2005－DBS－137（Ⅰ）（20）
2005／7／13

3次元地図を用いた自動索引付き映像データベースシステム
－映像データの格納と検索法－

榎 美紀† 増永 良文‡

あらまし 我々は，GPSとジャイロが装着されたビデオカメラを用いて街角で家並みを撮影して，そこから得られた撮影者の位置や姿勢に関するデータ，およびデータベースに格納されている3次元地図を使うことにより，映像のどのフレームからどのフレームまでどのような建物が写っていたかを自動抽出し，映像に建物名をリアルタイムで自動索引付けするシステムを開発してきた．本稿では，さらに研究を進めて，映像データの格納と検索法について得られた結果を報告する．特に，映像データの最適な格納サイズの検証と実験を行った結果，および映像に写しこまれている被写体オブジェクトの連続したフレーム列はAllenの導入した時区間であるという点に着目して体系化した映像問合せ言語，これをユニット論理と名づける，を報告する．

キーワード Multimedia，Video，Indexing，3-D Map，Database，Unit Calculus

An Automatically Indexed Video Database System Using a 3-Dimensional City Map
－Storing and Retrieving Video Data－

Miki ENOKI† and Yoshifumi MASUNAGA‡

Abstract We have been conducting a project for building a video database system where video contents are indexed automatically and in real-time. That is, a video shooter walks on a street being equipped with a GPS and a Gyro sensor so that the sequence of video frames is collected along with the shooter's position and the camera's posture data. These data are processed using a "3-dimensional" city map so that the name of the buildings captured in each frame of a video clip is created as its index in real-time. In addition, this paper reports an investigation result on the most suitable size for storing video clips in a video database, and introduces a video query language named the unit calculus which is designed based on Allen's time interval logic.

Keyword Multimedia，Video，Indexing，3-D Map，Database，Unit Calculus

1. はじめに

これまで，映像データに対し，画像理解，被写体オブジェクト抽出･追跡，音声認識等，さまざまな技術を使った処理法が研究されている[1,2].

我々は，GPSとジャイロをビデオカメラに装着し，3次元地図を使うことにより，撮影された映像のどのフレームからどのフレームまでどのような建物が写っていたかを自動抽出し，映像に建物名をリアルタイムで自動索引付けするシステムを開発してきた[3,4,5].

しかし，撮影された映像をどのようにデータベ

† お茶の水女子大学大学院 人間文化研究科博士前期課程 数理・情報科学専攻 miki@db.is.ocha.ac.jp
Graduate Division of Mathematics and Computer Science (Master's Program), Ochanomizu University
‡ お茶の水女子大学 理学部情報科学科 masunaga@is.ocha.ac.jp
Department of Information Science, Faculty of Science, Ochanomizu University

ースに格納するか，あるいはそのように自動索引付けされた映像データをどのように検索するかの体系については,今後の課題として残されてきた．

そこで，本研究では，まず撮影されたビデオクリップを映像データベースに最適に格納するための，格納サイズの検証と実験を行う．続いて，映像に写しこまれている被写体オブジェクトの連続したフレーム列をユニット(unit)と定義し，これは Allen が導入した時区間であるという特徴に着目して，映像問合せ言語，これをユニット論理(unit calculus)と名付ける，を提案する．これにより，例えば「銀座三越が 10 秒以上映っている映像が欲しい」といった検索要求に答えられるようになる．

2. 被写体建物オブジェクト自動索引付け検索システムの概要

ビデオカメラの撮影者はビデオ撮影者の位置と時刻を取得するために GPS を身につけ，ビデオカメラに撮影者の姿勢を知るためにジャイロセンサを取り付けて撮影する．また，ウェアラブルコンピュータを用いてこれらのデータとそこに格納されている 3 次元地図データを総合的に処理し映像データに被写体建物オブジェクトの自動索引付けをリアルタイムで行い，それを利用した映像検索システムを実現する．図 1 は我々が開発しているシステムの全体像を表している．

映像自動索引部では，撮影者の GPS データ，ジャイロデータを取得し（建物名称を取得するための）2 次元, 3 次元地図を用いてリアルタイムに被写体建物オブジェクト抽出と索引付けを行う．それにより INDEX DB の XBuilding テーブルが作成される．

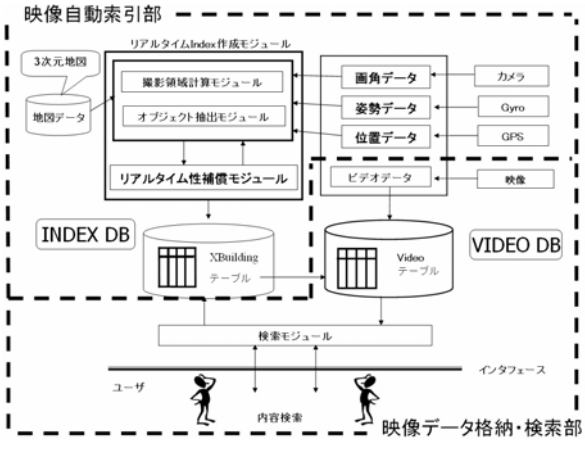

図1 3次元地図を用いた被写体建物オブジェクトの索引付け，検索システムの概念

本研究で実装する映像データ格納･検索部では，撮影した映像データがビデオクリップとして VIDEO DB の Video テーブルに格納され検索に供される．

3. 映像データの格納

3.1. 映像データ格納法

圧縮した映像データをリレーショナルデータベースに格納していく．格納方法には，以下の 2 つの方法が考えられる．

①映像格納場所であるファイルパスのみを格納
②映像そのものを格納

まず，①の場合，データベースには映像格納場所の文字列しか格納しないため，データベースの容量をそれほど取らなくてもすむ利点があるが，ユーザがデータベースサーバにアクセスしてもファイルパスのみしか得ることが出来ない．一方，②の場合，映像をデータベースに格納するため，データベースの容量は相当なものになるが，サーバに接続すれば誰でもアクセスして映像を閲覧，取得可能であり，映像編集中にシステム障害などに合った場合障害時回復の対象になる．本研究では，このような理由から，映像データの共有性やデータベースの一貫性を考えて②の手法を用いる．

また，映像データは非圧縮時で 1 時間分のデータの場合 40～50GByte もの容量を必要とする．そこで圧縮技術として MPEG-1 を用いてデータ圧縮を行うことにより,データは約 700MByte となる．この圧縮を行うと,15 フレーム（1 フレームは 1/30 秒ごとに撮られる）を 1GOP (Group Of Pictures)として圧縮されるので，映像の制御を行う際は GOP の整数倍単位での制御が好ましい．

野中らの研究[6]では，オブジェクト指向データベースシステムに GOP 単位で映像を格納しているが，現在リレーショナルデータベースでは画像や動画などをバイナリデータとして格納できる可変長データ型の BLOB(Binary Large OBject)型を持つので，本研究ではリレーショナルデータベースシステムを用いて格納する．この BLOB 型 1 カラムへは最大 4GByte までのデータが格納可能である．

3.2. Video テーブルの作成

映像データ（＝ビデオクリップ）を格納するため Video テーブルを定義する．このテーブルのカラムには図 2 に示すように映像の取り出しに必要となる VideoID，格納された映像そのものを示す V_file，映像のファイル名となる V_name を設定す

る．また，長時間の映像は等時間長に分割され，Partition_Num が 1 から順に付与される．

VideoID	V_file（BLOB型）	V_name	Partition_Num
040709	“ginza1.mpg”の映像	ginza1.mpg	1
040709	“ginza2.mpg”の映像	ginza2.mpg	2
040709	“ginza3.mpg”の映像	ginza3.mpg	3
…	…	…	…

図2 Video テーブル

3.3. 映像の格納サイズの検証

3.3.1. 再生待ち時間コスト

映像の格納はユニットの再生と密接に関係する．例えば再生したいユニットが 1 時間の映像データの 46 分目から 5 分間であったとする．もし映像を分割しないで，1 時間のビデオをそのまま BLOB データとして格納した場合，不要部分の切り出し処理のために相当の待ち時間を要することが想定される．

そこで，映像を d 秒ごとに分割して格納することによりその短縮が可能かを再生待ち時間のコストで検証する．そのため映像を d1, d2 秒(d1<d2)ごとに分割するとし，1 ユニットの時間長を u 秒であるとする．

(1)SQL 発行回数によるコスト

$\left\lceil \frac{u}{d1} \right\rceil$　$\left\lceil \frac{u}{d2} \right\rceil$

となり，1 分割サイズが大となるほど SQL 発行回数は減る．d1, d2 の比率を，

$\left\lceil \frac{u}{d1} \right\rceil \div \left\lceil \frac{u}{d2} \right\rceil$　$\frac{d2}{d1}$　とし，これを x とおく．

(2)ファイルの fetch コスト

fetch のコストは(1)とは逆に，分割サイズが大となるほどユニット外の部分も fetch してしまうことになる．よって(1)の逆数の $\frac{1}{x}$ となる．

したがって，再生待ち時間コストは(1) + (2)で，次のように定式化される．

$$\cos t = ax + \frac{b}{x}$$

つまり，図 3 に示すグラフの最小点が定性的なコスト最小値を示す．

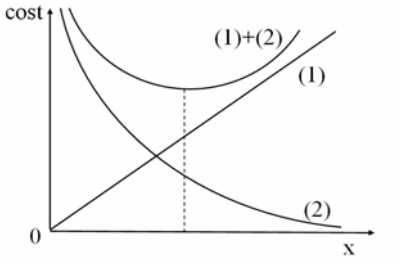

図3 再生待ち時間コスト

3.3.2. 格納サイズ

映像を分割するにあたり，GOP の整数倍をグループにして，それを一つの分割として格納し，ユニット単位で再生を行う．そこで，コスト最小となる分割サイズを定量的に検証する．

(1)分割による再生待ち時間の実験

この分割サイズを決定するため，映像ファイルを 3 秒, 15 秒, 30 秒ごとに分割して格納したものと，6 分 52 秒のビデオ 1 本分をそのまま格納した時の 130 秒目からの映像の再生待ち時間の比較を行った．実験環境は以下のとおりである：

Server：Windows2000　(HDD:90G)
データベースシステム：Oracle 9i（JDBC）
インタフェース：Java サーブレット

結果は，3 秒分割時の再生待ち時間が 10 秒，15 秒分割時の再生待ち時間が 9 秒，30 秒分割時の再生待ち時間が 8 秒，ビデオ 1 本時の再生待ち時間が 14 秒で図 4 に示すようになり，3.3.1 で検証したコスト式の最小値を実現するのは d2＝6 分 52 秒とすると，d1＝30 秒の時となる．

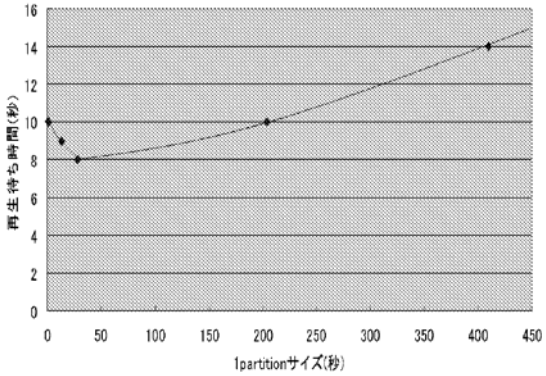

図4 再生待ち時間比較

(2)1 ユニットの時間長からの検証

ユニットは一つの建物が写し込まれている単位であるので，ビデオクリップの（格納のための）分割サイズを，ユニットの時間長データの分布に基づき検証する．人間がオブジェクトを「写って

いる」と認識できるのは3秒以上同じオブジェクトを見たときとされているため[7]，3秒以上の時間長のユニットを用いる．

図5にユニット時間長分布データを示す．1ユニットあたり5秒程度のものが多く，また大きい建物は写しこまれる時間が長く 30 秒前後に集中していることが観測できた．

以上，(1), (2)の結果より，再生待ち時間が最小となる分割サイズは 30 秒付近に存在することが明らかとなったので，実装ではビデオクリップを30秒ごとに分割して格納することとした．

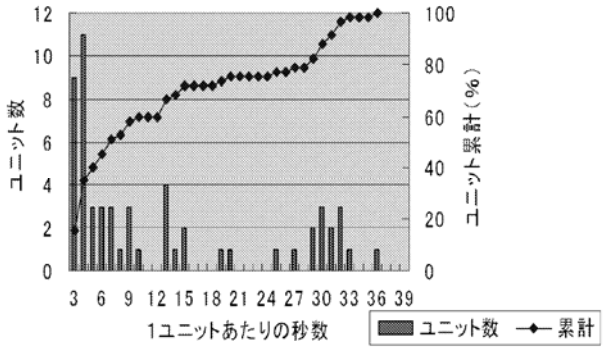

図5　ユニット時間長分布データ

4. ユニット論理の導入と映像データの検索

4.1. ユニット

図1の映像自動索引部で抽出された被写体建物オブジェクトのデータは XBuilding テーブルに格納される．図6に示すように，XBuilding テーブルの属性は映像番号（VideoID），建物番号(BuildingID)，建物名(BuildingName)，開始フレーム番号(Fs)，終了フレーム番号(Fe)，ユニット番号(UnitID)である．

VideoID	BuildingID	BuildingName	Fs	Fe	UnitID
041105	**1**	Ginza-Matsuya	**1**	2100	**1**
041105	**2**	SONY	**1**	7800	**2**
041105	**1**	Ginza-Matsuya	5800	24000	**3**
…	…	…	…	…	…

図6　Xbiuilding テーブル

撮影された映像の中で，$u_{v,o,i}$は建物オブジェクト（oとする）が，ビデオクリップ（vとする）のあるフレーム（bとする）から始まり，あるフレーム（eとする）まで連続して写しこまれている，第i番目の部分とする（i≧1）．このビデオフレームの連続を$u_{v,o,i}$=(v, o, i, b, e)で表し，ユニット（unit）と呼ぶ．ビデオクリップvの中に存在する全てのユニットのなす集合をU_Vと記す．

図7に被写体建物オブジェクトとユニットの関係を示す．例では，建物O_1とそれが連続して写っているユニットの対がINDEXデータベースのXBuildingテーブルに記録されるので,少なくとも(O_1, $u_{v,o,1}$)と(O_1, $u_{v,o,2}$)の２つのタップルが存在する.その結果,建物IDで問い合わせると，その建物が写っているユニット全てを知ることができる．

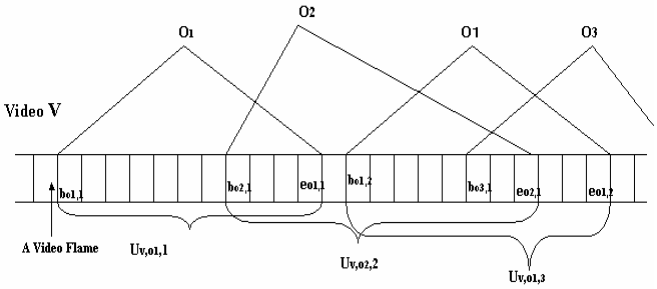

図7　被写体建物オブジェクトとユニットの関係

4.2. ユニットのオブジェクト指向表現

検索の対象となるユニットは連続したフレーム列なので，それは Allen の時区間[8]であると捉えることができる．さらにユニットは，ビデオ映像であり，ビデオクリップ構成要素であることを考慮すると,図8に示すような TimeInterval, Video, VideoClip, Unit のクラス階層が成り立つ．

TimeInterval クラスは開始時刻(start time point: stp)，終了時刻(end time point: etp)の属性を持つ．また，etp –stp でその時区間の時間長(duration)を計算できるので，それをメッソド（method）として定義する．stp(), etp()はそれぞれ stp 値と etp 値を返すメソッドとする．

Video クラスは再生を行う playback(), 映像の開始フレーム番号(start frame number: sfn)と終了フレーム番号(end frame number: efn)を返す sfn()と efn()というメソッドを持つ．

VideoClip クラスは撮影された映像そのものを表すクラスで，ビデオクリップを識別する vcid(Video Clip ID)と，撮影された日時(date)を属性に持ち,ビデオクリップの ID 番号を返す vcid(), 日時を返す date()といったメソッドを持つ．また，ビデオクリップはユニットを導出するので Unit

クラスのオブジェクト群を指す属性 derives と，derives()というメソッドを持つ．

Unit クラスは，ユニット独自の性質として，被写体建物オブジェクトの ID を表す bid (BuildingID)を属性に持つ．ユニットに一貫して写っている建物の ID を返す bid()，ユニットのビデオクリップを返す vc()，そのユニットの被写体建物オブジェクト（仮に A とする）が，ビデオクリップ内に写っている A の何番目の出現であるかを表現する sequence()をメソッドとして定義する．

VideoClip クラスは図 1 における Video テーブルに，Unit クラスは図 1 における Xbuilding テーブルに該当し，それぞれデータベースに格納されている．

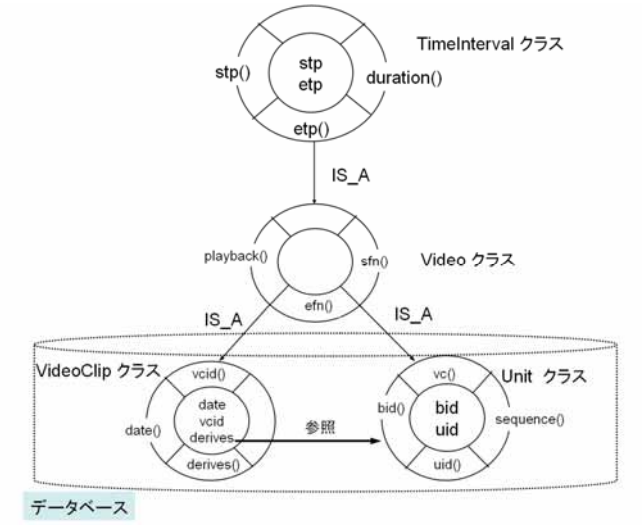

図 8 ユニットの クラス階層図

4.3. Allen の時区間論理

2 つの時区間の間に存在する時間的関連は，Allen の時区間論理が示しているように，図 9 に示す 13 種である．しかし，例えば”X before Y”であることと”Y after X”であることは同値なので，{before, after}，{meets, met-by}，{overlaps, overlapped-by}，{during, contains}，{starts, started-by}，{finishes, finished-by}の対については，それぞれ{before, meets, overlaps, during, starts, finishes}を代表元として使用してかまわない．

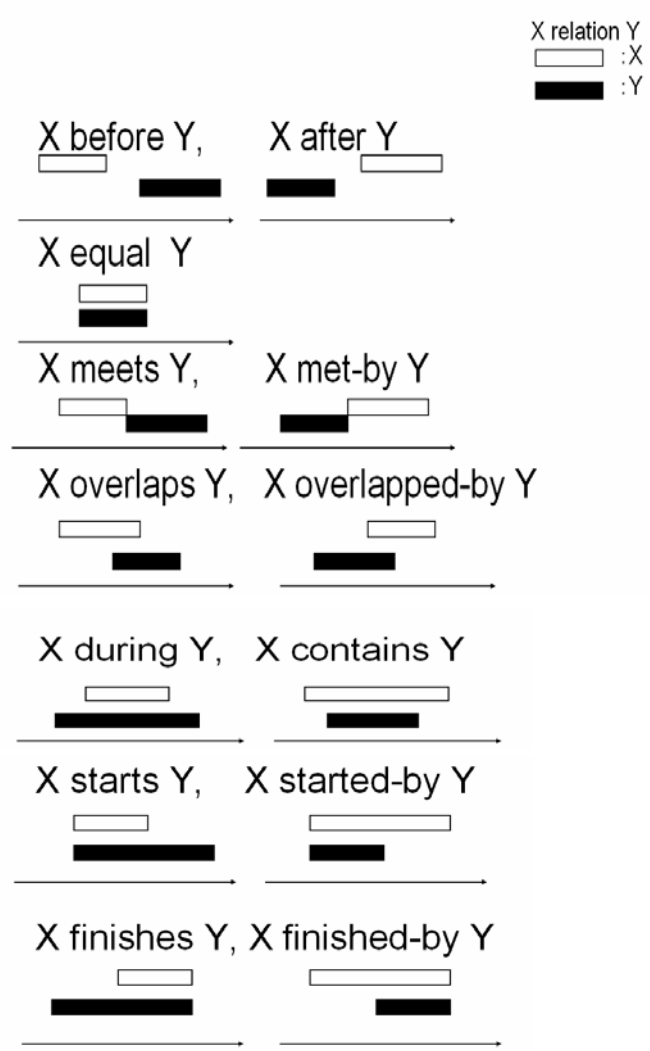

図 9 Allen の時区間の 13 種の時間的関連

4.4. Allen の時区間論理の拡張

Allen の時区間論理では，例えば時区間 X と Y が X before Y の関係にあるとしても，X が Y の何秒前に終了していたのか，ということは表現できない．しかしながら，映像の表現においては，このような時間関連を直接表現できる関連性を定義しておいたほうが使いやすい．

そこで，我々は Allen の時区間論理を図 10 に示されるように拡張する．例えば，X before(=，δ) Y は X が終了して，丁度δ秒後に Y が生起する関連を表す．

以降，本論文では，時区間論理と言う場合には，（特に，断りのない限り）拡張された時区間論理を指すこととする．

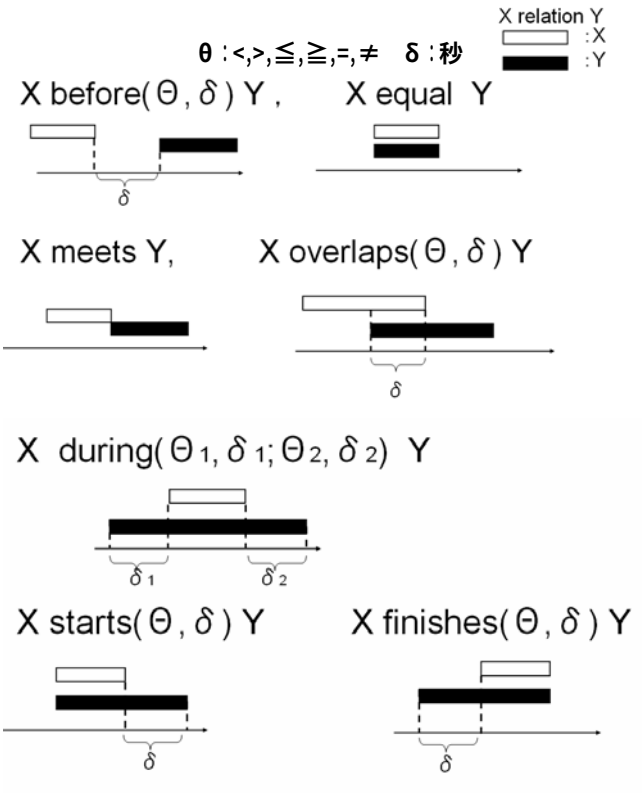

図 10 Allen の時区間論理の拡張

4.5. ユニット論理の提案と基礎的考察

本節では，映像をユニット単位で検索するための問合せ表現の体系として，ユニット論理(unit calculus)を提案し，基礎的考察を加える．

X をユニット（を表す）変数とし，P(X)を X のみを自由変数とする式(formula)とするとき，{X| P(X)} をユニット論理表現(unit calculus expression)という．この表現は映像データベースに対する問合せ表現である.図 8 に示したように，ユニットはビデオであり,ビデオは時区間なので，ユニットの検索論理は階層性を持つ．以下，この階層のレベルに合わせて，代表的な問合せ表現を考察する．

（1）時区間レベルの問合せ表現

このレベルの問合せは,ユニットを”閉”時区間とみなし,時区間の性質のみを使った問合せである．この問合せ表現には，次の（構成）要素を使える．

(a) 関数 stp()，etp()，duration()，比較演算子 θ（<,>, ≦,≧,=,≠），時間長δ(または δ1，δ2)，時刻 τ．

(b) 図 10 に示されたユニット同士の時間的関連．

【問合せ表現例 1】

Q: 時間長が 30 秒以上のユニット X を求める．

E: {X | (∃v)(U_v⊇X ∧ X.duration()≧30)}

ここに，Q は問合せを，E はユニット論理表現を，v はビデオクリップを表す．

【問合せ表現例 2】

Q:ユニット Y と before の関係にあり，その間隔が 25 秒以上のユニット X を求める．

E:{X | (∃v)(U_v⊇X ∧ X before(≧,25) Y)}

（2）ビデオレベルの問合せ表現

ユニットをビデオとみなした時の問合せ体系である．この時，次の（構成）要素を使える．

関数 playback()，sfn()，efn()，比較演算子 θ（<,>,≦,≧,=,≠），定値(constant) c.

【問合せ表現例 3】

Q:開始フレーム番号が 3000 番以降のユニット X を求める ．

E:{X | (∃v)(U_v⊇X ∧ X.sfn()≧3000)}

（3）ユニットレベルの問合せ表現

この時，次の（構成）要素を使える．

関数 bid(), vc(), sequence(), 比較演算子 θ（<,>, ≦,≧,=,≠），定値 c.

【問合せ表現例 4】

Q:SONY ビル(BildingID=12)が写っているユニットを求める．

E:{X | (∃v)(U_v⊇X ∧ X.bid()=12)}

（4）混合問合せ表現例

以上に示したレベルの質問を混合した問合せ表現．

【問合せ表現例 5】

Q:SONY ビルが写っているユニット(Y)と 10 秒以上 Overlap し，時間長が 50 秒以下のユニット X を求める．

E:{X | (∃v)(U_v⊇X ∧ X overlaps(≧,10) Y ∧ Y.bid()=12 ∧ X.duration()≦50)}

5. まとめと今後の課題

本論文では，索引付けされた映像データの格納において,その最適な格納サイズの検証を定性的,定量的に行い，格納サイズを決定した．また，検索の対象となるユニットは時区間であるという点に着目し，映像問合せ表現の体系として時区間論理を拡張したユニット論理を提案し，代表的な問合せ表現を考察した．

今後の課題として，ユニット論理の厳密な定義と，検索対象をユニットのみならずビデオクリッ

プをも包含するように拡張した問合せの体系化を目指すこと，およびリアルタイム索引付けを含む全ての機能をウェアラブル環境で実装したシステムの完成を目指すことが挙げられる．

文　献

[1] Vaughan, G., Smeaton, A., Gurrin, C., Lee, H., and McDonald, K.: “Design, Implementation and Testing of an Interactive Video Retrieval System, ” Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval, pp.23-30, November 2003.

[2] Wang, Y., Ostermann, J. and Zhang, Y-Q.: “Video Processing and Communications , ” (book) Prentice Hall, 2002.

[3] 石黒玲，佐藤有紀子，増永良文“3 次元地図を用いたビデオコンテンツの自動索引法－被写体建物オブジェクトの自動抽出－”，情報処理学会研究報告（DBWS2003），Vol2003，№133-55，2003 年 7 月．

[4] 佐藤有紀子，石黒玲，増永良文“3 次元地図を用いたディジタルビデオコンテンツの自動索引法の提案と検証”，日本データベース学会（DBSJ Letters），Vol.3, No.1, pp.149-152, 2004 年 6 月．

[5] Yukiko Sato and Yoshifumi Masunaga: ”A Novel Indexing Method for Digital Video Contents using a 3-Dimentional City Map ”, Proceedings of the 4th International Workshop on Web and Wireless Geographical Information Systems (W2GIS) , pp.333-343 , Springer, November 2004.

[6] 野中和明，増永良文“MPEG-1 動画像データベースシステムのプロトタイプ実装”，情報処理学会第 53 回全国大会，7R-7，vol.3, pp.89-91, 1996.9

[7] HonJian Zhang, Chien Yong Low, Stephen W.Smoilar, JianHua Wu:” Video Parsing, Retrieval and Browsing” , Intelligent Multimedia Information Retrieval , ed. Mark T.Maybury, PP.139-158, MIT Press, Massachusetts, 1997.

[8] J. Allen: “ Maintaining Knowledge about Temporal Intervals ”, Communications of the ACM, Vol.26, No.11, pp.832-843, November 1983.