映像とメタデータのウェブ化によるコンテンツ閲覧の質的向上

宮森　恒[†]　　田中 克己[†‡]

†通信総合研究所けいはんな情報通信融合研究センター　〒619-0289 京都府相楽郡精華町光台 3-5
‡京都大学 大学院情報学研究科 社会情報学専攻 〒565-0456 大阪府吹田市河田 4-5-6
E-mail:　†miya@crl.go.jp,　‡ktanaka@i.kyoto-u.ac.jp

あらまし　本稿では，長時間にわたるビデオやそのメタデータの意味内容をウェブを介してさまざまな形式で表示するための意味的変換処理の概念を提案する．従来，ファイルのダウンロードやストリーミングといったウェブによる映像視聴は，時系列データである映像を逐次再生ソフトで通常再生しながら閲覧する必要があり，伝統的なTV 的視聴形態に基づくものであった．一方，サマリーの一覧表示，ダイジェスト生成，特定シーン検索などの研究は活発になされているものの，いずれも個別の機能あるいはコンテンツを限られた範囲に限定して実現されている．我々のアプローチは，メタデータのもつ付加価値性と，ウェブのもつデザインの柔軟性，コンテンツ連携の多様性，インタラクティブ性を利用し，映像閲覧を向上させる機能を一体的にブラウザ上に実現することである．これにより，映像視聴・閲覧の効率化・付加価値化・多視点化を図ることができると考えられる．本稿では，本ウェブ化処理の概念とそのために必要な技術，および，実装したプロトタイプについて述べる．
キーワード　映像，メタデータ，ウェブ化，セマンティック，コンテンツ，閲覧

Enrichment of Content Browsing by Webification of Video and Metadata

Hisashi MIYAMORI[†]　　Katsumi TANAKA[†‡]

†Communications Research Laboratory　3-5 Hikari-dai Seika-cho Souraku-gun, Kyoto, 619-0289 Japan
‡Kyoto University　Yoshida Honmachi, Sakyo, Kyoto, 606-8501 Japan
E-mail:　†miya@crl.go.jp,　‡ktanaka@i.kyoto-u.ac.jp

Abstract　This paper proposes a concept of semantic transformation method of video and its metadata for efficient browsing using web and a user-friendly interface. Conventionally, the typical form of watching video via web, such as downloading, streaming, etc., requires a sequential playback of time-series data of video using certain player applications, meaning that it is based on a traditional viewing form of TV. Researches like summary listing, digest generation, specific scene retrieval, etc. have been conducted actively. However, these researches have been studied individually and each method has different scope of applications so far. Our approach is utilizing the value-added features provided by metadata, and the flexibility of design, the variety of content collaboration, and the good interactivity, provided by web, to develop functions enhancing video viewing on a browser in a unified manner. Consequently, the proposed method can improve efficiency of video browsing, add value to it, and help view the video by different perspective. This paper describes the concept of the webification, key technologies, and an implemented prototype based on the concept.
Keyword　Video, Metadata, Webification, Semantics, Content, Browsing

1. はじめに

近年，PC の処理速度向上やネットワークの広帯域化，各種メディア伝送に必要なデータ形式・プロトコルの整備に伴い，一般の利用者がウェブ環境を使うことは当たり前のこととなった．

一方，デジタルカメラの普及や放送番組のデジタル化などに伴い，さまざまな分野におけるデジタル映像の利用も増加の一途をたどっている．

現在，ウェブを介した映像の代表的な利用形態には以下の 2 つがある．

- ファイルのダウンロード
- ストリーミング

いずれの利用形態も，時系列データである映像を逐次再生ソフトで通常再生しながら閲覧する点で共通している．また，特に自分の見たい部分を確認するには，早送りや巻戻しなどの一次元的操作を行う必要がある．つまり，これらの方法は，従来のテレビ的視聴方法に基づいた形態であり，視聴するコンテンツが長時間に渡る場合や，コンテンツ数が多い場合に利用者にとって必ずしも効率的な方法とはいえない．

このような問題を解決するために次の 2 つのアプローチによる研究が行なわれている．

1 つめは，映像の全体概要を確認する技術によるアプローチである．映像の全体概要を確認するために，以下の 2 つの技術が研究されている．

(a) 映像をウェブページ上に概観表示する技術

(b) 映像のダイジェストを生成する技術

(a)については，映像セグメントが長くて稀なものの重要度を高く計算し，その値に応じて表示するキーフレームの大小を制御することで，マンガのような表示形態でサマリーを表示する Video Manga[15]が提案されている．また，映像とクローズドキャプションをセグメント，シーン，ショット単位に分割することで構造化し，ズームメタファを利用して各単位での分割結果をスムーズにつなぐことにより，映像とウェブの間をシームレスに移動可能な表示インタフェースを提供する TV2Web[10]が提案されている．関連する研究においては，キー画像の選択方法，レイアウト方法，インタフェース等にそれぞれ特徴がある[2][7][13].

(b)については，手入力された番組索引とルールに基づき重要度判定を行い，個人の嗜好に適応したダイジェストを生成するシステム[6]が提案されている．また，メディア認識技術により得られる動作索引とルールに基づき，ナレーションテキストの生成とそれに対応した重要場面の選択を行うことにより，個人の嗜好に適応したダイジェストを生成するシステム[9]が提案されている．関連する研究においては，ストーリー展開の把握方法，個別の重要シーン選択方法，ダイジェストの生成方法等に特徴がある[1][4][5].

2 つめは，映像の必要部分を検索する技術によるアプローチである．映像全体から必要な一部分を見つけるために，特定シーン検索やその前処理としてのコンテンツ解析技術が研究されている．例えば，顔のクローズアップや人物，屋外シーンなどを画像解析により検出し，これとクローズドキャプションを文法解析して得られるキーセンテンスを DP により関連付け，ニュース映像から特定シーンを発見する手法[11]が提案されている．また，ドメイン知識とメディア認識技術を用いて人物の基本動作を索引付けし，これと一般動作の成立ルールを利用することにより，テニス映像からスマッシュやネットダッシュといった複雑なシーンを検索可能なシステム[8]が提案されている．関連する研究においては，ジャンルを考慮した特徴量の選択方法やその解析方法，検索への利用方法等に特徴がある[3][12][14].

従来，これらの研究は個別に研究がなされ，適用範囲も手法により異なっている．また，コンテンツ，デザイン，インタラクティビティの相互関連性が高いと考えられるウェブの特徴を十分に活かした映像の視聴方法が現状では提供されているとはいい難い．

そこで本稿では，映像に関連付けられたメタデータの付加価値性に着目し，これと映像を統合的にウェブページ上に表示可能とするような一連の枠組を,「映像とメタデータのウェブ化」として提案する．本方式では，映像だけでなくそこから得られるメタデータを含めてウェブページ上に展開する対象と考える点が特徴である．これにより，ウェブ化したコンテンツの閲覧の付加価値化・効率化・多視点化を図ることができると期待される．

本稿の構成は以下の通りである．2 節では，映像とメタデータのウェブ化の概念について説明し，3 節ではウェブ化の処理概要を述べる．4 節では実現のために必要な要素技術を整理し，5 節ではメタデータを映像内にオーバーレイ表示する機能を実現したプロトタイプ実装例を示す．最後に 6 節でまとめを述べる．

2. 映像とメタデータのウェブ化の概念

現在，ウェブ上で閲覧可能な映像の利用形態には，ファイルのダウンロードとストリーミングの 2 種類の方法が存在する．このような映像コンテンツを掲載するウェブ文書の問題点として以下が挙げられる．

- 映像コンテンツの中身の概観表示や必要部分の視聴を効率よく行なう手段がウェブ上に十分提供されているとはいえない．
- メタデータを含めコンテンツ自身の効率的作成手段が十分提供されているとはいえない．

そこで，本稿では，映像だけでなくメタデータを積極的にウェブ上に表示可能とする枠組を考える．この際，字幕等のメタデータだけでなく，メディア認識技術や半自動化手法により生成される各種意味内容を表す特徴量からなるメタデータを含めて考えることとする.多様なメタデータをウェブ化の対象とすることで，さまざまな状況でより付加価値の高い映像の閲覧が可能になると期待できる．

図 1 に,映像とメタデータのウェブ化の概念を示す．

ウェブ化においては，入力となる映像とメタデータを用いて，コンテンツ解析や必要なメタデータ生成が行われ，それらをウェブ上に表示するための変換処理が行なわれる．得られたウェブコンテンツを適当なブラウザで閲覧することにより，さまざまなメタデータ要素を活用した早見や特定シーン検索，適応的視聴などが，ウェブのインタラクティブ性を活かした形で可能になる．利用者のフィードバックは，適宜コンテンツ変換処理に反映され，ウェブコンテンツは適応的に再構成される．

ウェブ化処理とは，以下の要求条件を満たす機能を利用者に提供することであると考えられる．

- 映像の全体概要や特定シーン，内容に関連した情報を効率よく取得・確認できること．
- メタデータ要素をウェブページや映像上に適当な書式に従い適宜表示できること．
- 利用者に分かりやすい簡単なインタフェースで操作できること．

これらの要求条件を満たすために行なうべきウェブ化処理の概要を図 2 に示す．

まず，図中(1)では，映像データ $\mathbf{V}$ を入力とし，知識

K やその時点で利用可能なメタデータ **M** を参照することにより，コンテンツ解析・注釈付けを行なう．この際，ユーザの嗜好や履歴情報 I を反映させた処理を行なう場合もある．次に，(2)では，生成したメタデータ **M**，映像データ **V**，知識 **K** を入力とし，全体概要や特定シーン検索，関連情報表示を提供できるよう書式変換されたウェブコンテンツ **W** を生成する．(3)では，ユーザとのインタラクションに基づき，さまざまな閲覧機能が提供される．ユーザからの要求は必要に応じて(1)や(2)にフィードバックされる．

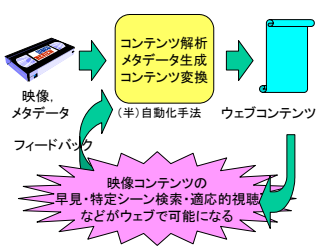

図 1．映像とメタデータのウェブ化の概念

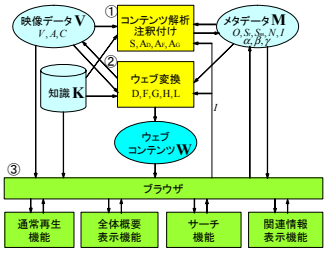

図 2．ウェブ化処理の概要

3. 映像とメタデータのウェブ化の処理概要

3.1. 映像とメタデータの定義

まず，映像データ **V** を以下のように定義する．

$$\mathbf{V}=\{V,A,C\}$$

ここで，V は動画像データ，A は音響データ，C はキャプション等のテキストデータを表す．各データは，次のようにいくつかのパラメータの関数で表現される．$V=V(p,f,r,s,c_s), A=A(p,f,r,s), C=C(p,c_c)$ ここで，p は時間や空間位置を指定する時空間位置パラメータ，f はフレームレートやサンプリング周波数，r は解像度や階調，量子化ビット数，s は SNR，ビットレート等の画質･音質をそれぞれ表すパラメータである．c_s は色空間と対象軸，c_c は文字コードをそれぞれ表すパラメータである．

また，メタデータ **M** を以下のように定義する．

$$\mathbf{M}=\{O,S_t,S_m,N,I\}$$

ここで，O はコンテンツの概要を表すメタデータ，S_t はコンテンツの構造を表すメタデータ，S_m はコンテンツの意味を表すメタデータ，N はコンテンツのナビゲーションおよびアクセスに関するメタデータ，I はユーザのインタラクションを表すメタデータを表す（表 1）．

3.2. ウェブ化の処理概要

まず，データ $data$ をパラメータ $param$ に関して $unit$ 単位または $element$ で例示される集合の各要素毎に分割する関数 $S(data, param, \{unit \mid element\})$ を考える．TV2Web[10]の例では，映像データ **V** について次のような時間軸方向の構造化を行なっている．

$$S(\mathbf{V},p,segment)=\{\mathbf{v}_i^{(segment)} \mid i=1,...,n_i\}$$
$$S(\mathbf{V},p,scene)=\{\mathbf{v}_j^{(scene)} \mid j=1,...,n_j\}$$
$$S(\mathbf{V},p,shot)=\{\mathbf{v}_k^{(shot)} \mid k=1,...,n_k\}$$

ここで，

$$\bigcup_i \mathbf{v}_i^{(segment)}=\bigcup_j \mathbf{v}_j^{(scene)}=\bigcup_k \mathbf{v}_k^{(shot)}=\mathbf{V}$$
$$\mathbf{v}_i^{(segment)}=\bigcup_{j_{i1}}^{j_{i2}} \mathbf{v}_j^{(scene)},\ \mathbf{v}_j^{(scene)}=\bigcup_{k_{j1}}^{k_{j2}} \mathbf{v}_k^{(shot)}$$

$\mathbf{v}_x^{(unit)}$ は，$unit$ 単位に分割された映像データの 1 セグメントを示す．この場合，分割処理は，動画像・音響・テキストの各データが互いに時間軸上で同期した位置で行なわれていることになる．映像データは図 3 のような構造に分割される．

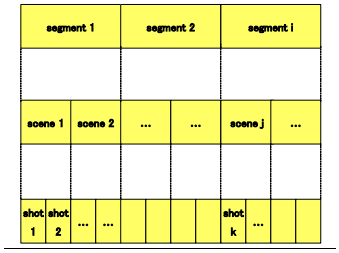

図 3．TV2Web による映像の構造化

一方，提案するウェブ化においては，分割関数 S を用いて映像データの各要素 V,A,C それぞれについてパラメータに関連した単位で分割することができる．例えば，ニュース映像のアンカーショットを表す動画像 V について，アンカーを表す前景とそれ以外の背景の 2 つの領域に空間分割することを考える．

$X=\{anchor, background\}$ とすると，関数 S により，以下のような結果が得られる．

$$S(V,p,X)=\{v_x^{(region)} \mid x=anchor, background\}$$

さらに，得られたアンカーオブジェクトに対し b_1=384[kbps]，b_2=1.5[Mbps]の 2 種類のビットレートを割り当てるようにさらに分割を行うと，$Y=\{b_1,b_2\}$ として，

$$S(v_{anchor}^{(region)},s,Y)=\{v_{anchor,y}^{(region)(bitrate)} \mid y=b_1,b_2\}$$

となる（図 4）．

これらの分割結果は，図 5 で示すストーリーボードインタフェース上での閲覧に利用される．例えば，時間方向に $\{segment, scene, shot\}$ の分割，空間方向に $\{anchor, background\}$ の分割，$anchor$ の画質に関して $\{b_1,b_2\}$ の分割がなされていたとする．

利用者が図 5 で示すインタフェース上のタイムラインのスケールを変換すると，表示シーンの内容が，図

5 の a と a-1 で示されるように,時間方向へ *segment* 単位から *scene* 単位へ，また，*scene* 単位から *shot* 単位へと変化する．

同様に，空間方向にレイヤーを移動すると，図 5 の b と b-{1,2}で示されるように，2 つのウィンドウが起動し，それぞれ *anchor* と *background* の内容のみが表示されるようになる．

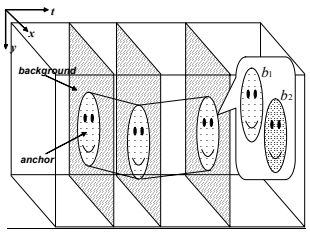

図 4．ウェブ化ビデオにおける動画像データの各パラメータ上での分割・構造化

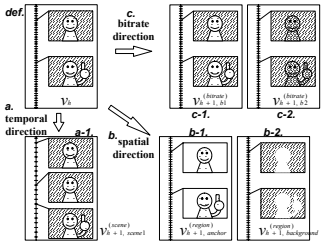

図 5．種々のパラメータ方向に沿ったレイヤー移動例

さらに，*anchor* についてビットレート方向にレイヤー移動すると，図 5 の c と c-{1,2}で示されるように，2 つのウィンドウが起動し，各々ビットレート b_1, b_2 の内容をもつ *anchor* と *background* が表示される．

また，以上のような分割は，音響データ A やテキストデータ C についても関連するパラメータ上で独立に行なうことが可能である．分割関数 S は，分割結果に関するメタデータ $\mathbf{M}$ を出力する．

さらに，以下のような関数を用いた構造化を行なう．

(1) ダイジェスト生成に関する関数

映像コンテンツ中の重要部分や全体概要を要約したダイジェストを生成する関数である．ダイジェスト用索引付け関数 A_D，および，ダイジェスト生成関数 D を以下のように定義する．

$$A_D(\mathbf{V}, \mathbf{M}, \mathbf{K}, I) = \{\alpha_{D,l} \mid l = 1,..., n_l\}$$
$$D(\mathbf{V}, \mathbf{M}, \mathbf{K}, \{\alpha_{D,l}\}, I) = \{\mathbf{v}_{D,i}\}$$

ここで，

$$\alpha_{D,l} \in \mathbf{M},\ \mathbf{v}_{D,i} \notin \mathbf{V}$$

$\alpha_{D,l}$ はダイジェスト生成に必要な索引，I はユーザによる嗜好・ダイジェスト生成条件，$\mathbf{v}_{D,i}$ は，ダイジェストを構成する 1 シーン，および，関連する説明文など音響・テキスト情報，場合によってはこれらの一部を表す．

ダイジェスト関数 D は，原データから重要と判断される部分を抽出する処理のみを意味するのではなく，メタデータ $\mathbf{M}$ や知識 $\mathbf{K}$ によりある程度の内容理解をした上で，新たな要約データを生成する処理を含んでいる．例えば，文献[9]では，テニスにおける試合状況の変化を選手の優勢度を使って把握し，重要部分を説明するためのナレーションテキストを新たに自動生成している．よって，$\mathbf{v}_{D,i}$ はオリジナルの映像データ $\mathbf{V}$ に必ずしも含まれていたものとは限らない．分割関数 S とダイジェスト生成関数 D により，映像データは図 6 のように構造化される（実際には，動画像・音響・テキストデータの各パラメータ上でそれぞれ構造化することが可能なので，より複雑で多次元的な立体構造をしている）．

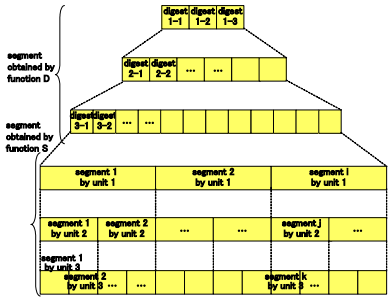

図 6．ウェブ化ビデオによる映像の構造化

(2) シーン検索に関する関数

コンテンツ中の与えられた条件を満たす特定部分を検索するための関数である．シーン検索用索引付け関数 A_F，および，シーン検索関数 F を以下の通り定義する．

$$A_F(\mathbf{V}, \mathbf{M}, \mathbf{K}, I) = \{\beta_{F,m} \mid m = 1,..., n_m\}$$
$$F(\mathbf{V}, \mathbf{M}, \mathbf{K}, \{\beta_{F,m}\}, I) = \{\mathbf{v}_{F,j}\}$$

ここで，

$$\beta_{F,m} \in \mathbf{M},\ \mathbf{v}_{F,j} \in \mathbf{V}$$

$\beta_{F,m}$ はシーン検索に必要な索引，I は検索時の問合せ，$\mathbf{v}_{F,j}$ は問合せに対して映像データ $\mathbf{V}$ から抽出した検索結果の 1 つを表す．

例えば，テニスの試合においてサービスエースシーンを検索する際に，$\beta_{F,m}$ は，サービスエースイベントを表す索引，あるいは，各選手やボールの動作イベントの組み合わせからなる索引に対応する[8]．A_F は索引を生成する過程，F は問い合わせに対して検索結果を出力する過程と考えられる．サービスエースという問い合わせ I の結果が $\mathbf{v}_{F,j}$ の集合として返される．

(3) 関連情報表示に関する関数

コンテンツに関する関連情報を適宜表示・生成するための関数である．関連情報表示用索引付け関数 A_G，および，関連情報表示関数 G を以下の通り定義する．

$$A_G(\mathbf{V}, \mathbf{M}, \mathbf{K}, I) = \{\gamma_{G,n} \mid n = 1,..., n_n\}$$
$$G(\mathbf{V}, \mathbf{M}, \mathbf{K}, \{\gamma_{G,n}\}, I) = \{\mathbf{v}_{G,k}\}$$

ここで，

$\gamma_{G,n} \in \mathbf{M},\ \mathbf{v}_{G,k} \notin \mathbf{V}$

ここで，$\gamma_{G,n}$は関連情報表示に必要な索引，Iはユーザによる嗜好・関連情報の生成条件，$\mathbf{v}_{G,k}$は関連情報表示用に生成・関連付けられた1データを示す．

例えば，俳優の着ている洋服などの関連情報を表示する際に，Iはユーザ入力，$\gamma_{G,n}$は俳優の洋服部分に対応した座標データ，洋服の内容データへのリンク等からなる索引に対応する．A_Gは索引を生成する過程，Gは関連情報へ表示画面を変更する一連の手続きと考えられる．俳優の洋服情報が$\mathbf{v}_{G,k}$の集合として表示される．

(4) データ階層化に関する関数

分割，ダイジェスト生成，シーン検索，関連情報表示のためのデータ生成結果を，詳細度制御によりスケーラブルに表示するための階層化データを生成する関数である．データ階層化関数Hを以下の通り定義する．

$H(\mathbf{V}, \mathbf{M}, \mathbf{K}, I) = \{\mathbf{v}_h, \mathbf{m}_h\}$

ここで，$\mathbf{v}_h$, $\mathbf{m}_h$は，それぞれ階層hを構成する映像データおよびメタデータを表す．

詳細度制御のための階層化としては，映像の各パラメータを各々軸とすることでいくつか方法が考えられる．

(a) 時空間方向に表示サイズが大きくなる
(b) フレームレートが高くなる
(c) 解像度が高くなる
(d) 画質がよくなる
(e) 同種の新しいデータが加わる
(f) メタデータによる付加価値情報が加わる
(g) 全く新しいデータに置き換わる

例えば，(a)のケースでは，動画像データをv，時空間位置パラメータpが図4に示すxyt座標系において

$p = p(x_{off}, x_{size}, y_{off}, y_{size}, t_{off}, t_{size})$

と表現されているとすると，

$$\begin{aligned} v_{h+1}^{(size)} &= V(p(t_{off,h+1}, t_{size,h+1})) \\ &= V(p(t_{off,h}, t_{size,h} + \delta t_{size,h})) \\ &= v_h^{(size)} \cup \delta v_h^{(size)} \end{aligned}$$

のように階層化できる(図7)．ここで，

$$\delta v_h^{(size)} = V(p(\delta t_{off,h}, \delta t_{size,h}))$$

$$\delta t_{off,h} = t_{off,h} + t_{size,h}$$

$$t_{off,h+1} = t_{off,h},\quad t_{size,h+1} = t_{size,h} + \delta t_{size,h}$$

なお，上記では，Vやpは考察対象となるパラメータのみ表記している（以下同様とする）．この階層化では，時間方向だけでなく，空間方向にサイズが変化する方法を考えることもできる．音響データAやテキストデータCについても同様の階層化が可能である．

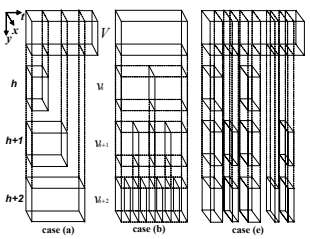

図7．(a), (b), (e)の場合における階層化

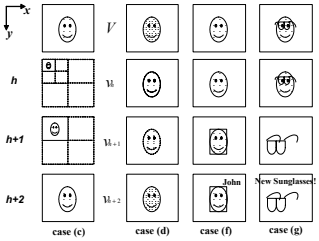

図8．(c), (d), (f), (g)の場合における階層化

(b)のケースでは，フレームレートを階層毎に変換する関数$F_f = F_f(f)$を用いて次のように階層化できる．

$v_{h+1}^{(framerate)} = V(f_{h+1}) = V(F_f(f_h))$

ここで，$F_f = 2f$とすると，

$$\begin{aligned} v_{h+1}^{(framerate)} &= V(2f_h) \\ &= V(p(t_{off}), f_h) \cup V(p(t_{off} + 1/2f_h), f_h) \\ &= v_h^{(framerate)} \cup \delta v_h^{(framerate)} \end{aligned}$$

のように構造化できる(図7).

(c)や(d)のケースについても関数F_fの代わりに，解像度を拡大する関数F_r(実際には，縮小画像を生成するフィルタ関数の逆関数という位置づけ)や，ベース信号に高 SNR を提供するエンハンス信号を付加する関数F_sを考えることにより，同様の階層化を実現可能である(図8)．音響データAについても同様である．

(e)のケースについては，

$$\begin{aligned} \{v_{h+1}^{(add)}\} &= \{V(p(t_{off,h+1}, t_{size,h+1}))\} \\ &= \{V(p(t_{off,h}, t_{size,h}))\} \cup \{V(p(\delta t_{off,h}, \delta t_{size,h}))\} \\ &= \{v_h^{(add)}\} \cup \{\delta v_h^{(add)}\} \end{aligned}$$

のように階層化できる(図 7)．新しいデータは任意の数追加できることを表している．これは，音響データAやテキストデータCについても同様である．

(f)や(g)のケースでは，映像データ$\mathbf{v}$に関連したメタデータ$\mathbf{m}$を用いて次のように階層化できる(図8).

$$\begin{aligned} &\{\mathbf{v}_{h+1}^{(add)}, \mathbf{m}_{h+1}^{(add)}\} \\ &\quad = \{\mathbf{v}_h^{(add)}, \mathbf{m}_h^{(add)}\} \cup \{\delta \mathbf{v}_h^{(add)}, \delta \mathbf{m}_h^{(add)}\} \end{aligned}$$

新しいデータは任意の数追加でき，階層が変わるごとに動画像・音響・テキストデータ間でメディアが変化するなど，映像データとしてのV, A, Cを任意に組み合わせた形式で階層データを構成できることを表している．例えば，映像データ上にメタデータから得られるテキストデータや座標データなどを重ね合わせること

によるデータ構成も可能であることを示している．

(5) ウェブデータ生成に関する関数

(1)から(4)の関数で生成された各種データを，ブラウザ上で表示するためにウェブデータへデータ変換する関数である．生成関数Lを以下のように定義する．

$$L(\mathbf{V}, \mathbf{M}, \mathbf{K}, I) = \mathbf{W}$$

ここで，$\mathbf{W}$はウェブ化ビデオとして表示可能なウェブページを表す．関数Lは，ユーザインタラクションIの値に応じて，映像とメタデータをウェブページ上でシームレスに利用・表示可能な仕組みを提供する．

特に，詳細度制御表示を行なう際には，(4)の(a)～(g)の例で階層化した各階層データを，ユーザインタラクションIの値に応じて適宜選択し，各階層データ間を移行しながらその間の表示をスムースに行なえるようなメタファと組み合わせること等により，直感的で効率のよいインタフェースをユーザに提供できると考えられる．図5は考えられるユーザインタフェースの一例である．

さて，以上の関数によって扱われる映像データ$\mathbf{V}$は複数のデータであってもよい．例えば，同じ日に放送された複数の番組グループのダイジェストを生成したり，同じタイトルあるいは同じトピックの番組グループのダイジェストを放送時間順に生成するといった処理が考えられる．基本的に，単一データを対象とした上記の関数を複数回使用する，あるいは，複数の映像データを結合した単一の映像データを考え上記の関数を適用するなどの方法により，複数データに対してもダイジェスト生成，特定シーン検索，関連情報表示などの機能を実現できると考えられる．

以上により，本稿で提案するウェブ化処理は以下のような関数で表現できる．

$$W(\mathbf{V}, \mathbf{M}, \mathbf{K}, I) = \{A(\mathbf{V}, \mathbf{M}, \mathbf{K}, I), T(\mathbf{V}, \mathbf{M}, \mathbf{K}, I)\}$$

$$A = \{S, A_D, A_F, A_G\},\quad T = \{D, F, G, H, L\}$$

ここで，Aは，映像の分割やコンテンツ解析・注釈付けを行ない，メタデータ$\mathbf{M}$を出力する関数群，Tは，ダイジェスト生成，シーン検索，関連情報表示用データ生成を行い，それらを階層データとして詳細度制御可能な状態に整え，ウェブ化ビデオとして表示範囲や形式を制御するウェブページを生成することにより，ウェブデータ$\mathbf{W}$，および，映像データ$\mathbf{V}$を出力する関数群である．

4. 実現に必要な要素技術

ここでは，前節までに説明したウェブ化処理を実現するために必要な要素技術について述べる．

まず，映像データの分割関数について．時間方向の分割については，カット検出，シーン識別・分類が基本となる．空間方向の分割については，領域分割，オブジェクト検出が重要である．フレームレートや解像度，画質に関する分割については，基本的に，映像符号化の際に用いられるスケーラビリティ技術によってデータを構成することが重要となる．スケーラビリティ技術により，時間・空間・解像度・SNRの各要素に対して階層的にデータを構成し，各階層に割り当てられた品質で映像を復元することができる．近年では，より細かい階層性を有するFGS(Fine Granular Scalability)技術の研究も活発に行なわれている．

ダイジェスト生成の関数については，重要度計算，重要部分抽出，映像内容のイベント解析，ストーリー理解，テキスト言い換え，要約文生成技術等が必須となる．複数データに跨る文脈理解・比較・ダイジェスト生成技術も重要である．さらに，ユーザの嗜好や履歴に応じて内容を動的に再構成する個人化技術も不可欠となる．

シーン検索に関する関数については，画像・音響処理，自然言語処理等を駆使したマルチモーダルなコンテンツ解析技術，映像内容のイベント解析，顔やジェスチャの認識技術，機械学習やパターン認識技術が不可欠となる．イベントと索引の柔軟な対応付けを可能とする索引構成技術，検索条件の柔軟な表現技術等も重要である．

表1．想定しているメタデータの種類

種別	内容	備考
コンテンツの概要	タイトル，製作者，作成日時，作成場所，ジャンル，キーワード，アブストラクト，著作権等のコンテンツの概要を表すデータ	手入力可
コンテンツの構造	各データ/セグメントの存在場所，各データ/セグメントの再生時間	手入力可，できれば自動化が望ましい．
コンテンツの意味	オブジェクト座標(群)データ，イベント定義，イベントクラスデータ，イベント時刻・位置データ，計数データ等のコンテンツの意味に関連するデータ	詳細なデータの手入力は非現実的．自動化が必要．
コンテンツのアクセス	要約再生に必要なセグメントあるいはキー画像に関するデータ	手入力可，個人化適応を考えると，自動化が望ましい．
ユーザインタラクション	ユーザの嗜好や履歴に関するデータ	フィードバックあるいは自動収集

関連情報表示に関する関数については，メタデータと映像の重ね合わせ表示，メディア間同期，映像変換，モザイク生成技術等が不可欠となる．映像データのジャンルや内容に応じてどのような情報をどのように表示するのが適当であるか学習する手法等も重要となる．

データ階層化に関する関数については，上述したスケーラビリティ技術が必須となる．また，メタデータによる付加価値情報の追加や全く新しい情報に置き換わるような詳細度制御に関しては，ダイジェスト生成技術，関連情報表示データ生成技術が不可欠となる．

ウェブデータ生成に関する関数については，映像内容の概要や必要とする特定部分・関連情報をなるべく簡単な操作・少ない操作で取得させるためのユーザインタフェース技術，ユーザの状況に合わせた適応的なレイアウト技術等が重要となる．

また，いずれの場合にも共通するが，メタデータ生成技術は特に重要である．表 1 にウェブ化処理において想定しているメタデータの種類の一覧をまとめる．

5. プロトタイプの実装

ウェブ化ビデオの応用例として，指定されたメタデータ要素を映像フレーム内にオーバーレイ表示する機能をプロトタイプとして実装した．

全体のシステム構成を図 9 に示す．本稿では，メタデータ要素をある程度限定するため，対象映像をテニスとした．

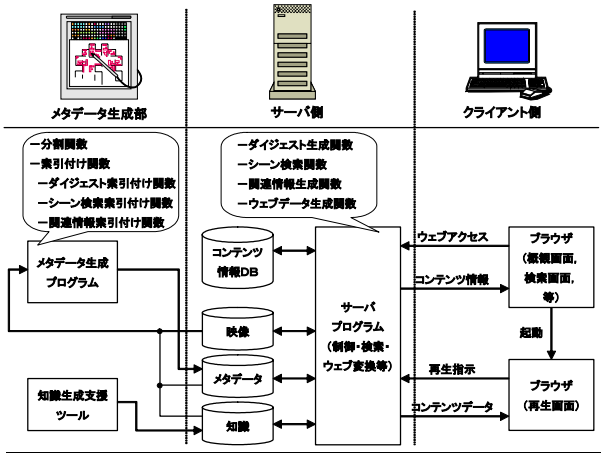

図 9．全体のシステム構成

まず，メタデータ生成部では，手入力および半自動入力により MPEG-7 形式のメタデータを生成する．メタデータの要素としては，選手やボールの座標，選手の基本動作（動作 ID，開始・終了時刻等からなる）などを含んでいる．

また，ドメインに特化した知識データは人手で準備する．本稿では，コート仕様，コート区画名とその範囲，基本動作とその成立条件，一般動作とその成立条件などを独自のスクリプトによってルール定義した．これらの知識は，将来的には，OWL など XML 形式のオントロジー言語によって表現できる可能性がある．

サーバは，利用者要求を受け取ると，適切な応答結果を含むウェブデータを生成しクライアントに返す．特に，ウェブデータ生成関数の処理手順は図 10 のように行う．例えば，サーバがコンテンツ概要を取得するよう要求された場合，コンテンツ情報 DB から必要な概要情報を取得した後，スタイルシートで HTML 形式に変換し，概要一覧を表示するウェブデータをクライアントに返す．また，30 秒のダイジェスト視聴が要求された場合，そのダイジェストデータが予めメタデータ内に含まれている場合は，該当部分をメタデータから探索し，適当なデータ変換の後クライアントにウェブデータが返される．ダイジェストデータが予めメタデータに含まれていない場合は，ダイジェスト処理エンジンがダイジェスト生成を行い，結果のメタデータを適切な形式にデータ変換した後，HTML データがクライアントに返される．

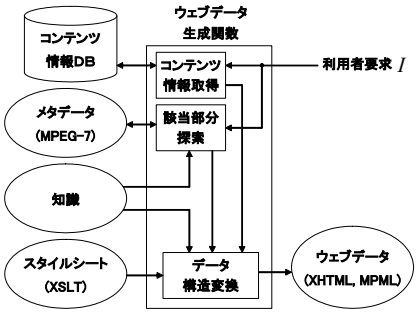

図 10．ウェブデータ生成関数の処理手順

本稿では，特に，図形やテキストといったメタデータ要素を映像フレーム内にオーバーレイ表示する機能に関して実装を行った．SMIL など従来のマルチメディアプレゼンテーション言語では，映像フレーム内に情報をオーバーレイ表示する機能はサポートされていない（正しくは SMIL2.0 でサポート済み）．よって，本稿では，次のアプローチで本機能の実装を行った．

- 図形やテキストのオーバーレイ表示に必要なマークアップ言語を新たに定義する（仮に，メタデータ表示記述言語 MPML; Metadata Presentation Markup Language と呼ぶ）．
- MPML を解釈しオーバーレイ表示するプログラムを，ActiveX を利用して作成する．本プログラムは，既存ブラウザにプラグインとして取り込むことができる．

現在，MPML は，映像ソースやその表示開始・終了時刻，フレームごとの座標，多角形，円などの基本図形，色やフォントを含むテキスト等を指定できる簡単な内容で構成されている（XML 形式）．また，ActiveX プログラムはプラグインとして既存ブラウザに取り込まれることで所定の機能を実現している．

さて，メタデータの映像フレーム内へのオーバーレイ表示機能の応用例の一つとして，シーン検索結果の根拠を示す機能が挙げられる．例えば，テニスの「スマッシュ」動作は，基本動作インデックスの組合せからなる次のルールで検索される(図 11)[8].

一方の選手がある時刻において“forehand_swing”か”backhand_swing”を行い，同一選手が次に行なう動作が“overhead_swing”である．

図 12 は，このルールが各検索結果においてどのように成立しているかという根拠を，映像に選手やボール位置，選手の基本動作名テキストを重ね合わせて視覚化することで示した例である．この視覚化により，各検索結果において該当ルールがどのように成立して

いたかという様子を簡単に確認できるようになる．

一般に，特定シーンの検索結果を視聴する際，単純に検索結果を通常再生していただけでは，どの部分がどの条件に一致してその検索結果が得られたのかという根拠を，通常再生画面から把握することは極めて困難である．フレーム内のオブジェクト数が増えた場合や，成立条件が複雑になってくればなおさらである．

検索結果の根拠を表示する機能は，生成したインデックスの精度を視覚的に確認可能なため，インデックス生成作業の効率そのものを格段に向上できると期待される．インデックス生成を行なう際の効率的な確認手段として不可欠な機能の一つになると考えられる．

さて，本稿では，映像フレーム内にオーバーレイ表示可能なメタデータ要素を，基本図形やテキストに限定して実装した．今後は，例えば MPEG-7 で規定されている他のメタデータ要素を，付加価値性を高めた形で表現するためのデータ変換，および，付随するスタイルシート・知識データを適宜充実させていく必要がある．

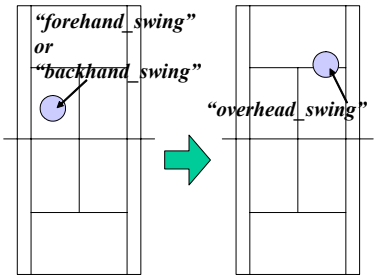

図 11．スマッシュの成立条件

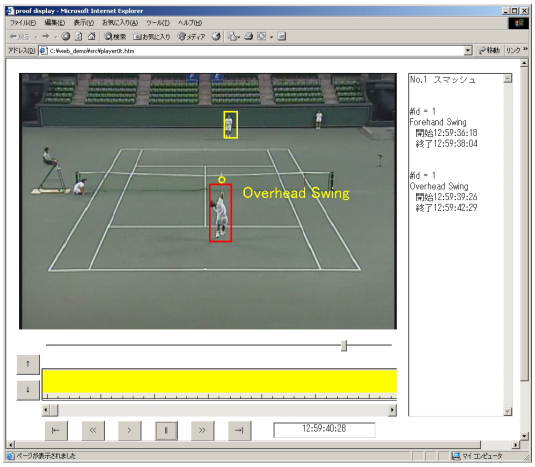

図 12．メタデータのオーバーレイ機能を用いた検索結果の根拠表示

6. まとめ

本稿では，映像に関連付けられたメタデータの付加価値性に着目し，これと映像を統合的にウェブページ上に表示可能とするような一連の枠組を，「映像とメタデータのウェブ化」として提案し，その処理概要，実現のために必要な技術，試作したプロトタイプについて述べた．

本稿では特に，自動抽出した基本図形やテキストといったメタデータ要素を，映像フレーム内にオーバーレイする形で表示する機能の実装を試みた．その結果，シーン検索結果の根拠を効率よく表示する機能を実現することができ，映像視聴・閲覧の付加価値化・効率化・多視点化を図る応用例の一つを示すことができたと考えられる．

今後は，表示可能なメタデータ要素と表示形態の種類・形式をさらに工夫し，ウェブ化ビデオブラウザのプロトタイプ作成を進める予定である．

文　献

[1] Babaguchi, N., Kawai, Y., and Kitahashi, T.: Generation of personalized abstract of sports video. ICME, FP4.4, 2001.

[2] Christel, M.G., Huang, C.: Enhanced access to digital video through visually rich interfaces. ICME, MD-L5.1, 2003.

[3] Gong, Y., Sin, L.T., Chuan, C.H., Zhang, H., Sakauchi, M.: Automatic parsing of TV soccer programs. Proc. ICMCS, pp.167-174, 1995.

[4] Hanjalic, A.: Generic approach to highlights extraction from a sports video. ICIP, MA-S1-1, 2003.

[5] Hashimoto, T., Kataoka, T., Iizawa, A.: Personal Digest System for Professional Baseball Programs in Mobile Environment. Mobile Data Management 2003, pp.396-400, 2003.

[6] Hashimoto, T., Shirota, Y., Iizawa, A., Kitagawa, H.: A Rule-Based Scheme to Make Personal Digests from Video Program Meta Data. DEXA, pp.243-253, 2001.

[7] Haubold, A., Kender, J.R.: Analysis and interface for instructional video. ICME, AIVP-L6.5, 2003.

[8] Miyamori, H.: Automatic annotation of tennis action for content-based retrieval by integrated audio and visual information. CIVR2003, LNCS2728, Springer-Verlag, pp.331-341, 2003

[9] Miyamori, H.: Automatic generation of personalized video summary based on context flow and distinctive events. VLBV03, LNCS2849, Springer-Verlag, pp.111-121, 2003

[10] Munisamy, M., Sumiya, K., Tanaka, K.: TV2Web: generating and browsing web contents from video with metadata. DEWS2003, 8-P-9, 2003.

[11] Nakamura, Y., Kanade, T.: Semantic analysis for video contents extraction - spotting by association in news video. ACM Multimedia, pp.393-401, 1997.

[12] Saur, D.D., Tan, Y-P., Kulkarni, S.R., Ramadge, P.J.: Automated analysis and annotation of basketball video. Storage and Retrieval for Image and Video Databases V, SPIE-3022, pp.167-187, 1997.

[13] Smith, M., Kanade, T.: Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques. CVPR, 1997.

[14] Sudhir, G., Lee, J.C.M., Jain, A.K.: Automatic classification of tennis video for high-level content-based retrieval. CAIVD'98, 1998.

[15] Uchihashi, S., Foote, J., Girgensohn, A., Boreczky, J.: Video Manga: generating semantically meaningful video summaries. Proc. ACM Multimedia 99, 1999.