# 位置や時刻の近さに基づいた文書検索

寺西 由利香　黒木 進　北上 始

広島市立大学情報科学部

〒731-3194 広島県広島市安佐南区大塚東 3-4-1

E-mail: {yurika,kuroki,kitakami}@db.its.hiroshima-cu.ac.jp

**あらまし** ユーザが指定した位置や時刻に空間的にあるいは時間的に近い位置や時刻を表すキーワードを含む文書を検索する方法について論じる．このような時空間検索を可能にするためには、文書が持つ空間的なあるいは時間的な語句を抽出し、それをメタデータとして文書に添付する．添付されたメタデータについて時空間近傍検索が行なわれ、検索キーワードに近いキーワードを持つ文書が検索される．そこでわれわれは、位置的あるいは時間的な観点からの文書の分類と近傍検索、および検索結果の表示について述べる．地図やカレンダーを利用することにより検索された文書の空間的あるいは時間的な関係をわかりやすくできる．

**キーワード** 時空間データベース，近傍検索，時空間検索

# Document Retrieval Based on Proximity of Location and Time

Yurika TERANISHI　Susumu KUROKI　and　Hajime KITAKAMI

Faculty of Information Science, Hiroshima City University

3-4-1 Ozuka-higashi, Asaminami-ku, Hiroshima-city, Hiroshima, 731-3194 Japan

E-mail: {yurika,kuroki,kitakami}@db.its.hiroshima-cu.ac.jp

**Abstract** We discuss a search method of documents whose keywords are spatially and/or temporally near to location and time which a user specified. In order to make such spatio-temporal search feasible, spatial and temporal keywords are extracted from documents and are attached to them as meta-data. Spatial and/or temporal proximity query are carried out on the meta-data to retrieve documents whose keywords are closer to the query keywords. Therefore, we address classification of documents from the point of spatial and/or temporal point of view, proximity search of meta-data and presentation of query results. Using maps and calendars as presentation method makes spatial and/or temporal relationship among retrieved documents easy to understand.

**Keyword** Spatio-Temporal Databases，Proximity Search，Spatio-Temporal Retrieval

## 1. はじめに

今日，われわれは Google[1]などのサーチエンジンを用いて「広島市安佐南区大塚東」といった地名を含む文書や「11 月 7 日」などの日時を含んだ文書を検索できる．しかし，既存のサーチエンジンでは，時間や位置の近さに関する知識を組み込んでいないので，例えば以下のようなことができない．

- キーワードとして与えられた地名や日時を文字列として含んだ文書を返すが，与えられた地名の近隣や日時の前後に該当する語句を含んだ文書を返すことができない．
- 同じ位置を表す言葉には住所・施設名などいろいろあるが，サーチエンジンはそれらを同じものとして扱うことができない．

そこで，この研究では，インターネット上にある文書について，ユーザが指定した位置や時刻に近い位置や時刻を表すキーワードを含むものを検索できるようにすることを試みる．例えば，引越し先として考えている場所の近辺の情報を知りたい，あるいはＧＰＳや携帯端末を使って今いる場所の近くの情報を知りたい，その場所での出来事について現在から過去へと順にた

どって調べたいなどの要求にこたえる．

類似研究として横路らの研究[2]がある．彼らは，位置に関する近傍検索が行えるようにしているが，この研究では時間に関する近傍検索も行えるようにしようと試みている．また，相良らは地名を軽度・緯度に変換するシステムを実装したり[3]，空間サーチエンジンを構築している[4]が，時刻情報に関する検討を行っていない．

## 2. 位置と時刻に関する検索

### 2.1. 位置と時刻に関する問合せ

この節では，位置と時刻に関する問合せの種類について述べる．

#### 2.1.1. 位置に関する問合せ

位置に関する問合せには，ある位置の近隣（例えば，周辺 2km）についての記述がある文書について問い合わせる近傍検索がある．

#### 2.1.2. 時刻に関する問合せ

時刻に関する問合せには，以下に示す 3 種類が考えられる．

- ある特定の時刻に関する問合せ
  - 2003 年 1 月 10 日 13：00 などの，ある特定の時点のデータを条件に問い合わせる．近傍検索が可能である．
- 周期的な時刻に関する問合せ
  - 毎週月曜日，毎月 1 日など，周期的な時刻を表すデータを条件に問い合わせる．
- ある区間をもつ時刻に関する問合せ
  - 2003 年 1 月 1 日から 2003 年 1 月 10 日や毎週月曜の 13：00～15：00 などの，時刻に幅のあるデータを条件に問い合わせる．

本研究では，ここで述べた位置に関する問合せ，ある特定の時刻に関する問合せ，周期的な時刻に関する問合せとある区間をもつ時刻の問合せの一部を可能とした．

### 2.2. データベース

この節では，本研究で使用するデータベーステーブルとその使用方法について述べる．

#### 2.2.1. 地名辞書

本研究では，検索条件として与えられた位置や時刻を表す文字列を含む文書だけでなく、その位置の近隣あるいはその時刻の前後についての記述がある文書も検索できるようにする．

この機能を実現するために，以下の 3 種類のデータベーステーブルを作成する．

表 1 のテーブルは，キーワードとして与えられた地名を経度・緯度に変換するためのものである．

表 1　地名辞書

| 地名 | 経度 | 緯度 |
|---|---|---|
| 広島市安佐南区大塚2 | 1322201 | 342836 |
| ⋮ | ⋮ | ⋮ |

#### 2.2.2. 文書データベース

表 2 のテーブルは，位置や時刻に関する記述を含む文書を収集し，作成したものである．例えば，図 1 のニュースの文書を文書データベースに挿入するとする．文書中にある住所などの位置情報(「広島市中区白島北町」，「原爆ドーム」，「元安橋」)を抽出し経度・緯度に変換したもの，時刻情報(2002.11.4)，ＵＲＬの情報を抽出し，文書番号をつけて文書データベースに挿入する．今回はこれを手動で行った．

表 2　文書データベース

| 文書番号 | 経度 | 緯度 | 時刻 | URL |
|---|---|---|---|---|
| 1 | 1323124 | 342111 | 2002.10.15 | http://www... |
| 2 | 1322701 | 342345 | 2002.11.4 | http://www... |
| 3 | 1321955 | 342205 | 2002.11.4 | http://www... |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

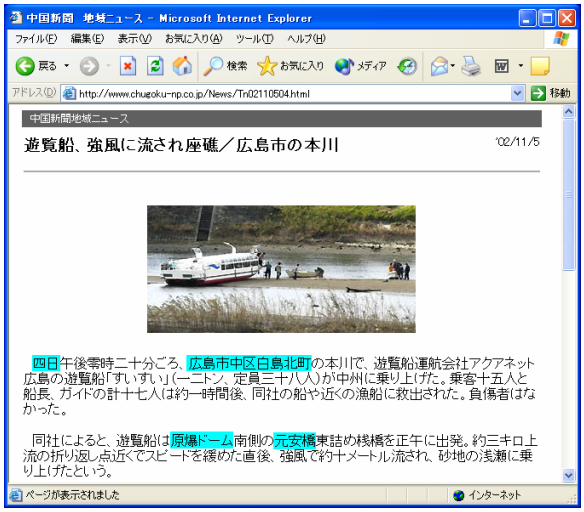


図 1　文書の例

### 2.2.3. ハッシュテーブル

前節の表２のテーブルから以下のようなテーブルを作成する．

表３は，近傍検索をするために文書を整理したインデックス(オープンハッシュ)である．各ハッシュ値は以下のように決定する．

$h_1$…経度を１００で割ったときの商とする．

$h_2$…緯度を１００で割ったときの商とする．

$h_3$…ある基準となる時刻を０とし，順次数値を割り振る．

ただし，今回の実験では広島県内に限っているため $h_1$ と $h_2$ の上位２桁を省略した．

表３　ハッシュテーブル

| $h_1$ | $h_2$ | $h_3$ | 文書番号 |
|---|---|---|---|
| 219 | 22 | 35 | 3 |
| 219 | 22 | 36 | NULL |
| 219 | 22 | 37 | NULL |
| 231 | 21 | 15 | 1 |
| 227 | 23 | 35 | 2 |
| 227 | 23 | 36 | NULL |
| ⋮ | ⋮ | ⋮ | ⋮ |

また，2.1.2 節の時刻に関する問合せで述べた，周期的な時刻に関する問合せについて考える．周期的な時刻を表す「曜日」を入力としての検索を可能とするには，次のようなテーブルが必要となる．そのテーブルを表４に示す．

表４　「曜日」検索用ハッシュテーブル

| $h_1$ | $h_2$ | $h_4$ | 文書番号 |
|---|---|---|---|
| 219 | 22 | 0 | 3 |
| 219 | 22 | 1 | NULL |
| 219 | 22 | 2 | NULL |
| 231 | 21 | 6 | 1 |
| 227 | 23 | 4 | 2 |
| 227 | 23 | 5 | NULL |
| ⋮ | ⋮ | ⋮ | ⋮ |

表４は表３のハッシュテーブルと類似しているが，違いは $h_3$ ではなく $h_4$ となっている部分である．ハッシュ値 $h_4$ は，0 から 6 までの値をとり，0 が日曜日，1 が月曜日，という具合に曜日を割り当てる．この「曜日」検索用ハッシュテーブルを用いることにより，「曜日」を指定した周期的な時刻に関する問合せが可能となる．

以上の３種類のテーブルを利用し，次のように近傍検索を行う．検索条件として与えられた地名を表１のテーブルで経度・緯度に変換し，そのハッシュ値を計算する．例えば，そのハッシュ値が $h_1$＝256，$h_2$＝23 とすると，図２の $h_1$＝256，$h_2$＝23 の位置にある文書(点)がハッシュテーブルより求まる．図２の例では２つの文書が求まったことになる．また，$h_1$，$h_2$ の数値を変化させて，図２の色のついた位置の，近隣にある文書もハッシュテーブルより求めることができる．このようにハッシュ値を用いて検索するとことにより，調べる必要のない位置についての文書を調べなくてすむので，検索速度を速めることができる．

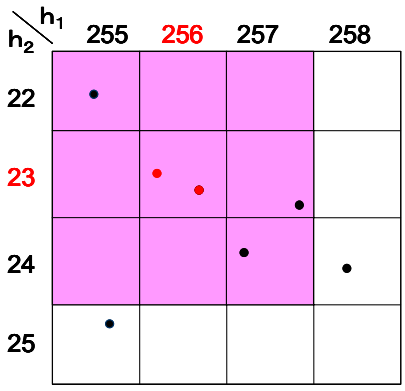


図２　地図

時刻に関する検索も，$h_3$ または $h_4$ を変化させ，同様に行うことができる．

### 2.3. 検索アルゴリズム

図３に，サーチエンジンに検索条件を与えた後，データベースにアクセスし検索する簡単なプログラムの流れを示す．図中の★は，データベースへの問合せを表し，その説明での矢印の左側は問合せをするときの条件，右側は問合せの結果得られる情報を表している．

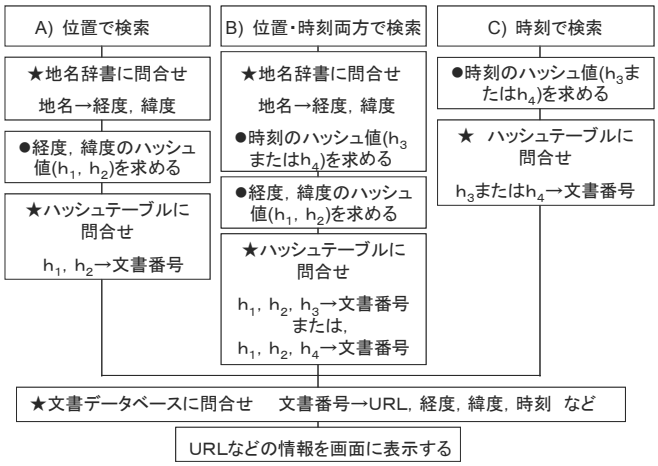


図３　検索アルゴリズム

## 3. 試作システム

### 3.1. システム構成

試作したシステムの構成を図４に示す.

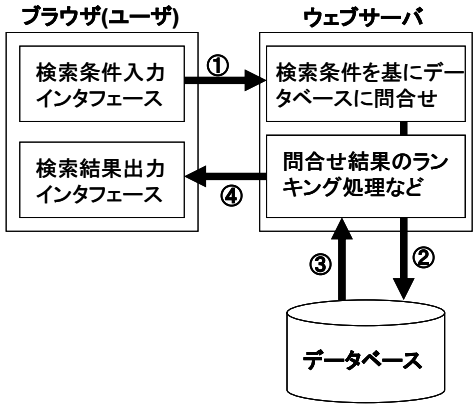


図４　システム構成

本システムにおける処理の流れは以下のようになる.

① ユーザがブラウザから検索条件を入力する.
② ウェブサーバで，検索条件を基に，前節の図３で示した手順でデータベースに問い合わせる.
③ 問合せの結果返された情報をユーザにとって利用しやすいかたちにするための処理をする.
④ 検索結果をユーザ側に表示する.

### 3.2. 検索プログラム

試作したシステムの検索プログラムでは，次のようなことができるようにした.

住所，施設名，駅名などで同じ位置を表す言葉を同じものとして扱い，検索できる.

住所，施設名，駅名のそれぞれから経度・緯度を求めるためのデータベーステーブルを作成することにより，入力として位置には，住所，施設名，駅名のいずれかを用いて検索できるようにした.

時刻入力においては，ある特定の日時を入力する方法のほかに，祝日や月，季節，曜日も入力して検索できる．月や季節を入力としての検索は，ある区間をもつ時刻に関する問合せとなる．また，曜日を入力としての検索は，周期的な時刻に関する問合せとなる.

祝日，月，季節が入力された場合，時刻のハッシュ値($h_3$)を求める際に必要となる月日の指定方法を表５に示す．ただし，年は，検索が行われた年とする.

表５　月日の指定方法

| 入力 | 月日の指定方法 |
|---|---|
| 祝日 | その祝日と対応する月日. |
| 月 | 指定した月の一日からその月の最後の日. |
| 季節 | 春:3月1日から5月31日.<br>夏:6月1日から8月31日.<br>秋:9月1日から11月30日.<br>冬:12月1日から2月の最後の日. |

このようにそれぞれの入力方法に応じて時刻のハッシュ値($h_3$)を求め，そのハッシュ値に該当する文書をハッシュテーブルより探し出す．このようにして検索を行っていく．また，祝日，月，季節が入力された場合に指定される年は検索が行われた年としているが，同時に,過去５年の年も指定して検索を行うようにした.

### 3.3. インタフェース

検索結果として返された文書に含まれる地名などの位置関係や時刻の前後関係が一目でわかるようなインタフェースがある．このインタフェースを用いて検索を行うときにユーザがとるアクションを以下で説明する.

#### 3.3.1. 入力画面

図５は，検索条件入力画面である．位置のみを入力して検索を行うと，検索条件として与えた位置とその近隣についての記述がある文書を探すことができる．時刻のみを入力して検索を行うと，検索条件として与えた時刻とその前後の時刻についての記述がある文書を探す．また，位置と時刻両方を入力して検索を行うと，その両方についての検索を行う.

位置入力は「住所」,「公共施設名」,「駅名」のいず

れかを入力できる．

時刻入力は，「日にち」，「祝日」，「月」，「季節」，「曜日」のいずれかを入力できる．「日にち」を入力する場合は，オプションで「頃」，「以前」，「以降」を選択できる．ここで，「頃」を指定して検索すると，その日の前後あわせて３日間についての記述がある文書を探す．「以前」や「以降」を指定すると，その日以前，またはその日以降についての記述がある文書すべてを探す．

また，位置と時刻両方を入力して検索する場合は，どちらを優先して検索するか，その割合を指定しなければならない．これは，検索結果として返された文書情報を出力する際に，どの文書を優先的に表示するかを決めるためのものである．

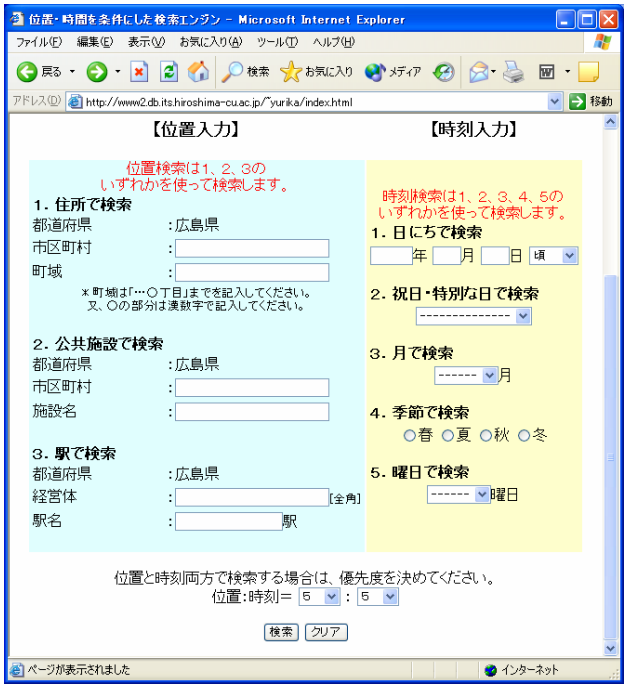


図５　入力画面

### 3.3.2. 地図表示画面

図６は，図５で位置を入力して検索ボタンを押した後の出力画面である．検索結果として返された文書に含まれる位置や時刻の情報を基に，位置関係や時刻の前後関係を表示する Java アプレットがある．位置関係は，文書中の位置情報を基に，地図上に点をプロットして表示する．時刻の前後関係は，文書中の時刻情報を基に，プロットした点に色をつけることでおおまかな前後関係を表示する．具体的には，検索結果として返された文書の時刻情報をハッシュ値( $h_3$ )に変換し，その中で大小比較し，６段階にわけて色で表示する．その点をクリックすると図７のリスト表示画面に変わる．

地図には，総務省が定めた標準地域メッシュの１つである第３次地域区画を使用する．これは，経度差 45 秒，緯度差 30 秒，面積約１ｋｍ$^2$の地図表示方法である．また，この地図画像は地図左下の位置の経度・緯度を組み合わせたファイル名で保存しておき，検索時にはプログラム中で経度・緯度を指定することで地図を表示する．

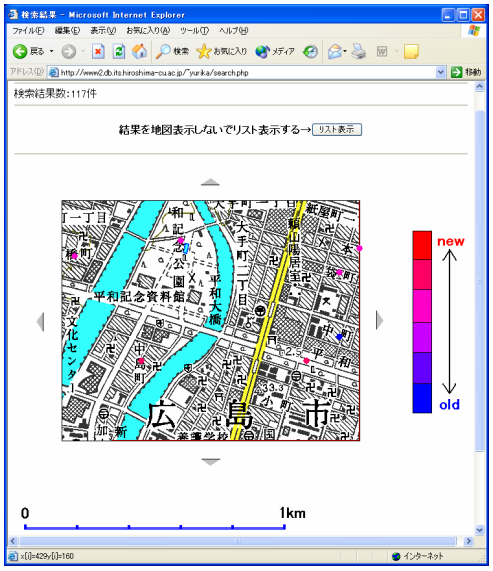


図６　地図表示画面

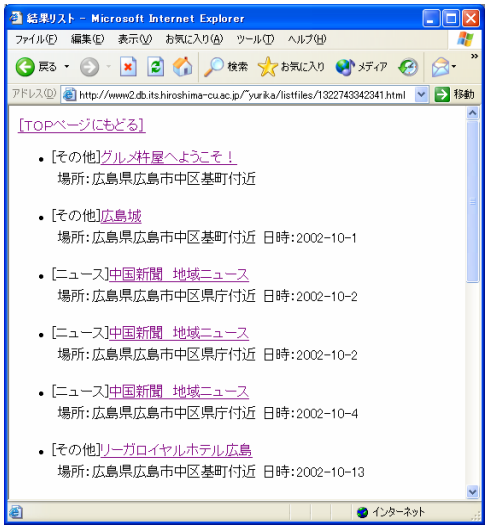


図７　リスト表示画面

図７は，検索結果として返された文書で図６の地図のクリックされた位置に関する文書の情報をリスト表示し，目的ページへのリンクが張られてある画面である．このページのＨＴＭＬ文書名は，経度・緯度を組み合わせた名前で，検索時に動的に作成される．

### 3.3.3. カレンダー表示画面

図８は，図５の入力画面で，時刻のみを入力して検索した場合の検索結果出力画面である．検索結果として返された文書に含まれる時刻の前後関係をわかりやすく表示するため，文書情報がカレンダー形式で表示される．

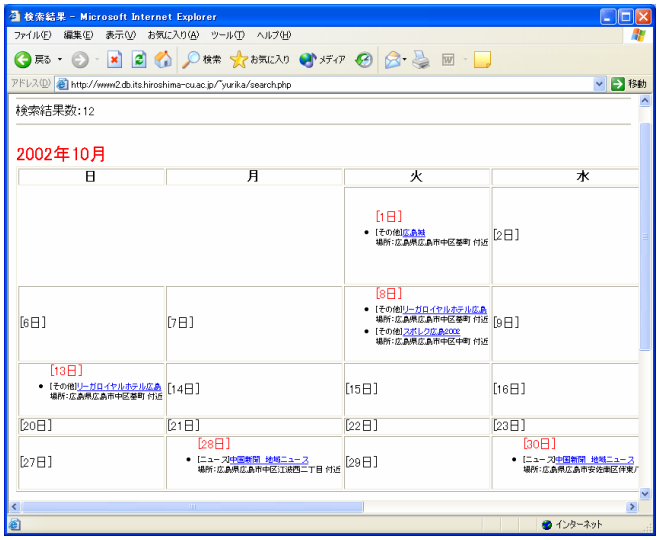


図８　カレンダー表示画面

### 3.3.4. ランキング表示画面

3.3.2 節の図６や前節の図８のようなインタフェースは，検索結果として返された文書に含まれる地名などの位置関係や時刻の前後関係を知るのに役立つ．しかし，どの文書がユーザの知りたい情報のある文書なのかが，一目でわかりにくい．そこで，検索結果として返された文書それぞれに，どれだけユーザの知りたい情報のある文書かを示す適合度を計算し，適合度順にランキングした画面も作成した．そのランキング表示画面との切り換えが，図６や図８の画面から行えるようにした.ランキング方法についての詳しい説明は，次節で行う．

## 3.4. ランキング

検索結果として返された文書に含まれる位置情報と検索条件として入力した位置との距離の近さや，文書に含まれる時刻情報と検索条件として入力した時刻との差から，文書の適合度を決め，検索結果をランキングして表示する．ただし,「曜日」を入力としての検索の場合は，周期的な時刻の問合せとなるため基準となる時刻がないので，ランキングを行わない．

検索条件として位置のみを入力する場合，検索条件として与えられた位置の経度・緯度を $\mathbf{x}_q = (x_q, y_q)$ とし，検索結果として返された文書に含まれる位置の経度・緯度を $\mathbf{x} = (x, y)$ とすると，適合度 d は以下のように求める．

$$d(\mathbf{x}, \mathbf{x}_q) = \sqrt{(x - x_q)^2 + (y - y_q)^2}$$

また，検索条件として時刻のみを入力する場合，検索条件として与えられた時刻のハッシュ値($h_3$)を $d_q$，検索結果として返された文書に含まれる時刻のハッシュ値($h_3$)を $d$ とすると，適合度は以下のように求める．

$$|d - d_q|$$

検索条件として位置と時刻両方を入力した場合の適合度 D は以下のように求める．

$$D = \sqrt{\alpha d(\mathbf{x}, \mathbf{x}_q)^2 + \beta |d - d_q|^2} \qquad \alpha + \beta = 1, \alpha, \beta \geq 0$$

ここで，α はランキングを決める際の位置に関する優先度を表し，β は時刻に関する優先度を表している．この優先度は，ユーザが検索条件入力画面で指定するものとする．

検索条件として位置と時刻両方を入力した場合の，α と β の選び方によるランキング表示画面の違いを以下に説明する．

図９は，位置入力を「広島市中区紙屋町」，時刻入力を「2002 年 11 月 20 日以降」とした場合の上位７つのランキング表示画面である．(a)の画面が α ＝0.9，β ＝0.1 で，位置を優先的に検索した結果の出力で，(b)の画面が α ＝0.1，β ＝0.9 で，時刻を優先的に検索した結果の出力である．

(a)の画面を見ると，検索条件として与えた「広島市中区」の情報のある文書が上位７位を占めている．(b)の画面を見ると，上から 2002 年 11 月 20 日から日付順に比較的間をあけずランキングされている．具体的な例を挙げると，(a)の画面で３つ目にランキングされている，場所が「広島市中区宝町」で日時が「2002 年 11 月 25 日」の文書に注目すると，位置を優先した(a)では，日時が，検索条件である「2002 年 11 月 20 日」と５日離れているにもかかわらず，「広島市中区紙屋町」に近いため，３位にランキングされている．しかし，時刻を優先した(b)では，７位にランキングされている．

このように，ユーザの知りたい情報のある文書をランキング表示することができる．


(a) $\alpha = 0.9$, $\beta = 0.1$


(b) $\alpha = 0.1$, $\beta = 0.9$

図 9　ランキングの比較

## 4.　今後の課題

今後の課題として次のことが挙げられる．

3.3.2 節で述べた，位置と時刻両方を入力した場合の出力画面である図 6 のような地図表示画面について，現在は，結果として返された文書に含まれる時刻の前後関係を色でわけて表現しているが，これでは，具体的な時刻の差が一目ではわからない．そこで，検索結果として返された文書に含まれる地名などの位置関係や時刻の前後関係の両方が同時に詳しくわかるようなインタフェースに工夫する必要がある．

また，現在，この検索を行うのに必要なデータベーステーブルに格納する文書情報の収集には，1 件約 20 秒程度かかる．これは，文書から位置情報，時刻情報，ＵＲＬ，タイトルを抽出し，データベーステーブル挿入用のページにこれらの情報を張りつけて送信するのにかかるおおまかな時間である．このデータベーステーブルの更新にかかる時間の短縮は，今後の課題の 1 つであり，文書から自動的に位置情報や時刻情報などを抽出しデータベースに挿入する方法を検討する必要がある．

## 5.　おわりに

現在のサーチエンジンでは，検索条件として与えられた文字列を含んだ文書だけを返す．検索条件として位置や時刻を与えた場合，キーワードを含む文書は返すが，与えられた位置の近隣や与えられた時刻の前後について記述されている文書は返すことができない．そこで，上記で説明した時空間データベースの技術をサーチエンジンに導入して，文書に含まれる位置や時刻に関する空間的・時間的な条件に基づいた検索を可能とする．

文書検索において，文書中の文字列と検索条件のキーワードの一致ではなく，位置や時刻の意味の一致や近さに基づく検索を可能にすることによって，サーチエンジンの検索能力を高めることができる．

## 謝辞

本研究は，日本学術振興会若手研究(13)(14780331)の補助を受けています．

## 参　考　文　献

[1] http://www.google.co.jp/

[2] 横路誠司，高橋克己，三浦信幸，島健一，”位置指向の情報の収集，構造化および検索手法”，情報処理学会論文誌，Vol.41，No.7，pp.1987-1998，2000.

[3] 相良毅，有川正俊，坂内正夫，”分散位置参照サービス”，情報処理学会論文誌，Vol.42，No.12，pp.2928-2940，2001.

[4] 相良毅，有川正俊，坂内正夫，”ジオリファレンス情報を用いた空間情報抽出システム”，情報処理学会論文誌データベース，Vol.41，No.SIG6(TOD7)，pp.69-80，2000.