ビデオアイコンダイアグラムによる映像内容の構造表現

村山 正司† 伊津野 英克† 中村 裕一†‡ 大田 友一†

†筑波大学 機能工学系
〒 305-8573 つくば市天王台 1-1-1
E-Mail: murayama@image.esys.tsukuba.ac.jp
‡科学技術振興事業団, さきがけ研究２ 1

あらまし:
映像内容を簡潔に表現する方法として，ビデオアイコンダイアグラム（VID）を提案する．VID は，映像の代表的なフレームをアイコンとし，それを映像中の順序関係，包含関係，同値関係等の重要な関係にしたがって概念図的に配置するものである．これにより，各映像断片の持つ情報や役割を簡潔に表現する．これにより，映像全体の構造もわかりやすく表現される．本稿では，VID の提案とともに，VID を生成する方法として，XML 記述されたシナリオと映像を対応づけ，それによって VID のための記述を得る方法について述べる．また，実際に撮影された映像を概念図として表現する実験を行った．

キーワード: ビデオアイコンダイアグラム，映像処理，マルチモーダル処理，構造可視化，映像のタグ付け

Video Icon Diagram: Representation of Video Contents Structure

Masashi Murayama† Hidekatsu Izuno† Yuichi Nakamura†‡ Yuichi Ohta†

† Institute of Engineering Mechanics and Systems, University of Tsukuba
1-1-1 Tennoudai,Tsukuba, 305-8573,Japan
E-mail: murayama@image.esys.tsukuba.ac.jp
‡ PRESTO, Japan Science and Technology Corporation(JST)

Abstract:
We proposes “Video Icon Diagram(VID)“ for representing inner structures of a video. VID is a graphical diagram composed of video icons each of which illustrates a video segment, such as a shot. The icons and their layout represent semantics relations such as order, hierarchy, or equivalence, and combination of those structures represent the semantic structure of a video. In this paper, we first propose our framework and the methods for generating a VID by making correspondence between a scenario tagged by XML and a video. Our experiments shows the effectiveness of our method.

key words: video icon diagram, video analysis, multimodal analysis, video structure visualization, video tagging

1　はじめに

映像は動画と音声の複合によって，分かりやすく多くの情報を伝えることができるため，これからの電子メディアとして一層の活用が期待されている．しかし，映像は時間軸を持ったストリーム形のメディアであるため，冗長性が高く，一覧性が悪いことが問題となっている．例えば，料理番組を考えてみた場合，次のような内部構造を把握するためには，視聴者が映像を注意して見たうえで，頭のなかで内容を再構成する必要がある．

- どの食材がどのように調理されて形を変えていったか．例えば，野菜炒めでは野菜を切り，炒める等という一連の手順が存在する．
- どのような処理が並行して行われているか．例えば，ライスカレーを作る際に，白飯を炊くと同時にカレールウを調理するという二つの作業が並行して行われる．

これまで映像処理や構造化のために種々の方法が提案されてきたが，このような関係を分かりやすく表現することは難しい．例えば，重要部分だけを残して要約映像を生成しても [1]，出力が映像であるために，分かりやすさは改善されない．

本研究では，この問題に対して，映像の構造を図化する新しい手法，**ビデオアイコンダイアグラム** (Video Icon Diagram, 以下 VID と略記する) を提案する．VID は，空間的，時間的関係を始め，映像の種々の意味的構造を概念図として表現する手法である．これにより，例えば順序関係や因果関係にしたがって手順を分かりやすく表現したり，階層関係や同値関係にしたがって，類似するものをまとめあげて表現することができる．これらは映像のインデックスや要約として適切な表現となる．

以下本稿では，まず VID の基本的な考え方について説明し，次に VID のためのデータ記述形式を提案する．さらに，シナリオ等と映像を対応づけることにより VID のデータ記述を得る方法，また，それを用いて VID を生成する方法を提案する．

2　ビデオアイコンダイアグラム

2.1　映像の内部構造

視聴者が，教材映像のような物事を説明する映像を見る場合，空間的関係，順序関係，因果関係，階層関係，その他の典型的な関係を理解することが重要となる．

例えば，以下のような関係を考えてみよう．

- 被写体 A と被写体 B との関係．
  例えば,「まな板」と「食材」の間には空間的上下関係がある．あるいは,「かつお」と「かつおの切身」の間には階層関係がある．
- 被写体 (人物) の行動 A と行動 B との関係．
  例えば,「食材を切る」シーンと「切られた食材を焼く」シーンの間に一連の順序関係などがある．
- 行動 A と被写体 B の関係．
  例えば,「かつおを切る」シーンと「かつおの切身」の間には，ある動作とその結果物を表わす因果関係が存在している．

これらの重要な関係にしたがって映像内容を整理し，表現することが，映像の要約として効果的であり，視聴者の理解を助ける．VID はこのような関係の概念図化を目的とする．

ここで，各々の関係を持つ実体は被写体，映像内での出来事，映像外の被参照物等，場合により異なる．しかし，本研究ではこれら実体を細かく分類せず，全て**ビデオアイコン** (各映像断片の代表画像) で代用し，アイコン間の関係を図示する．これは，VID を見るのが人間であり，関係を持つ実体を人間が推測することが比較的簡単だからである．さらに，このプロセスを確実にするために，VID では各アイコンに文字で注釈を添える機能を備える．

関連する研究としては，従来より提案されてきた種々の映像の要約表現法 [2] があげられる．それらは，大別すると次の二つになる．

時間圧縮型: 映像を時間的に縮めて見せる方法．要約映像や早回し，代表フレーム画像を順に提示していく方式などがある．時間的な文脈構造を崩さずに表現できるという利点を有する．

空間展開型: 映像を平面上に展開して見せる方法．シーンの代表フレーム画像を時間順に並べていくもの (図 1 左) から，平面を自由に使うコラージュ的な方式まで様々なものがある．

時間圧縮形の要約では，時間的に離れた部分の関係を表現することは難しく，そこで我々は，空間展開型をより発展させた形として VID(図 1 右) を提案する．

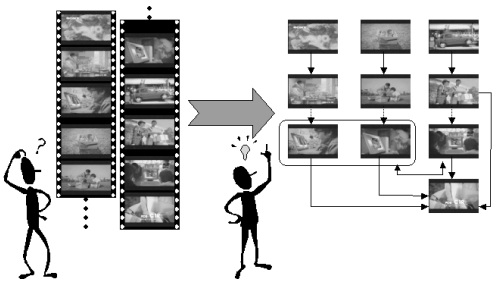


図 1: 映像の表現：単純な空間展開 (フィルム) とビデオアイコンダイアグラム

2.2 ビデオアイコンダイアグラムの構成

VID では，前述した関係を図 2 のように表現する．この関係を取り扱うため，関係の種類と属性という二つの分類を導入する．関係の種類には次の 3 つがある．

同値関係： 被写体間の同値関係や**セグメント** (映像断片のことを以後，セグメントと呼ぶ) の同時性など．例えば図 2(a) は違う視点から撮られた同じ時刻の映像であり，アイコンを線分で連結することで時間属性を持つ同値関係として表現している．

順序関係： セグメント間の時間的順序関係や，対象物体の変化などに代表される意味的順序関係，またイベント間の因果的順序関係など．図 2(b) は手順を示すセグメントにその入力要素と出力要素が有向線分によって連結し，過程の属性を持つ順序関係を示す．

包含関係： ある被写体と他の被写体の集合的，時間的階層関係など．図 2(c) では，上位要素がそれに属する 4 つの下位要素を囲むような配置によって集合属性を持つ包含関係を表現している．

関係の属性には次の 6 つがある．

時間： 時刻，あるいは時系列上の関係を示す．

空間： 実空間での物理的位置関係を示す．

因果： 原因・理由を示す．

過程： 入出力や処理の流れを示す．

集合： 組織や集合論上の関係を示す．

その他： 上記のいずれにもあてはまらないもの．

図 2 にあげたように，VID では領域図と連結図という 2 つの基本パターンを用いている．

連結図: 連結によって同値関係を，有向矢線によって順序関係を表現すること等ができる．

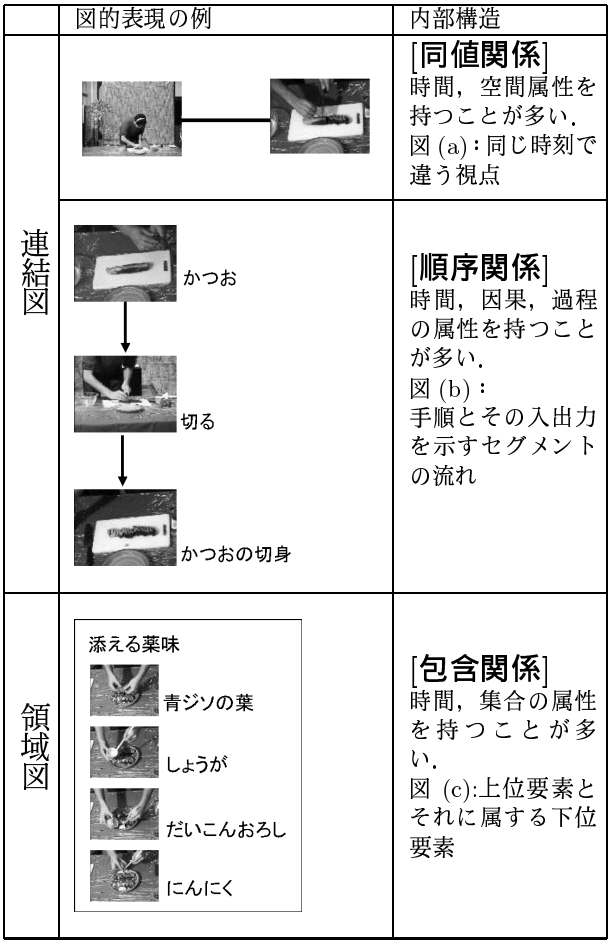


図 2: 図の基本パターンと映像の内部構造との対応

領域図: 閉領域によって包含関係を表現すること等ができる．

この 2 つのパターンを組み合わせることにより，同値・順序・包含等の，映像の基本的な内部構造を表現することができる．

ここで提案する VID の記述・生成システムは，筆者らが提案している文書の概念図化システム [3] を基にしている．この文書の概念図化システムでは，文書に構造記述タグを付け，それにしたがって文書の意味的構造を概念図とする．このシステムを VID のために利用できるのは，映像の内部構造が文書の基本的な構造と似た面を持つため，基本的な考え方をそのまま利用できるからである．

VID の基となる記述は，シナリオまたは発話文 (トランスクリプト) とする．シナリオの場合，セリフや動作説明に XML 形式でタグを付与し，対応する映像のフレーム番号，カメラ番号 (ストリーム番号) を記述する．これを基に，シナリオ中の文字列や映像中の代表画像を概念図の要素とし，要素を概念図中に配置する．以下，記述の詳細を次節で説明する．

表 1: VID スクリプトに用いる XML タグ

XML タグ名	説明
<scenario>	シナリオ全体を定義する.
<definition>	映像ストリームや映像外オブジェクトを定義する.
<serif>	<scenario> 中でセリフ全体を定義する.
<event>	<scenario> 中で動作全体を定義する.
<sentence>	<serif> 中でのセリフの一文. 発話者, 発話時刻などを属性値として持つ.
<word>	セリフ中の単語などを定義する.
<scene>	<event> 中で一連の動作を定義する.
<action>	<scene> 中での動作要素. 動作の種類や動作が起こるタイミング, 動作主などを属性値として持つ.
<input>	<action> の入力要素. 要素の実体参照などを属性値として持つ.
<output>	<action> の出力要素. 要素の実体参照などを属性値として持つ.

表 2: シナリオ例

セリフ	動作
村山「かつおを一センチ厚に切ります」	村山:かつおを切り始める.
村山「次に切身を盛り付けます」	村山:皿に切身を盛り付けていく.

3 VID のデータ表現: VID スクリプト

3.1 XML による VID スクリプトの記述

VID の基となる記述はシナリオまたはトランスクリプトとし, これらに XML タグを付けて XML 文書とする. 本研究ではこれを **VID スクリプト**と呼ぶ. 具体的には, 映像セグメントを記述するための XML 要素と, セグメント間の関係を記述するための XML 要素を表 1 にあげるタグを用いて定義し, 映像の内部構造を宣言的に記述する. ただし, 映像はバイナリデータであるため, VID スクリプトから直接の参照ができないので, 以下のようにして記述する.

セグメントの表現: 映像中のセグメント開始フレームとセグメント最終フレームをそれぞれ時刻で記述することで XML 要素から参照する. また, 各 XML 要素にユニークな ID を割り振ることで, 各セグメントに対して間接的な参照ができる.

セグメント間関係の表現: 関係そのものは XML 要素として記述する. 先に割り振った ID を属性値として参照することで, どのセグメント間の関係であるかを示す. また, 関係の内容については属性や要素名によって記述する.

```
<scenario>
<definition>
  <object id="o1" type="material"> かつお</object>
</definition>
<serif>
  <sentence speaker="村山" id="s1" time="">
  まず、事前に焼き付けておいた<word ref="#o1" id="A">
  かつお</word>を一センチ厚に切ります</sentence>
</serif>
<event>
  <scene summary="かつおのたたきの調理">
    <action time="type:exist; begin:#s1"
                  agent="村山" label="切る"
               type="cook:transform" id="a1">
      <input ref="#A" label="かつお"/>
      <output label="かつおの切身"/>
    </action>
  </scene>
</event>
</scenario>
```

図 3: VID スクリプトの例

また, XML 自体は木構造のデータ表現であるが, XLink などの拡張を用いることで, 映像の内部構造のようなネットワーク的な構造をも表現することができる. ただし, フレーム画像アイコンだけでは図が分かりにくいことがあるため, ビデオアイコンに添える文字列をタグを使って付加しておく. このように MPEG-7 と良く似た枠組みで, 映像の構造記述を行う. 将来 MPEG-7 への移行を行う要求があった場合にも, XSLT[4] などを用いることでデータの変換を可能にする.

本研究で想定するシナリオ例を表 2 に示し, このシナリオにタグを付けた例を図 3 に示す. まず <definition> タグで「かつお」というオブジェクトを定義し, それに $o1$ という ID と「食材」を意味する属性値を付加している. 次に<serif>タグによりセリフを定義しており, <sentence>タグに ID 属性を設定することで, セリフ文に $s1$ という ID を振っている. また, <word> タグを用いることで単語「かつお」に A という ID を付けている. 後半はシナリオ中の動作説明であり,「切る」という動作が<action>タグにより定義されている. <action>タグ中の type 属性は動作の種類を, また time 属性は対応する映像セグメントの時刻情報を示すが, これらは 4 章で説明する. 動作の入力である<input>要素は ID 参照を示す ref 属性によりセリフ中の「かつお」を示しており, 一方で出力要素を示す<output> 要素は独立した要素「かつおの切身」を記述している.

3.2 VID スクリプトからの VID 生成

VID 生成処理は基となる概念図生成システムの処理に, ビデオアイコンの抽出処理を加えたものになる.

ビデオアイコンの抽出は，該当映像セグメントの中から代表フレーム画像を選ぶ処理となる．その概要は以下のようになっている．

- 代表画像を抽出すべきフレームの時刻は VID スクリプト中で区間的に指定されている．
- 各々の意味的構造と，代表画像を選ぶべきセグメント内の場所 (時刻) との対応関係を予め決めておき，外部に記述しておく．例えば，<input>(処理の入力) に対しては，対応区間の開始 (時刻) を選ぶ等．
- また，本研究では複数カメラ (映像ストリーム) を想定しているため，カメラ (映像ストリーム) を選ぶ処理も必要となる．これに対しても，各意味的構造を表現するタグの種類ごとに予め決めておく．例えば，<input>(処理の入力) に関しては作業台のクローズアップ映像を指定する等．

基本的には上記の設定により，映像ストリームから簡単に代表画像を抽出することができる．ただし，セリフと直接関係しない被写体は，どの時刻に映るかがシナリオから分からないため，人手により最適な時刻を指定することが必要となる．これらは動作認識や物体認識などと組み合わせることによりある程度解決できると考えられるが，本研究では今後の課題となっている．

概念図生成のアルゴリズムは，図 4 に示すようなデータ構造の多段階変換である [3]．概念図生成システムにデータを入力すると，記述されているセグメントに対応するビデオアイコンが描画され，またセグメント間の関係に応じて空間内に再配置される．例えば，順序関係ならば上位アイコンの下方もしくは右方に下位アイコンが配置され，それらが有向矢線で結ばれる．また，本システムには編集支援機能が備わっており，概念図が満足いくものではない場合に，ユーザが簡単に編集することができる．

4 シナリオと映像の照合による VID の生成

映像のシナリオやトランスクリプトが得られる場合，それにタグ付けを行うことによって，VID スクリプトの大部分を用意しておき，その後，自動的に映像と対応付けることによって VID を生成する手法について説明する．映像の制作現場では，ほとんどの場合，シナリオが予め得られ，また，視聴者側でも文字放送などからトランスクリプトを得ることも可能であるため，このような手法は多くの場面で利用可能であると考えられる．

1. **記述の入力・構文解析**
2. **関係ネットワーク構成**
   (a) セグメントを示すノード生成
   (b) 各ノードの意味的属性決定
   (c) セグメント間関係を示すリンク生成
   (d) 推論による関係伝播などの知識処理
3. **図フレーム構成**
   (a) 図要素を示すフレーム生成
   (b) フレームの幾何学的属性決定 (囲みか矢線か等)
   (c) フレーム間の幾何学的関係決定
4. **描画ルール適用による図の描画**
   (a) ビデオアイコン等の図要素を初期化
   (b) 図要素の生成（初期配置）
   (c) 図要素のレイアウト

図 4: 概念図生成の流れ

4.1 料理番組のための VID スクリプト

これまで述べてきた VID スクリプト書式，および VID 生成処理を，料理番組を模擬した実際の映像データに適用する．料理番組を選んだ理由は，定型的な説明が多く，説明しなければならない事柄がはっきりとしているからである．

ここではまず，料理で重要となる動作を図 5 にあげ，これを前述の構造と対応付ける．この動作は VID スクリプトで<action>中の属性 type により記述されるが，その多くは「変化 (type 属性の値としては transform)」などに代表されるような過程属性を持つ順序関係と対応している．他には，映像セグメント同士の入れ子関係は時間属性を持つ包含関係に対応している．このような対応関係を用いることで，前述の図生成システムを VID 生成に適用できる．

これらの動作記述等を用いることにより，図 3 のような VID スクリプトを用意する．ただし，この段階では対応する映像セグメント開始時刻など，必要な情報の一部は未知であるので，その部分の属性は空きとしておく．例えば，6 行目の time 属性の値は，対応するセグメントの時刻情報が入るため空にしておく．これら空欄になっている部分の値を次節で述べる方法により求める．

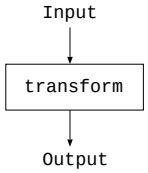
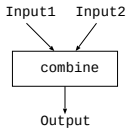
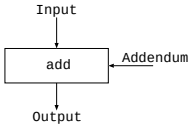
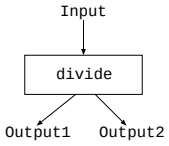
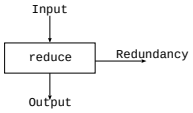
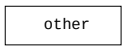
動作とその説明	動作記述の図的表現
変化，遷移 ・切断による変化：切る，割る． ・力による変化：つぶす，のばす． ・熱による変化：炒める，煮る． 入出力要素と過程属性の順序関係を持つ．	
合成 ・混ぜる，くるむ，詰める 上の「変化」と同じ順序関係を持つ．	
加算 ・かける，まぶす，盛る，のせる 上の「変化」と同じ順序関係を持つ．	
分割 ・分ける 上の「変化」と同じ順序関係を持つ．	
減算 ・アクをとる，流す，移す，取り出す 上の「変化」と同じ順序関係を持つ．	
変化を伴わない動作 ・説明する，あいさつする． 特に他の要素と関係を持たない．	

図 5: 料理映像シナリオでの動作定義

4.2 映像とシナリオの対応付け

実際の映像データに対し，映像中の発話情報を利用して VID スクリプトとの対応をとり，スクリプト中で空になっている映像セグメント時刻などを求める．

まず映像中の発話は，音声認識によってトランスクリプトに変換する．このデータと，シナリオ中の文字列とのパターン間の距離を類似度として定義し，動的計画法に基づく文字列パターンマッチングを行う．ここで用いている音声認識システムが出力するトランスクリプト，およびシナリオはかな漢字混じりの文章であるが，そのままでは部分対応がとりにくいため，双方のデータをひらがなで統一しておく．そのために，日本語形態素解析システム JUMAN[5] を用いて，漢字の読みを取得する．

具体的なマッチングアルゴリズムは以下のようになる．文字列 T 中からパターン R を探すとし，これら

1. $g(0,0)=0, B(0,0)=0$
2. $for(i=1,2,\cdots,I)$
   $g(i,0)=0, B(i,0)=i$
3. $for(j=1,2,\cdots,J)$
   $g(0,j)=g(0,j-1)+2, B(0,j)=0$
4. $for(i=1,2,\cdots,I)$
   $for(j=1,2,\cdots,J)$

$$g(i,j)=min\begin{cases} g(i-1,j)+1 & (a)\\ g(i-1,j-1)+2 & (b)\\ g(i,j-1)+2 & (c)\end{cases}$$

$$\begin{cases} (a)\text{ の場合} & B(i,j)=B(i-1,j)\\ (b)\text{ の場合} & B(i,j)=B(i-1,j-1)\\ (c)\text{ の場合} & B(i,j)=B(i,j-1)\end{cases}$$

5. $g(i,J)$ を最小にする i の値 i_{match} を探索．

図 6: 文字列 DP マッチングのアルゴリズム

```
<sentence speaker="村山"
                time="begin:972567724290">
  かつおを一センチ厚に切ります．</sentence>
```

図 7: 時刻データが埋め込まれた VID スクリプト

の二つの文字列を次のように定義する．

$$T=a_1a_2a_3\cdots a_I,\ R=b_1b_2b_3\cdots b_J$$

ただし，a_i, bj は個々の文字を表す．図 6 に示すアルゴリズムの過程 1-3 で配列を初期化し，過程 4 のループによって配列 $g(i,j)$ と 配列 $B(i,j)$ の各値を求める．最後の過程 5 で，$g(i,J)$ を最小にする i の値を探索する．こうして求まった i の値を i_{match} とする．配列の値 $g(i,J)$ は，T の部分文字列 $a_{B(i,J)}\cdots a_i$ とパターン R との間の距離を示しており，この値を最小にする i_{match} を求めることで R にマッチングする T の部分文字列 $a_{B(i_{match},J)}\cdots a_{i_{match}}$ が得られる．

本稿では，シナリオ中のセリフ文が発話された時刻を得るために，トランスクリプト全体を T とし，句点を検出することで取り出したシナリオのセリフ一文を R として上記アルゴリズムを適用する．そして得られた T 中の文字 $a_{B(i_{match},J)}$ の生起時刻がセリフ R の開始時刻となる．

対応関係が求まった後，各々の発話時刻を用いることにより，シナリオのセリフの各部分と映像フレームとの対応が求まる．これにより，映像との対応が取れた図 7 のような VID スクリプトが得られる．これは，セリフの一文を示す記述で，2 行目の `time` 属性に開始時刻を示す値が代入されている．

5 実験例

前節で述べたように，実験対象とするのは料理番組とし，研究室でその模擬撮影を行った．

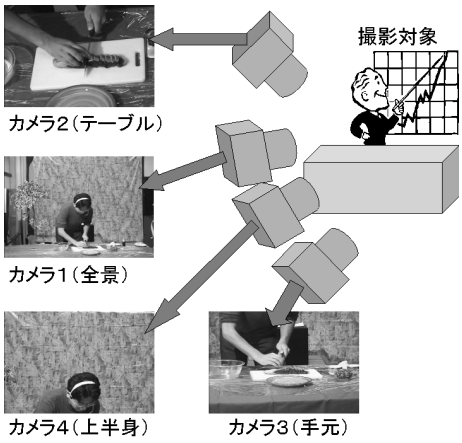


図 8: 料理映像の撮影環境と各カメラの役割

本日 の お 料理 は カツオ の たたき 出す
まず 事前に 引き付けて おいた 鰹を
一千 辛さ に 聞きます
...

図 9: 音声認識で得られたトランスクリプト (誤認識を含む)

撮影には図 8 のように，4 台のカメラを用い，それぞれのカメラの役割は以下のように設定した．

全景 ：作業環境全体を正面から映す

上半身 ：人物の上半身を正面から映す

テーブル ：テーブル上を斜め上方から映す

手元 ：作業を行う手元付近を正面から映す

発話からの音声認識には IBM 社製 ViaVoice を用いて，発話時刻を含むテキストデータを取得する．シナリオは，前節で説明した「かつおのたたき」を用いた．

このような撮影環境で「カツオのたたき」を実際に調理し，得られた映像の例を図 8 中に示す．音声認識で得られたトランスクリプトを図 9 に示す．このトランスクリプトとシナリオとの対応を取って得られた VID スクリプトは図 10 のようになる．かなり音声認識の誤りがあるにもかかわらず，良好な対応結果が得られている．

この VID スクリプトを概念図生成システムに入力して VID を生成した結果が図 11 である．左側の一連の流れが，説明の大まかな順序を表す．また右側の部分は，それぞれの部分に対応するシナリオ中の動作を図示したものである．シナリオには食材や中間生成物，作業などが細かく記述されており，各々に対応するビデオアイコンが配置されている．ここで，作業を示すアイコンには「手元作業」を捉えるカメラからの画像を用い，また物体を表わすアイコンには「テーブル」を捉えるカメラからの画像が用いられている．

```
<scenario>
<definition>
  <object id="o1" type="material">かつお</object>
  <object id="o5" type="material">青ジソの葉</object>
  <object id="o6" type="material">しょうが</object>
  <object id="o7" type="material">だいこんおろし</object>
  <object id="o8" type="material">にんにく</object>
</definition>
<serif>
  <sentence speaker="村山" time="begin:972567724290" id="s3">
  まず、事前に焼き付けておいた<word ref="#o1" id="A">かつお
  </word>を一センチ厚に切ります</sentence>
  <sentence speaker="村山" time="begin:972567834635" id="s8">
  最後に、<word ref="#o5" id="F">青ジソの葉</word>、<word ref
  ="#o6" id="G">しょうが</word>、<word ref="#o7" id="H">だいこ
  んおろし</word>、<word ref="#o8" id="I">にんにく</word>を添
  えていきます</sentence>
</serif>
<event>
  <scene summary="かつおのたたきの調理">
    <action time="type:exist; begin:#s3; end:#s5" agent="村
山" label="皮目を上に切る" type="cook:transform" id="a2">
      <input ref="#A" label="かつお"/>
      <output ref="#B" label="かつおの切り身"/>
    </action>
    <action time="type:exist; begin:#s8; end:#s9" agent="村
山" label="添える" type="cook:add" id="a5">
      <input type="addendum" ref="#F" label="青ジソの葉"/>
      <input type="addendum" ref="#G" label="しょうが"/>
      <input type="addendum" ref="#H" label="大根おろし"/>
      <input type="addendum" ref="#I" label="にんにく"/>
      <output id="c" label="かつおのたたき"/>
    </action>
  </scene>
</event>
</scenario>
```

図 10: VID スクリプト（抜粋）

図 11 では概要がわかりにくいが，発話または手順を示すセグメントのみを別に図化したのが図 12 である．これらを見れば，映像の大まかな流れを掴むことができる．

このように簡単な例については，VID を用いて映像の内容を空間的に表現できた．また，表示する関係の属性を考慮することで，目的に応じた VID を動的に生成できることが示された．しかし，付けられているタグの種類が限定されているなどの制約から，限られた目的・視点以外には対応できないといった問題があり，今後の研究を必要としている．

6 おわりに

本稿では，ビデオアイコンダイアグラム (VID) による映像内容の表現方法を提案した．そのために，映像セグメントの代表画像をアイコンとし，その間の関係を概念図として表現する手法を説明した．また，シナリオに XML タグを付けた VID スクリプトで映像の内容を記述し，映像データとの対応をとることにより VID を生成する手法について紹介した．この実験により得られた VID によって，簡単な例については映像の概略と細かい作業手順を同時に一覧できることを確認した．

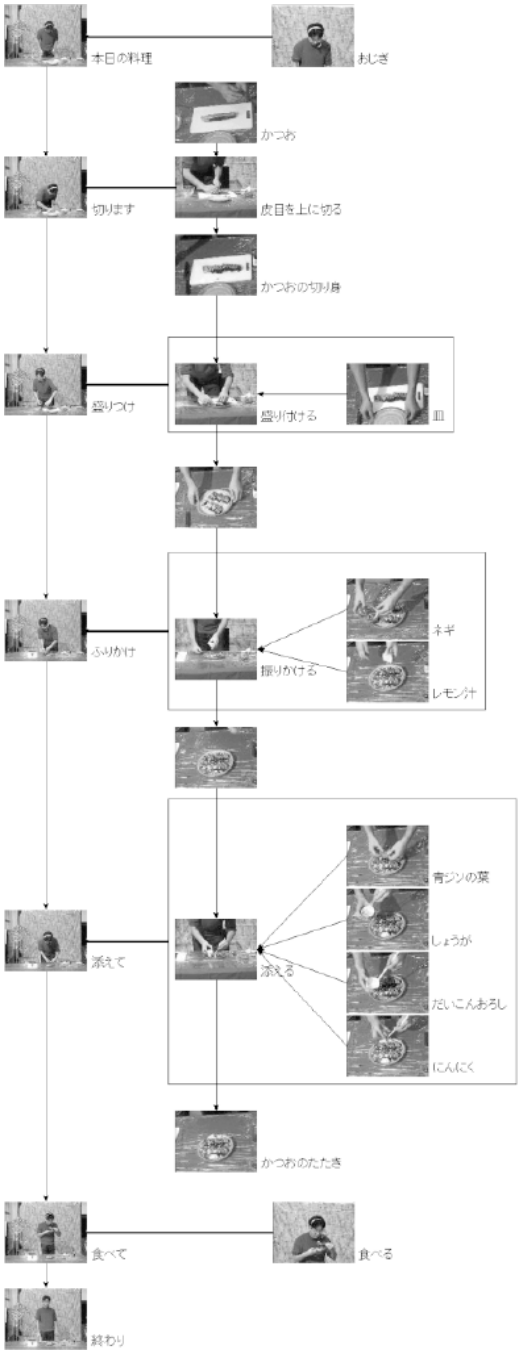


図 11: 料理映像の持つ構造の全体像

今後の課題としては以下のようなことがあげられる．現状では非常に簡単な料理の映像でしか実験を行っておらず，より複雑な料理映像を対象にした実験に取り組むことが必要である．また VID のための XML タグセットについて，妥当性・汎用性を検証していく必要がある．さらに，動作認識を用いることで，シナリオの動作記述を映像と直接対応づけることを検討する予定である．

(a) 発話セグメント　(b) 手順セグメント

図 12: 料理映像の概要

参考文献

[1] Smith, M. and Kanade, T.: Video Skimming and Characterization Through the Combination of Image and Language Understanding Techniques, in *IEEE CVPR*, pp. 775–781 (1997).

[2] 中村裕一, 外村佳伸：見たい部分を簡単に短時間で～気の利いた映像メディア技術を目指して～, 電子情報通信学会誌, Vol. 82, No. 4, pp. 346–353 (1999).

[3] 村山正司, 中村裕一, 大田友一：概念図の自動生成による文書内容の可視化-タグ付き文書からの自動変換-, 第５回知能情報メディアシンポジウム, 電子情報通信学会 (1999).

[4] Clark, J.: XSL Transformations (XSLT) 1.0, http://www.w3.org/TR/xslt/ (1999).

[5] 京都大学言語メディア研：日本語形態素解析システム JUMAN, http://pine.kuee.kyoto-u.ac.jp/nl-resource/juman.html.