチャのスライスやパラメータセット等のデータが格納される．NAL ヘッダには NAL ユニットの種類を示す情報が含まれる．

2.2 NAL ユニットの種類

現時点で定義されている NAL ユニットは 12 種類である [3]．NAL ユニットタイプが 1-5，12 の NAL ユニットは動画像のデータであり，VCL NAL ユニットと呼称される．それ以外の NAL ユニットは非 VCL NAL ユニットと呼ばれ，符号化を制御するためのパラメータや補足的な付加情報等が存在する．NAL ユニットの中でも特に重要と考えられる IDR ピクチャ，SPS，PPS について説明する．

Instantaneous Decoding Refresh(IDR) ピクチャとは画像シーケンスの先頭となるピクチャである．デコーダにおいて IDR ピクチャが受信されると，デコードに必要な全ての状態がリフレッシュされ，そこから新たなシーケンスが始まる．

Sequence Parameter Set(SPS) とはシーケンス全体の符号化にかかわる情報が含まれたヘッダ情報に相当するパラメータセットである．また，Picture Parameter Set(PPS) とはピクチャ全体の符号化にかかわる情報が含まれたヘッダ情報に相当するパラメータセットである．

H.264/AVC は 1 つのビットストリームの中で，複数のシーケンスを扱うことができ，1 つのシーケンスの中に複数のピクチャが含まれている．よって，シーケンスやピクチャを識別するために SPS と PPS には番号がふられている．また，PPS の中で SPS の番号を指定することにより，ピクチャがどのシーケンスに属しているかの識別を行う．さらに，ピクチャのスライスデータ (Coded Slice) にはスライスヘッダが付いており，その中で PPS の番号を指定することにより，スライスがどのピクチャに属しているかの識別を行う．PPS 番号，SPS 番号を辿ることにより，スライスがどのピクチャやシーケンスに属しているかを識別している．

また，パラメータセットとスライスの順序における制約は，パラメータセットを参照するデータよりも先にデコーダに届いていれば良いという，非常に柔軟なものとなっている．図 1 に各パラメータセットとスライスの依存関係を示す．

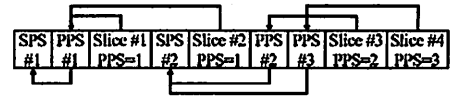

図 1: パラメータセットとスライスの関係

3 関連研究

ストリーミング伝送を行う際に，ストリーム認証を効率化する技術について，いくつかの提案がなされている．

Gennaro らの Chain 方式 [6] では，各パケットが 1 つ後のパケットのハッシュ値を持ち，最初のパケットのみに署名を施す．よって，この方式では，全てのパケットが揃わない限り送信側で署名演算が行えないため，リアルタイム送信を行う際には，複数のパケットをブロックに区切って署名を行う必要がある．一般にストリーミング伝送はリアルタイム性を重視するため，パケットの再送を行わない UDP を使って送信される．Chain 方式では，パケットロスによって署名が連続しない部分が生じると認証が途切れてしまうため，パケットロスに対する耐性がないことが欠点である．

Park らは，誤り訂正符号の 1 つである IDA[7] を利用した SAIDA 方式 [8] を提案している．この方式はまず，各パケットのハッシュ値を連結したもののハッシュ値をとることでグループハッシュを生成

し，これらに対してのみ署名を施す．この後，グループハッシュと署名に対してそれぞれIDAの処理を施すことにより，これらのデータを分散する．各パケットには分散後のそれぞれのデータが付与される．この方式はIDAを利用することで，パケットロスに対する耐性を持たせると共に，各パケットのオーバヘッドを抑えている．

これらの方式に共通することはパケットレベルでの処理を行っていることである．しかし，パケットレベルで処理を行うことによりH.264/AVCの伝送システムに対する汎用性が失われる．また，H.264/AVCには制御データ，動画データといった様々なデータが存在し，データにより重要度が異なる．しかし，既存のパケットレベルでの認証方式では，各パケットにどのようなデータが格納されているかは隠蔽されているため，データの重要度に応じた処理を行うのは困難である．そこで，NALにおいて認証処理を行うことによりH.264/AVC特有のデータに対し重要度を設定することが可能となり，データ損失に対する耐性と，伝送システムに汎用性を持つストリーム認証方式が可能となる．

4 提案

本章では，本稿で提案するH.264/AVCのためのNALにおけるストリーム認証方式を解説する．

4.1 提案概要

提案方式の署名・検証処理は全てNALユニットレベルで行われる．これは異なるデータタイプへの重要度の設定，及びH.264/AVCが可能としている伝送システムへの汎用性を損なわないためである．

提案方式では，Coded Slice，IDR，SPS，PPSの4つのNALユニットに焦点をあてる．ただし，本提案を拡張することで他のNALユニットにも対応は可能である．ここで，本提案で想定しているNALユニットの順序と，各NALユニット間の依存関係を図2に示す．図中のSはSPS，PはPPS，IはIDR，CはCoded Slice，実線矢印はNALユニット間の依存関係を表している．

図 2: NALユニットの順序及び依存関係

本提案では既存のストリーム認証方式同様，ディジタル署名とハッシュの組み合わせを用いる．さらに重要度の高いNALユニットをデータ損失に耐性を持たせるためForward Error Correction(FEC)を用いる．SPS，PPS，IDRの制御データが失われた場合は次の制御データまでの全てのNALユニットが影響を受けるため，これらのデータに対してFECを用いる．FECとしては元のデータからn個のFECデータが生成された場合，m個以上のFECデータが受信されれば元のデータの復元が可能といった$(n,n-m)$特性を持ったFECを用いる．つまり$n-m$個のパケットロスに耐えうる．

本提案のストリーム認証方式では，署名処理に先立ち，3つの新たなNALユニットタイプを定義する．定義するNALユニットは以下である．NALユニットタイプ29は動画データのハッシュ値を連結した値，NALユニットタイプ30はディジタル署名，NALユニットタイプ31はFECデータとなっている．図3に提案方式を含んだ符号化レイヤから伝送システムまでのフローを示す．認証による処理をNALにのみ追加していることからH.264/AVCの汎用性を損なっていないことがわかる．

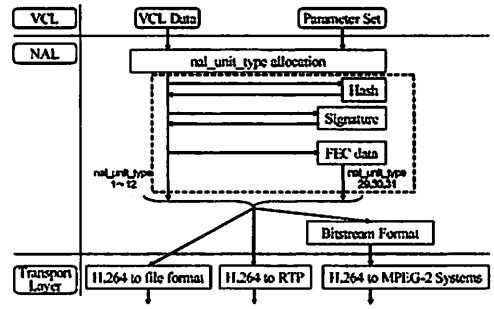

図 3: 署名フロー

4.2 署名処理

本節でエンコーダ(送信側)における署名処理について述べる．提案方式では，NALユニットのストリームをN個のNALユニットのグループに分け，認証グループと呼ぶ．認証グループの最大長をN_{max}とする．

NALユニットのストリームから形成される認証グループは図4に示されている．一般性を損なわずに説明を簡単にするため，N_{max}が5の場合の例とする．

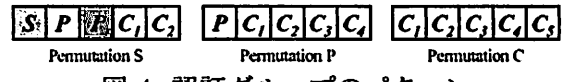

図 4: 認証グループのパターン

パターンSはシーケンスの先頭を示すパターンであり，認証グループの先頭が必ずSPS，PPS，IDRであり，後続する残りのNALユニットがCoded Sliceで構成される．パターンPはPPSが単独で更新された場合を示すパターンであり，認証グルー

プの先頭がPPSであり，残りのNALユニットはCoded Sliceで構成される．パターンCは認証グループの全てがCoded Sliceで構成されるパターンであり，3つのパターンで唯一制御データを含まない．またパターンCは最頻出のパターンである．各パターンにより重要度が異なることから，提案方式では各パターンによって署名処理が異なり以下に各パターンの署名処理について述べる．

まずはパターンSの署名処理を図5に示す．

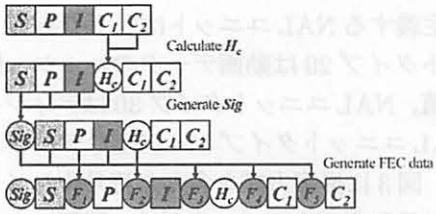

図 5: パターンSの署名処理

動画データである各Cのハッシュ値をとり，それらを連結した値をH_cとする．このH_cはNALユニットタイプ29のNALユニットに格納される．

$$H_c = Hash(C_1) \parallel Hash(C_2) \quad (1)$$

次に，制御データである，S，P，I及び生成されたH_cを連結する．これに対し以下の式で表されるディジタル署名Sigを生成する．

$$Sig = Enc(KEY_s, Hash(S \parallel P \parallel I \parallel H_c)) \quad (2)$$

KEY_sは公開鍵暗号の秘密鍵である．ディジタル署名はNALユニットタイプ30のNALユニットに格納される．

次にSig，S，P，I，H_cを連結し，これに対しFEC処理を施すことによりFECデータ(図中F)を生成する．ここで生成されるFECデータ数nを認証グループサイズNとする．このFECデータはNALユニットタイプ31のNALユニットに格納される．$n > N$の場合，データ損失の際，デコーダ側(受信側)での最大バッファ遅延が大きくなるためnはNに設定される．

パターンPの署名処理はパターンS同様，各Cのハッシュ値を連結しH_cを生成する．このH_cをPと連結し，これに対しSigを生成する．このSig，P，H_cを連結した値に対してFEC処理を施し，FECデータを生成する．生成されるFECデータの数nは認証グループサイズと同数にする．

パターンCでは各Cのハッシュ値を連結しH_cを生成し，これに対してSigを生成する．このSigとH_cを認証グループの先頭に挿入する．パターンCには重要データが存在しないため，FEC処理は施されない．

図6にNALユニットのストリームに対する提案方式の適用例を示す．図6はNALユニットのストリーム，提案方式の処理，最終的に送信されるNALユニットの3段階に分けられている．この例はN_{max}=5の場合を示しているが，全ての認証グループが5NALユニットで形成されていないことがわかる．これは提案方式において認証グループはSPS，PPS，IDRのNALユニットの生成に応じて形成されるためである．

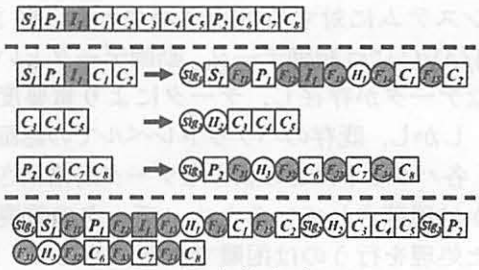
図 6: 提案方式適用例

4.3 検証処理

本節で提案のストリーム認証方式の検証処理について，重要データの損失がない場合とある場合の2つに分けて述べる．前述のようにSPS，PPS，IDRが重要データとなっている．

重要データの損失がない場合は受信したNALユニットと認証データを用いて，通常のディジタル署名を用いた検証を行う．デコーダにおいて，認証データSig，H_c，及び制御データS，P，Iが全て到着次第Sigを用いて検証を行う．続いて認証したH_cを用いて，Cを受信次第検証する．

重要データの損失がある場合はエンコーダで生成されたFECデータを用いる必要がある．FECデータを復元閾値m以上受信した場合，損失したデータを復元することが可能となる．損失データを復元した後は，データ損失がない場合の検証と同様の処理でSigから検証を行う．

5 評価

前章において提案したストリーム認証方式の有効性を実証するため，提案方式のプロトタイプを実装し，H.264/MPEG-4 AVC Reference Software[9]に組み込んだ．実際のネットワークではパケットロス率は時間とともに変動してしまうため，正確な評価を行うことは困難である．そこで，バーストパケットロスモデルとしてtwo-state Markov Chain Loss Modelを用いた仮想的なネットワーク上で評価を行った．

5.1 実装環境

性能測定にはCPUがPentium4 3.4GHz，メインメモリが2.0GBのマシンを用いた．提案方式のプロトタイプの実装の開発言語にはVisual Studio C++ .Netを使用した．プロトタイプをH.264/MPEG-4 AVC Reference Software JM9.6に組み込んだ．またハッシュ関数と公開鍵暗号方式にはそれぞれOpenSSLライブラリ0.9.8aから160bits SHA-1と1024bits RSAを用いた．FECとしてCrypto++ライブラリ5.2.1のInformation Dispersal Algorithm(IDA)を用いた．パケットレベルでの既存方式と比較評価を行うため，伝送システムはRTPとしている．また，JM9.6の仕様によりNALユニットが1RTPパケットに格納される．

5.2 パラメータ

評価に使用するパラメータを以下に示す．

最大グループ長N_{max}は5,...,15に設定した．復元閾値Mは3,...,N_{max}に設定した．復元閾値は重要データを復元するために必要なFECデータの個数を表している．インターネットにおけるパケットロスの研究結果[10][11]により，パケットロス率の最大値と平均バースト長をそれぞれ40%と8パケットに設定した．

エンコーダで生成されるフレーム数F_nとエンコーダで1秒間に生成されるフレーム数F_rをそれぞれ900フレームと30フレーム/秒に設定した．F_n/F_rはエンコードされる動画像時間を表す．SPS挿入間隔S_iは2000,...,5000$msec$の範囲内の乱数に設定した．これは，IDRを挿入する間隔が，最低2秒最大5秒程度[12]と考えられているからである．また，パターンPの評価を行うためにPPS挿入間隔P_iを1000,...,2000$msec$の範囲内の乱数に設定した．また，画像サイズはCIFとした．

5.3 評価項目

評価項目は実効率とオーバヘッドの2項目である．実効率は，エンコーダ(送信側)で生成されたNALユニット数に対する，デコーダ(受信側)で認証され，かつ，再生が可能なNALユニット数の比である．既存のストリーム認証方式のパケットロスに対する耐性の評価は認証率(送信側から送信されたパケット数に対する，受信側で認証されたパケット数の比)が用いられている．しかし，動画像のように依存関係があるデータを扱う際，認証されたデータ全てが動画生成できるとは限らない．よって認証率が必ずしも実効率に等しくはならない．実効率の方がより重要な値となる．

オーバヘッドは提案方式を用いて認証を行うことにより増加したデータサイズの割合を表す．実際は生成NALユニットサイズに対する，認証情報NALユニットサイズの比である．認証情報NALユニットとは，H_c, Sig, FEC等の提案方式によってH.264/AVCエンコーダに追加されるデータである．生成NALユニットとは，Coded Slice, IDR, SPS, PPS等のH.264/AVCエンコーダで生成されるデータである．

提案方式と比較する既存方式として，誤り訂正技術を使用しているSAIDA方式を用いる．

5.4 評価結果

5.4.1 実効率

図7にN_{max}=9とM=5,9の際のパケットロス率と実効率の関係を示す．

M=5とM=9の両場合とも提案方式の実効率がSAIDA方式より高い値となっている．例えばパケットロス率20%，M=5の場合では提案方式とSAIDA方式の実効率はそれぞれ0.65と0.47となっている．SAIDA方式を基準にした場合，提案方式の実効率は約38%向上しており，提案方式の有効性が示されている．提案方式は重要データである制御データを復元しているが，SAIDA方式には制御データを復元する機能が存在しない．SAIDA方式では認証データのみがパケットロスに対する耐性を持たされている．したがって，制御データがロスした場合，その制御データに依存している後続の動画データの認証が可能であるが，再生が不可能となる．これが2つの方式間の実効率の大きな差の起因である．

また，提案方式は認証グループを制御データの生成に応じて形成しているが，SAIDA方式は常にN_{max}の値で認証グループを形成している．例えばSAIDA方式によって形成された認証グループがC,C,C,C,S,P,Iであるとする．Sは新たなシーケンスを示すので，この認証グループは2つのシーケンスから成っている．よって，前半部のCが多量にロスした場合，後半部の制御データの認証が不可能となる場合がある．これにより，後半部の制御データに依存する後続の動画データが再生されない状況が生じる．これに対し，提案方式では制御データは必ず認証グループの先頭に配置され，認証グループ内にその制御データに依存しない動画データは存在しない．そのため，依存関係のないNALユニットから制御データの認証が不可能となるような非効率なことは生じない．

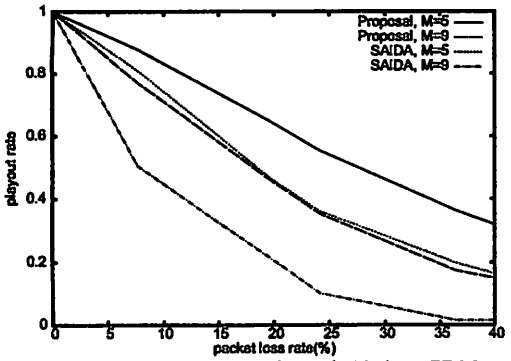

図 7: パケットロス率と実効率の関係

5.4.2 オーバヘッド

図 8 にオーバヘッドと実効率の関係を示す.

提案方式の実効率は SAIDA 方式より高い値となっているが，オーバヘッドも SAIDA に比べ高い値となっている．実効率が高くなる理由は前章で述べた通りである．オーバヘッドに関しては，SAIDA 方式に比べ提案方式は約 10 倍の値を示している．提案方式において，NAL ユニットの中で最もデータサイズの大きい動画データを含む IDR に対して FEC 処理を行っているためである．それに対して，SAIDA 方式はハッシュ及びディジタル署名に対してのみ FEC 処理を施すため，オーバヘッドが数十バイトに抑えられる．SAIDA 方式に比べて提案方式のオーバヘッドが大きいが今回用いた画像サイズにおいて H.264/AVC ビットストリームに対しては約 10% 以下に抑えられている．また提案方式のパターン P で生成される FEC のデータサイズは SAIDA 方式で生成される FEC のデータサイズと同等であり，パターン S 時の FEC のみがオーバヘッドが増加する原因となっている．

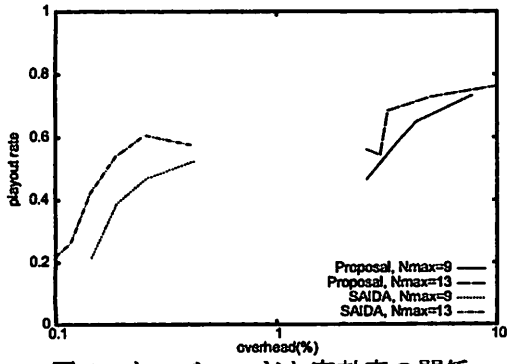

図 8: オーバヘッドと実効率の関係

6 おわりに

本稿では H.264/AVC のためのストリーム認証方式を提案した．既存の認証方式とは異なり，提案方式では NAL において認証処理を行う．これにより H.264/AVC 特有のデータに対し規格を阻害せずに重要度の設定を可能とする．提案方式のプロトタイプを実装し H.264/MPEG-4 AVC Reference Software に組み込み，実効率とオーバヘッドの 2 項目において比較評価を行った．提案方式が既存の方式に比べ実効率を約 38%向上することを示した．

謝辞

本研究は日本学術振興会科学研究費補助金によって行われた．関係者各位に深謝する．

参考文献

[1] ITU-T Recommendation H.264. Advanced Video Coding for generic audiovisual services, May 2003.

[2] ISO/IEC International Standard 14496-10:2003.

[3] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Lutra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 13, No. 7, pp. 560-576, July 2003.

[4] S. Wenger, "H.264/AVC Over IP," *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 13, No. 7, pp. 645-656, July 2003.

[5] S. Ueda, S.Kaneko, N. Kawaguchi, H. Shigeno, and K. Okada, "A Real-Time Stream Authentication Scheme for Video Streams," *IPSJ Journal*, Vol. 47, No. 2, pp.415-425, February 2006.

[6] R. Gennaro and P. Rohatgi, "How to Sign Digital Streams," *CRYPTO* 1997, LNCS1294, pp. 180-197, 1997.

[7] M. Rabin, "Efficient Dispersal of Information for Security, Load balancing, and Fault Tolerance," *Journal of The ACM*, 2:335-348, 1989.

[8] J. Park, E. Chong, and H. Siegel, "Efficient Multicast Stream Authentication Using Erasure Codes," *ACM Trans. on Information and System Security*, pp. 258-285, May 2003.

[9] http://iphome.hhi.de/suehring/tml/ Nov. 2005.

[10] M. Yajnik, S. Moon, J. Kurose, D. Towsley, "Measurement and modeling of the Temporal Dependence in Packet Loss", In *Proc. of the IEEE Conf. on Computer Comm.*, pp.345-352, 1999.

[11] J. Boyce, R. Gaglianello, "Packet Loss Effects on MPEG Video Sent Over the Public Internet", In *Proc. of the Sixth ACM International Conf. on Multimedia*, pp.181-190, 1998.

[12] 境田慎一，井口和久，合志清一，" 携帯端末向けサービス用 AVC/H.264 エンコーダの開発", NHK R&D, No.93, pp.26-31, 2005 年.