

仮想壁による検索インタフェースを備えた 分散型映像データ管理システム

王 軼 群[†] 土方 嘉 徳[†] 西 田 正 吾[†]

ネットワークに接続可能な監視カメラが増えつつある。本研究では、このような分散するカメラの映像データを効率よく管理し、ユーザにとって分かりやすいインタフェースで検索が可能なシステムを提案する。映像データの管理は、映像データ本体をカメラ側のローカルの PC に保管し、その撮影情報だけをサーバで時空間データ索引付けアルゴリズムにより管理することとする。さらに、データの通信を HTTP で行い、検索を Web ブラウザから行えるようにする。検索インタフェースでは、仮想壁という概念を導入することで、カメラの位置を気にする必要のない検索キー入力方式を実現し、さらに検索に適合した映像を空間的にまたは時間的に絞り込みが可能な出力画面を実現する。

Distributed Video Data Management System with Search Interface based on Virtual Wall

YIQUN WANG,[†] YOSHINORI HIJIKATA[†] and SHOGO NISHIDA[†]

Monitoring cameras with the capability of connecting to the Internet are increasing. This paper proposes a system which manages the distributed video data efficiently and has an interface which can be operated easily. For managing video data, the raw video data is stored in a local PC which directly connects to the camera and only the recording information is managed in the server by using a spatio-temporal indexing algorithm. The data communication is conducted through HTTP and the search input and output is realized in Web browser. For the search interface, we propose an input method of search key which does not insist users to care the locations of cameras by introducing the idea of virtual wall. We also provide a search result interface which allows users to narrow down the video data in the search result.

1. はじめに

近年、監視カメラが日常生活にも普及し始めている。例えば、建物の入り口や駅構内、道路脇など、多くの場所で監視カメラが見られるようになってきている。これらの監視カメラは、主に防犯の目的で設置されているが、他にも防災やマーケティングなどの観点からも、これら監視カメラの映像を解析し、利用することが考えられる。例えば防災では、災害発生後に、災害要因を検討したり、被災状況の変化の様子を解析したりすることが考えられる。また、マーケティングでは、人の流れを解析したり、その流れにおける人種（サラリーマンが多いのか、子供が多いのかなど）を調査したりすることが考えられる。このような解析を行うには、これら監視映像データを一つずつ人手で調べていく必要がある。将来、ネットワークカメラが普及し、

接続可能なカメラが増加してくると、大量の映像データを取り扱うこととなり、この作業を手で行うのは困難になってくる。そのため、計算機による検索機能が必要となってくる。

本研究では、監視カメラのように分散環境にある映像データを、映像を特定するのに重要となる時間情報と空間情報に基づき、Web ブラウザから GUI (graphical user interface) で検索が可能なシステムを提案する。GUI での検索には仮想壁 (virtual wall) という概念を導入し、カメラの位置や方向を意識しない検索を可能とする。映像データの管理は、映像データ本体と映像の撮影情報を別に管理し、通信と検索インタフェースを含めて Web 環境で実現する。この 2 点が本研究の技術的な新規性となる。

まず、検索インタフェースに関する研究動機について述べる。映像に自動で撮影情報を付加した場合、センサが備わっているのはカメラと考えられるため、映像には撮影対象の物体の位置情報が付加されているのではなく、カメラの位置情報が付加されている。一般

[†] 大阪大学大学院基礎工学研究科
Graduate School of Engineering Science, Osaka
University

に検索には範囲検索があるが、街角のカメラなど比較的広範囲の対象を撮影している場合、検索結果として検索範囲内のカメラが撮影した検索範囲外の映像を返すことが有り得る。このようなずれを考慮してユーザに検索範囲とカメラの方向を指定させるのは、ユーザにとって大きな負荷となる。

そこで、都市の中に仮想的な壁を想定し、その壁のある方向から眺めた時に見られる情景を画面に映し出す検索キー入力方式とインタフェースを提案する。このような検索ができるようになると、例えば「ある建物の裏側はどう見えるのか?」や「町を海側から見ると、災害状況はどうなっているのか?」などの要求をそのまま入力することができる。以降では、この仮想的な壁のことを仮想壁と呼ぶ。また本インタフェースでは、GISの地図データを画面に表示し、ユーザはその地図上でマウスを使って仮想壁を入力できるようにする。さらに、検索結果は一度にすべての映像を表示するのではなく、最初は概観の分かる遠くからの映像だけを検索し、ユーザのインタフェース上の操作に応じて、インタラクティブに詳細を写している近くからの映像も検索し、表示させることとする。

次に、分散型映像データの Web 環境での管理に関する研究動機を述べる。データを検索するためには、そのデータを一箇所で集中管理することが効率的である。しかし、リアルタイムで撮影しているカメラの映像データを一箇所のサーバに送ると、そのサーバの通信容量を簡単にオーバーしてしまう。本研究では、映像データ本体をローカルなカメラで管理し、その撮影情報だけをサーバで索引付けするが、データの通信には HTTP を使い、検索を Web ブラウザから行えるようにする。具体的には、仮想壁を GUI でインタラクティブに入力するため、検索キーの入力を Web ブラウザ上の Java アプレットから行えるようにする。Web ブラウザで検索結果を閲覧できるように、サーバにて検索結果の HTML メッセージを作成し返す。この際、HTML メッセージには、ローカルなカメラの映像データの URL のみを挿入する。また、インタラクティブに映像を絞り込むためのボタンも挿入する。このようにすることで、検索を行うクライアントには、専用のアプリケーションソフトをインストールする必要がなく、サーバがある場所まで行かなくても、映像を検索できる。さらに、普段慣れ親しんでいる Web ブラウザと標準でプラグインされた映像表示ソフトを利用できる。また、検索にかかる時間コストを最小限にするため、映像情報は時空間データ索引付けアルゴリズムで索引付けする。これにより、大規模な映像データを

扱ったとしても、その検索時間の悪化の程度を軽減することが可能である。

本稿では、2 章で本研究の対象とする映像を利用する状況と仮想壁実現のための課題について詳しく述べる。3 章でシステムの構成について述べる。4 章では、撮影情報の管理手法と検索インタフェースの設計について説明し、さらに実装したシステムの画面例を示す。5 章でシステムの評価を行う。6 章で関連研究について述べ、最後に 7 章でまとめを述べる。

2. 映像利用の目的と仮想壁実現のための課題

本章では、本システムが想定するユーザが映像を利用する目的について述べ、仮想壁を実現するための課題について述べる。

2.1 映像利用の目的

本システムは、ユーザが時間情報と空間情報から映像データを検索することを支援するものである。一般にユーザが監視カメラの映像を利用する目的を大きく分けると、問題の発見と問題の解析の 2 種類が考えられる。後者は、何か問題が発生した後に、問題が起こったことによる影響を調査するために、またはその問題の起こった要因を調査するために、関連しそうな映像を確認することで行う。本研究では、後者の目的を支援することを目指す。以下に、どのように映像を利用するかを具体例を挙げる。影響の調査では、問題が発生した後に、問題の通報があった場所付近の映像やセンサなどが感知した場所付近の映像、学校や病院などの重要な建物がある場所の映像などを調べることが挙げられる。さらに、問題の発生した時刻から一定時間ごとに、それらの映像の変化を調べることも考えられる。また、問題の要因の調査では、問題の通報があった場所付近の映像を調べ、さらに問題が起こった数分前、数時間前、数日前といった映像も確かめてみたりすることが考えられる。本研究では、ユーザが問題の解析を目的として映像データを利用する状況を対象とし、ユーザが上述のように何度も映像の検索を繰り返し、状況を確認するような状況を対象とする。

2.2 仮想壁実現のための課題

提案する検索キーを実現するには、ユーザの入力する検索キーと時空間データ索引付けアルゴリズム(以降、時空間 DB(spatio-temporal DB))の入力となる検索キーの違いが問題となる。ユーザからみた検索キーは、ある時間帯(t_1, t_2)にある仮想壁(x_s, y_s, x_e, y_e)をある方向 d から眺めるというものである。しかし、映像データの URL を管理している時空間 DB の検索キーは、最も汎用性の高い範囲検索の場合、ある時間

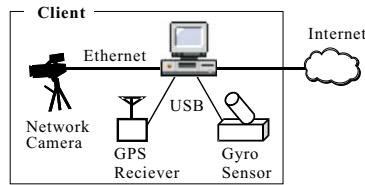


図 1 想定するネットワークカメラ
Fig.1 Network camera for our system

帯 (t_1, t_2) にある空間範囲 (x_1, y_1, x_2, y_2) に存在するオブジェクトを求めることになる。したがって、ユーザと時空間 DB の検索キーは異なり、変換する仕組みが必要となる。そこで本研究では、既存の時空間 DB を用いて方向 d も索引付けすることとし、ユーザからみた検索キーを時空間 DB の範囲検索の検索キーに変換することとする。具体的な時空間 DB を使ったデータ管理手法と検索キーの変換方法については、システム全体の説明を行った後に、説明する。

3. システム構成

本章では、本システムのカメラ側のハードウェア構成を述べた後、システム全体の概要、システムのエージェント構成について述べる。

3.1 カメラ側のハードウェア構成

本システムでは、映像は一定時間おきにファイル分割されていることを想定する。本研究が想定しているネットワークカメラとは、図 1 のようなものである。市販のネットワークカメラを、イーサネットでローカルの PC (以降ローカル PC(local PC)) に接続する。そのローカル PC がネットワークカメラに一定時間ごとに映像を要求し、配信される映像を 1 つのファイルに保存することでファイル分割を行う。ローカル PC には内蔵時計が備えてあり、また GPS とジャイロセンサも USB など接続してある。ローカル PC は、ファイル分割のタイミングでそれらのデータを取得し、保存する。保存した映像データの URL と、空間情報、時間情報、カメラ方向は、サーバに HTTP で送信される。映像データ本体はローカル PC のハードディスクに格納される。ローカル PC は Web ブラウザを兼ねており、HTTP で他からアクセスがあれば、対応する映像ファイルを返すものとする。

3.2 システムの概要

システムの概要を図 2 に示す。カメラ側のローカル PC は、その受け持ちのカメラが撮影した撮影情報 (空間情報・時間情報・カメラ方向) をサーバに送る。これらの撮影情報を、サーバ側で時空間 DB で管理する。ユーザは、Web ブラウザを介して検索キーを入力

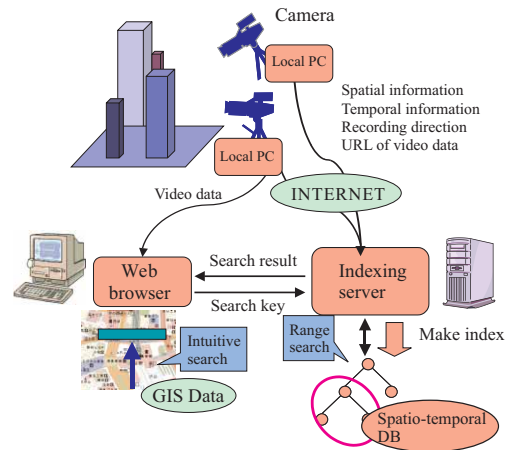


図 2 システム概要
Fig.2 System structure

し、その検索結果の映像を閲覧する。Web ブラウザ上に表示された GIS の地図データ上で、ユーザはマウスを使って検索キーを入力する。具体的には、マウスで仮想壁とどの方向から見るかを描画する。また、時間範囲はキーボードから直接入力する。検索結果の映像は、別のウィンドウを立ち上げて、そこに表示させる。サーバ側では、受け取った検索キーを、時空間 DB の検索キーの形式に変換し、時空間 DB を検索する。サーバは検索された映像データの URL を検索結果のページに埋め込み、Web ブラウザに返す。Web ブラウザは検索結果表示後に対象の映像データをカメラ側のローカル PC からダウンロードする。ユーザが、検索結果のページ上で、映像絞り込みの操作を行えば、検索キーを書き換えて、再度サーバ側に検索をかける。

4. データ管理構造と検索インターフェースの設計

本章では、本システムの主要となるモジュールである時空間データの管理構造と検索キーの変換方法、検索結果の表示方法について述べる。

4.1 データ管理構造

本システムにおける映像データの管理には、既存の時空間 DB を利用する。また、カメラの方向を 8 方向に離散化して、方向ごとに時空間 DB を構築する。ここで 8 方向としたのは、時間範囲や空間範囲に比べて方向に関するユーザの指定には、それほど厳密さはないと思われることと、カメラの位置が仮想壁から離れるほど、ユーザが予想する撮影範囲と実際の撮影範囲にずれが生じるため、方向の厳密さが検索結果に反映されなくなるためである。入力は、時間範囲 (t_1, t_2) 、

空間範囲 (x_1, y_1, x_2, y_2) 、カメラ方向 (d) で、出力は映像の URL である。入力のうち、空間範囲はオペレータの検索キーのうちの仮想壁 (x_s, y_s, x_e, y_e) と閲覧方向 d の 2 つから変換されたものである（変換方法は、4.2.2 節で説明する）。検索時には、カメラ方向 d に対応する一つの時空間 DB を特定し、それを探索する。今回の実装では、動的データが扱え、しかも比較的に高速検索が可能な 3D R-Tree²⁾ を用いることにする。

4.2 検索キーの変換方法と適合映像の表示方法

本システムにおける検索結果の表示インタフェースは、最初に遠くからの映像を表示し、ユーザの絞り込みの操作により、より近くの映像を表示する。また、検索も一度に可能性のある映像全てを検索するのではなく、ユーザの操作によりインタラクティブに検索キーを生成している。ここでは、映像の表示インタフェースと検索キーの変換方法について述べる。

4.2.1 適合映像の表示方法

ユーザは、検索に適合した映像を一つずつ確認する。この時、ユーザはある程度の数の映像を一度に見られることと、そこから映像を絞り込むことができることが重要と考える。また、映像の絞り込みには、空間的に絞り込むことと時間的に絞り込むことの二つが要求されると考えられる。そこで、本システムの検索結果の表示には、図 3 に示すような表示方法を採用する。映像表示の基本として、画面を縦に M 個、横に N 個に分割して、ここに適合した映像を表示していく。分割の個数は、ユーザが一覧するのが不可能にならない程度の数（数個程度）とする。検索キーに適合する映像が一度に表示しきれないほど多い場合は、画面左上の [More] ボタンを押すことで、表示し切れなかった映像を表示する。本表示方法には、二つの表示モードがある。一つを空間整列モード (spatial-ordering mode) と呼び、もう一つを時間整列モード (temporal-ordering mode) と呼ぶ。各モードでは、映像の並べ方が異なっている。画面上部のラジオボタンで、それぞれのモードを切り替えることができる。

空間整列モードでは、 $M \times N$ の配置において、縦方向には意味はないが、横方向は仮想壁に平行な方向に対応する。つまり、 n 番目の列に表示される映像は、データ管理構造への空間検索範囲を仮想壁に垂直に分割した時の、 n 番目の領域に入るカメラからのものである。例えば、検索結果の表示画面の左にある映像は、実空間においても仮想壁の左側を写している映像となる。また、時間を表すインディケータがある。このインディケータ全体は、ユーザの検索時間範囲を表しており、そのうち色のついている時間帯の映像を画

面に表示している。インディケータの上についているラジオボタンで、表示する時間帯を変更できる。また、画面の各列の下にはボタン [>>] が付いている。このボタンを押すと、この列に対応する空間範囲に検索キーを絞り込み、画面全体を使ってこの範囲にあるカメラの映像を再表示する。ウィンドウの一番下にある [Forward] ボタンを押すと、今よりも近くから撮影している、詳細の分かる映像を表示する。逆に、[Back] ボタンを押すと、今よりも遠くから撮影している、概要の分かる映像を表示する。

時間整列モードでは、 $M \times N$ の配置において、縦方向には意味はないが、横方向は時間範囲に対応する。つまり、 n 番目の列に表示される映像は、検索時間範囲を N 個に分割した時の、 n 番目の時間帯を表示している。映像の下には、時間を表すインディケータがある。このインディケータ全体は、ユーザの検索時間範囲を表している。インディケータの各時間範囲の下には、ボタン [>>] があり、このボタンを押すと、この列に対応する時間範囲に検索キーを絞り込み、画面全体を使ってこの時間範囲にある映像を再表示する。また、インディケータの両端には、[] ボタンと [] ボタンがついており、それぞれ今の検索キーの検索時間範囲を、前にずらすか後ろにずらすかすることができる。

4.2.2 検索キーの変換方法

ユーザの検索キーのうち、仮想壁 (x_s, y_s, x_e, y_e) とカメラ方向 d から、データ管理構造用検索キーの空間範囲 (x_1, y_1, x_2, y_2) に変換する。図 4 に変換方法を示す。太線内部が検索すべき範囲を示している。時空間 DB の空間検索範囲は X 軸と Y 軸に平行な長方形領域となるので、本変換方法では投影面の前に長方形を配置することで近似的に検索の解を得ることとする。具体的に、この長方形は、投影面の長さ L を一辺とする正方形（または、一辺を $\sqrt{2}L$ とする正方形）とし、それをユーザの閲覧方向に K 段配置する。検索の際、最初は $k = K_1$ 段目の正方形領域を選択する。検索結果表示後、ユーザがより近くのカメラスを表示するように要求した場合は $k=k-1$ 段目とし、ユーザがより遠くのカメラスを表示するように要求した場合は、 $k=k+1$ 段目として、時空間 DB の空間検索範囲とする。

4.3 画面例

図 5-(a) に Map viewer の画面例を、図 5-(b) に検索結果の表示画面例を示す。Map viewer では、マウスでまず仮想壁を描き、次にその仮想壁を見る方向を描く。次いで、検索時間範囲を入力し、[Search] ボタンを押す。サーバ側で検索された結果は、Web ブラウ

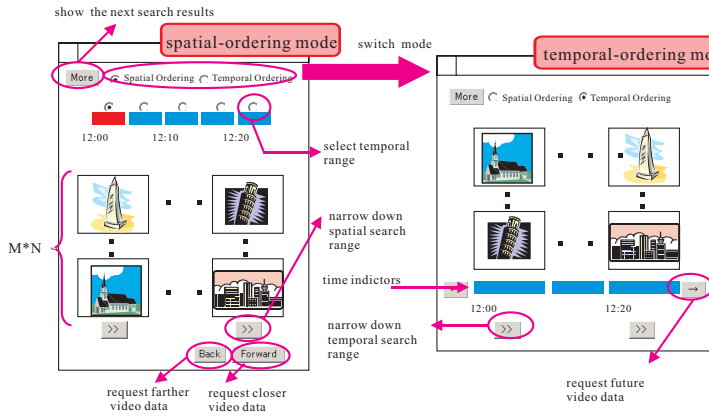


図3 検索結果表示インタフェース
Fig.3 Interface of the search result page

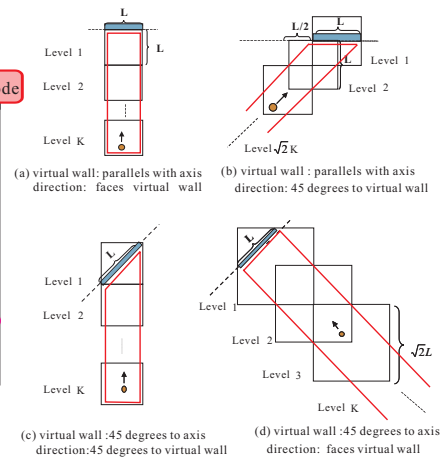


図4 空間検索キーの変換
Fig.4 Transformation of spatial key

ザに表示される。この画面例では、空間整列モードとなっており、画面中で左から順に並んでいる映像は、実際の空間でも仮想壁に向かって左から順に対応するカメラが並んでいる。画面中の各ボタンとインディケータは、前章で説明したとおりである。

5. 評価

本システムの特徴は、映像データを分散管理し、その索引情報から時空間 DB で検索することと、仮想壁による検索インタフェースにある。評価は、データ更新の時間や検索時間について行うものと、インタフェースの使いやすさについて行うものの、2つの観点から行う。

5.1 データ更新・検索に関する評価

本評価では、一つの町に大量のカメラがあることを想定し、それらの映像のシミュレーションデータを作成し、その上で評価を行うこととした。

5.1.1 シミュレーションデータの作成と実験方法

シミュレーションデータは、以下のような条件を想定して作成した。

- 対象とする地域の面積は $1,000,000m^2$ (一つの町を想定)
- カメラの設置間隔は $100m^2$ に1台でランダムに配置
- 全時間範囲は8時間
- 映像データの分割時間単位は5分
- カメラ1台あたりの撮影時間は2時間

これで、総計 240000 個の映像ファイルを得る。

このシミュレーションデータをカメラ側のローカル PC から一つずつ順次にサーバに投入していきデータ

構造を構築した。その次にローカル PC から映像データを1つ投入してデータ更新にかかる時間を計測した。また、仮想壁として建物の壁程度の長さのもの(幅 100m)を想定し、時間検索範囲を30分として、Web ブラウザからアクセスした際の検索時間を計測した。ここで検索時間は、時空間 DB を使う場合と使わなかった場合(つまり全検索した場合)の両方で計測した。また、検索キーには、それぞれ図4に対応の4種類を用意した。それぞれ、検索キー a~検索キー d と呼ぶ。シミュレーションには、Pentium4(1.5GHz)のマシンを使用した。

5.1.2 実験結果

データ構造更新時間に関しては、最初に 240000 個分のデータを索引付けするのに要した時間は、通信時間も含めて 68.7 秒であった。ついで、その状態で1つの映像データを1つ投入した際に要したデータ構造更新時間と通信時間は、それぞれ $0.000154[s]$ と $0.00016[s]$ である。この結果から、最大で毎秒約 3000 個 ($1/(0.000160 + 0.000154)$) のデータが挿入されても、データ構造を更新していけることが分かる。ファイル分割の単位を5分おきとすると、最大で90万個のカメラからの映像を管理できることが分かる。このシミュレーションでは、カメラの個数は10000個としたので、およそ一つの町の監視カメラ程度であれば、リアルタイムに映像を管理できることが分かる。

次に検索時間に関しては、サーバ側で、時空間 DB への検索に要した時間(データ構造検索時間)、検索結果として得られた映像データの URL をソートして検索結果のページに挿入するのに要した時間(ページ生成時間)、作成したページをクライアントに送信するの

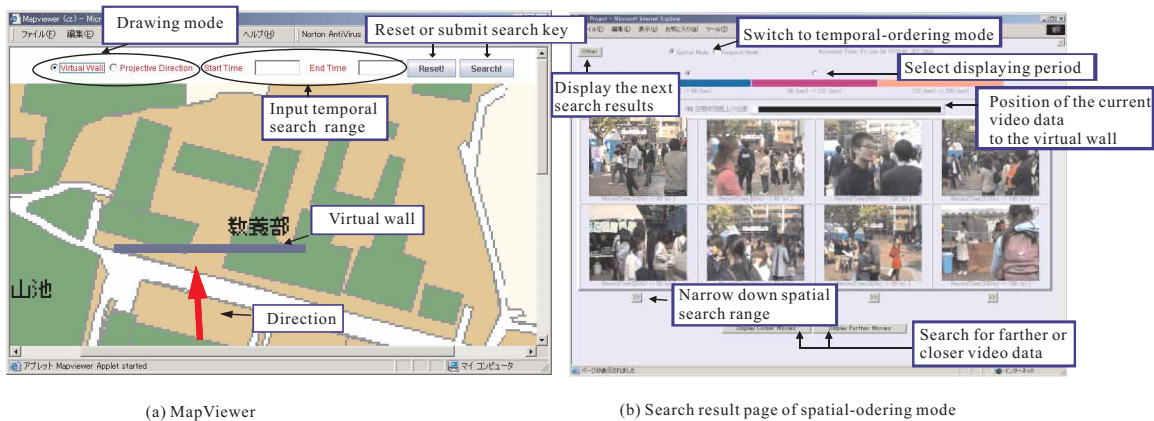


図 5 システム画面例
Fig.5 Screenshot of the system

に要した時間(データ通信時間)に分けて計測した。時空間 DB を用いた場合の結果を図 6 に示す。検索キー a~d のこれらの合計時間の平均は、0.094[s]であった。アクセスして結果が返ってくるまでの、ユーザが待つことができる許容時間を 5 秒以内として、何人アクセス可能かを計算すると、53 人 ($1/0.094 * 5 = 53$) となる。このことから、1 つの組織で使う分には十分な速度であると言える。

また、時空間 DB を用いなかった場合の結果について、データ構造検索時間について全ての検索キーの平均に注目すると、時空間 DB を使った場合に 0.0011[s]であったのが、全検索を行った場合には 0.086[s]になっている。このことから、時空間 DB によりデータ構造検索時間は、約 78 倍早くなったことが分かる。また、すべての処理時間の合計に注目すると、時空間 DB を使った場合に 0.094[s]であったのが、全検索を行った場合には 0.178[s]になっている。その他の処理に(特にページ生成)時間がかかるため、検索にかかる処理の全体としてはそこまで早くはなっていないが、それでも 2 倍近く早くなっていることが分かる。このことから、時空間 DB ありの場合は同時アクセス可能人数が 53 人であったが、これが 27 人に減少することがわかる。このことから、時空間 DB の効果が検証された。ただし、特にページ生成に時間がかかっており、ページ生成に必要なテキスト処理を簡素化するなどの工夫を行う必要があることも分かった。

5.2 検索インターフェースに関する評価

本評価では、現実起こった出来事を記録した映像データを収集するために、筆者らの大学の学園祭の日に、ステージや露店を中心にビデオカメラで撮影をして廻ることとした。ここでは、映像データの撮影方法

| Search Key type | Search time [s] | | Page creation time [s] | Data transmission time [s] |
|-----------------|-------------------------|----------------------------|------------------------|----------------------------|
| | with spatio-temporal DB | without spatio-temporal DB | | |
| Type (a) | 0.00091 | 0.084 | 0.054 | 0.0036 |
| Type (b) | 0.0012 | 0.085 | 0.092 | 0.0044 |
| Type (c) | 0.0010 | 0.085 | 0.081 | 0.0042 |
| Type (d) | 0.0012 | 0.088 | 0.127 | 0.0042 |
| average | 0.0011 | 0.086 | 0.089 | 0.0041 |

図 6 データ検索評価結果
Fig.6 Evaluation of data search

と、ユーザ実験による評価方法、及びその評価結果について述べる。

5.2.1 映像データの撮影方法

大学構内に多数のカメラを配置するのが理想ではあるが、配置するカメラの台数には限界があるため、一つのカメラを持って、大学構内の主要な箇所を廻って撮影することにした。具体的な映像データの作成方法は次のとおりである。

- 撮影時間帯は大学祭の開催される二日間の 10:30 ~ 16:30 とした。
- 撮影地点を 25 個とした。
- 一つの撮影地点では、複数の撮影方向を設けた。
- 1 回の映像撮影時間は 30 秒とし、これを 1 つのファイルとした。30 秒おきに撮影方向を変えるか、または新たな撮影地点で撮影することとした。
- 一つの映像データを撮影した後、新たな映像データを撮影するまで、1 分間の移動・準備時間を設けた。

この方法により、513 個の映像データを得た。各映像のカメラの方向データと位置データは、後から手動でつけた。映像データの時間情報を付加する際は、以下のルールに従うこととした。

- 実時間の同じ 15 分間で撮影したすべての映像データは、時間情報を付加する際に同じ時刻のものとする（例えば、撮影時刻 10:00:00 から 10:15:00 まで撮影した映像は、時間情報を付加すると 10:00:00 から 10:00:30 まで撮影したものとみなす）
- 次の 15 分間に撮影した映像は、前の 15 分間に撮影した映像から、10 秒遅らせた時刻の映像とする。（例えば、撮影開始時刻が 10:00:00 の場合、10:15:00 から 10:30:00 まで撮影した映像は、10:00:10 から 10:00:20 まで撮影したものとみなす）

これにより、実際の時刻とはずれはあるが、同じ時刻に異なる場所を撮影した映像が複数あるようにした。場所を変えるのに 1 分ほど要するため、平均して同じ時間帯にカメラが 30 台程度あるようにした。

5.2.2 実験方法

検索インタフェースの評価は、検索インタフェースがある場合とない場合とを比較することとする。検索インタフェースのない場合は、時空間 DB に対する検索キーを直接入れてもらうことにした。このうち、空間検索範囲とカメラ方向は、地図上でマウスで入力する。時間検索範囲は、数値をキーボードから入れる。また検索結果の表示は、映像を $M \times N$ の行列にランダムで映像を埋めていくことにした。

被験者には 20 代の大学生・大学院生の 16 人に参加してもらった。彼らは、映像を撮影した大学構内の地理については詳しいが、学園祭のどこに露店が出るかや、どこでイベントが行われるかと言う情報は全く知らない。最初に被験者には練習用のタスクを行ってもらった。練習タスク用の映像として、筆者らの大学の建物周辺に、前節と同様の方法で 30 箇所の映像を用意した。練習タスクでは、検索インタフェースの「概要から提示し後から詳細が見れる機能」「空間絞り込み機能」「時間絞り込み機能」「ある角度から眺める機能」の 4 つを経験できるように設定してある。被験者は、この練習タスクを検索インタフェースなしの場合とありの場合とで 2 回繰り返した。

次に被験者は、以下の評価用のタスクを行った。

- (1) 学園祭中、雨の降っている時間帯を調べる。
- (2) 学園祭のステージのある場所を検索し、以下のことを調べる。
 - (a) ステージで行われているイベントの内容を記述する。
 - (b) それらイベントのうち、一番客の多かったものを挙げる。
 - (c) 上記イベントで、ステージの上の人々の

特徴を 5 つ記述する。

これらのタスクでも、上述の 4 つの機能を用いるように設定してある。被験者は、片方のタスクを一方の検索インタフェースで行った後、もう一方のタスクを別の検索インタフェースで行う。ここで、カウンターバランス（被験者実験において水準の順序効果の相殺を行うこと）をとるため、被験者 16 人を 4 人ずつの 4 つのグループに分けた。それぞれ、上述の 2 つのタスクを、順番とインタフェースあり・なしの組（IF あり (with IF)・IF なし (without IF)）で、振り分けている。評価にはタスク実行時間を計測した。

5.2.3 実験結果

全ユーザのタスク遂行時間の平均と標準偏差を図 7 に示す。この図からインタフェースがある場合の方が、タスク遂行時間が低いことが分かる。それぞれのタスクで、 t 検定（片側）を行ったところ、5%の危険率で有意差があることが分かった。このことから、検索インタフェースがある方が、タスクの実行時間が低くなることが検証された。

さらに、上記のような結果に至った要因として、ユーザは提案する検索インタフェースのどこを評価しているのかをアンケート（5 段階評価とその評価結果に至った理由を自由記述）で調査した。

アンケートの内容及び結果を図 8 に示す。質問 (1) では、提案する検索インタフェースを支持する人が 75%に達しており、概ね多くのユーザに受け入れられていることが分かる。次に、質問 (2) と質問 (3) を比較することで、入力部と出力部のどちらをより支持しているかを確かめる。その結果、入力部を支持する人は 56%で、出力部を支持する人は 100%となっている。どちらも、従来のインタフェースよりも高い評価を得ているが、より出力部を支持する結果となっている。自由記述の回答を見てみると、入力部を支持した人は「検索する際にカメラのことを意識しないですむ」、「方向と合わせて検索できるので感覚的に分かりやすい」と言ったコメントを残しているが、支持しなかった人の中には「場合によっては検索範囲をはっきりさせたい」という意見もあった。このことから入力インタフェースには、依然として従来型の範囲検索のインタフェースを好むユーザがいることが分かる。したがって、空間範囲の指定には、仮想壁を使ったインタフェースと従来型のインタフェースの両方を提供するのが良いと考えられる。

最後に、質問 (4) に関しては、81%の人が最初に概要を見てそこから絞り込んでいくアプローチを支持しているが、19%の人に関しては、あまり支持していな

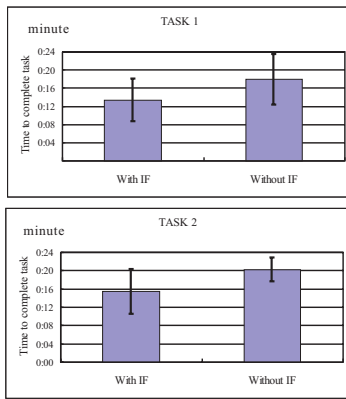


図7 タスク遂行時間

Fig. 7 Time required to complete the task

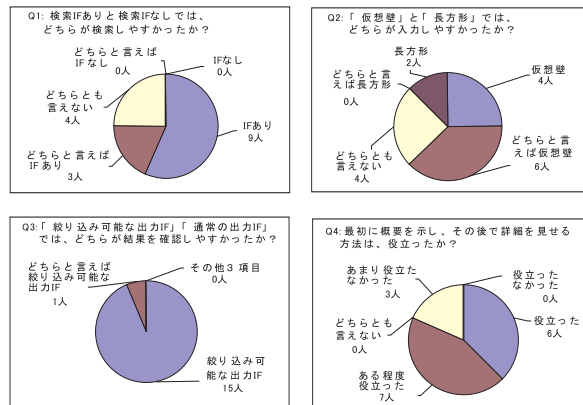


図8 アンケート結果

Fig. 8 Results of questionnaires

い。質問 (3) ほど高い評価が得られなかった理由としては、「最初から詳細が見たい場合に、遠くから撮影された映像が出力された」という意見があった。つまり、仮想壁を閲覧する最初の距離がユーザの要求に合わない場合があったと言える。これに関しては、検索時に「初めから詳細を見る」のようなチェックボックスを設けることで対応できると考える。

6. 関連研究

分散環境にある映像を効率よく管理することを目指すシステムとしては、Feng らのシステム³⁾ や秦らのシステム¹⁾ がある。Feng らは、ネットワークが遮断しても映像を撮影しつづけて、それらの映像をローカルのメモリに一旦蓄え、ネットワーク復帰後にサーバに格納した映像を送ることができる映像ストリーミングのシステムを開発している。しかし、彼らはサーバ側で映像データ本体も集中管理することを目的としており、サーバ側の通信負荷は考慮していない。秦らは映像データ本体をローカルのカメラに置き、撮影情報のみをサーバ側で管理する方式を提案している。撮影情報の通信を TCP レベルで行い、実際の検索はサーバ上に実装されたオリジナルのアプリケーションから行っている。しかし、これでは現在のインターネット環境を考えると、ファイアウォールをデータが通ることができず、誰もが任意の場所から映像を検索することはできない。本研究は、サーバ側の通信負荷を考え、秦らと同様に撮影情報のみをサーバ側で管理するが、そのやりとりには HTTP を用い、Web ブラウザからインタラクティブに検索ができるようにしている。また、検索時間を短くするため、撮影情報を時空間 DB で管理している。

7. まとめ

本稿では、GUI による検索を可能とした分散型映像データ管理システムを提案した。カメラの位置と撮影対象との位置がずれる従来型の範囲検索ではなく、仮想壁を使ったユーザの見え方に主眼を置いた検索キーの入力インタフェースを提案した。ユーザ 16 人による評価実験では、タスクの実行時間が削減され、また検索インタフェース自体も概ね支持された。また、インデックスの作成時間とユーザの検索時間により評価したところ、一つの地方行政機関が持つカメラの数程度であれば実用可能で、組織内で複数のユーザが検索する分には十分な応答速度を持つことが分かった。今後の課題としては、検索範囲中のカメラの台数なども考慮し、従来型の範囲検索手法とうまく組み合わせた検索インタフェースを構築することが挙げられる。

謝辞 本研究の一部は、文部科学省の学術創成研究費 (課題番号 13GS0018) と科学研究費補助金基盤研究 (B)(2) (課題番号 15300034) の助成を頂きました。

参考文献

- 1) 秦 淑彦, 塚田晶宇, 尾崎 稔, 坊 覚: 分散型履歴映像データの効果的検索・再生方式と実装, 電子情報通信学会論文誌 D-I, Vol. J82-D-I, No. 1, pp. 234-246, 1999.
- 2) X. Xu, J. Han and W. Lu, RT-tree: An Improved R-tree Index Structure for Spatiotemporal Databases, *Proc. of the 4th Intl. Symposium on Spatial Data Handling*, pp. 1040-1049, 1990.
- 3) W.C. Feng, et al., Panoptes: Scalable Low-power Video Sensor Networking Technologies, *Proc. of ACM Multimedia*, pp. 562-571, 2003.