

写真撮影の位置軌跡を利用した旅行支援システム

奥山 幸也[†] 柳井 啓司[†]

[†] 電気通信大学 〒182-8585 東京都調布市調布ヶ丘 1-5-1
E-mail: [†]okuyama-k@mm.cs.uec.ac.jp, ^{††}yanai@cs.uec.ac.jp

あらまし 近年、様々な写真共有サイトの登場により Web 上で写真を管理、利用することが多くなっている。また、GPS 機能付きデバイスの普及によって位置情報付き写真も増大してきている。それらは画像認識や旅行ガイドシステムなどの研究に広く利用されている。そうした研究の多くでは、多数の写真の位置情報を「点」の集合として扱っているのに対して、本研究では、写真の撮影場所を単なる「点」として捉えるのではなく、撮影者の移動した軌跡を利用し「線」として捉えることで、有用な旅行支援システムの実現を目指す。そのシステムは次の二つの機能を備えている。(1) 観光旅行に行く前のユーザを対象として、仮定の観光ルートを提案する「観光ルート推薦」機能、(2) 観光旅行から戻ってきた人を対象とする、その人の移動軌跡を要約し画像を補完する「旅行記録の整理」機能、である。
キーワード 位置情報、画像、旅行

A travel support system using a large number of geotagged photos

Kohya OKUYAMA[†] and Keiji YANAI[†]

[†] The University of Electro-Communications 1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585 Japan
E-mail: [†]okuyama-k@mm.cs.uec.ac.jp, ^{††}yanai@cs.uec.ac.jp

1. はじめに

1.1 背景

近年、GPS 機能とカメラ機能を持った端末の普及によって、写真撮影時にその撮影位置の位置情報を記録することが容易になっている。同時に、Web 上の写真共有サービスである Flickr^(注1)や Panoramio^(注2)なども位置情報付きの画像に対応し、地図上に撮影位置を表示することが可能となっている。そのため、Web 上には位置情報付きの画像が増大してきており、そのような画像を対象とした研究は盛んになっている。また、撮影位置の遷移からユーザの移動軌跡が分かるため、その移動のログを利用した研究も増えてきている。

1.2 目的

本研究では、他のユーザが撮影した画像の位置軌跡のパターンを利用した旅行支援のシステム開発を行う。旅行写真の研究において位置情報は重要である。それは、その場所に特有の被写体、建物やモニュメントがあるということから、多くの写真が撮影される場所は多くの人が興味を持っている建物やモニュメント、風景などの被写体が存在するということであり、そう

した情報が多数の位置情報付き画像を収集することによって明らかになる。そのことから多くの研究がある。本研究では、写真の撮影場所を単なる「点」として捉えるのではなく、撮影者の移動した軌跡を利用し「線」として捉えることで、より有用な旅行支援システムの実現を目指す。

有名なランドマークがいくつかある中でどういう順に回るのがいいのか、という事は時間的な効率や意味的な回る順序が重要なことであり、人気の観光地の場所だけが分かっても解けない問題である。一方、軌跡を使うことでそれぞれの観光地間のつながりや、順序が測れるという利点がある。本研究では、写真が多く撮影された場所を観光地として定義し、軌跡を主要な観光地間の移動順序とする。

軌跡としては GPS ログデータの方が人々の動きや速度が正確に分かるという利点があるが、そうした GPS による移動軌跡データは大量に入手することは困難である。そこで、本研究では同一撮影者による位置情報付き画像の撮影位置変化の時系列の情報を移動軌跡として利用する。

具体的には次のような機能を持ったシステムを提案する。

- (1) 旅行前の支援のための「観光ルート推薦」機能
- (2) 旅行後あるいはルート決定後の支援のための「旅行記録の整理」機能

(注1): <http://www.flickr.com/>

(注2): <http://www.panoramio.com/>

1.2.1 観光ルート推薦

これは、旅行前に利用することを想定する、次のような機能である。

(1) 観光を予定するユーザからの要求(ランドマーク名や観光写真の選択)を受ける。

(2) 他ユーザの観光写真の情報から適したルートを地図上、観光写真で提示する。

観光ルート推薦のイメージを図1で示す。



図1 観光ルート推薦例

この機能は旅行のプランを決めることへ利用ができる。他のユーザの記録から多くの人はどこに行くのか?そこではどんなものがあるのか?という疑問の回答や、画像を提示することで興味を引き出すことができる。

1.2.2 旅行記録の整理

自分の旅行の際のGPSのログや移動したルートを基に他のユーザの写真やそのメタデータで自分の旅行記録を補う機能である。これは、旅行後やルート決定後に使うことを想定する。

ランドマークやモニュメントの写真を表示することで記録がより詳細になる。近隣の行くことができた観光地を紹介することでプラン変更や旅行の反省、次回の参考となる情報を示す。

これは自分で写真を撮影しなくてもGPSなどで軌跡を保持しておけば写真を補い軌跡を単純化することができる機能である。イメージとして図2を示す。図2の左がユーザの旅行記録で、右が他のユーザのデータを利用し整理した例である。



図2 旅行記録を整理する例

2. 関連研究

GPSのログデータから人の移動軌跡、行動をモデル化し、それに基づき観光地や観光ルートを推論するY. Zhengらの研

究[1]がある。GPSデータを直接扱うので、ユーザの行動軌跡は精密にとれるが、GPSデータがまだWeb上に少なく、データ量があまり取れないという問題がある。

位置情報付き画像を利用して旅行のアシストをする研究もあり、L. Caoらの研究[2]では位置情報付き画像を利用した世界規模での観光地の推薦システムがある。このシステムはユーザが観光地に関連するキーワード、Beach, Diving, Mountainなどを入力すると、各観光地の画像クラスタ内で選抜された代表画像から、キーワードに関連する代表画像を提示し、観光地を推薦する。この推薦では広い範囲での推薦、(Santa Maria, Virginia Beachのような)が返ってくるので、本研究が目指すユーザの移動軌跡を利用した、例えば「京都内の観光ルート」などは推薦ができない。

また、Xin Luらの研究[3]では観光予定のルート生成に位置情報付き画像を利用している。複数のユーザの撮影軌跡を組み合わせることで、観光地での滞在時間を考慮したルートを生成することを可能にしている。このルート生成のためのユーザからの要求には画像を利用することや、観光地内のランドマークや観光名所に関する情報がないという点、軌跡をクラスタリングせずに扱う点で本研究とは異なる。

Y. Araseらの研究[4]ではユーザのある一連の写真集をイベント分類して、そのイベントをクエリにして旅行を推薦している。しかし、[2]と同様に広い規模での旅行を推薦しており、都市の中でのランドマーク間の細かいルートを提示するという事はできない点で本研究とは異なっている。

3. 手法

3.1 データ収集

FlickrのWeb APIを利用してデータを集める。余計なデータやノイズになるデータを除去する。

3.1.1 Flickr APIについて

Flickr API^(注3)とは、Web上の写真共有サービスFlickrが提供しているWeb APIである。Flickr APIを利用するためには、Flickrに登録しAPI Keyを取得する必要がある。Flickrに登録するためには、米国YahooのIDを取得する必要がある。Flickr APIで写真を検索するためには、flickr.photos.searchメソッドを用いる。flickr.photos.searchメソッドを用いることで以下のような写真のメタデータが得られる。

利用したメタデータ

- id 写真にそれぞれつけられたユニークなid。
- owner その写真をアップロードした人の識別id。
- datetaken 撮影日と時間。
- accuracy 16段階のジオタグの精度。値が高いほど精度が高いことを表す。
- latitude, longitude 緯度経度。
- tags 写真を説明する複数の短いテキスト。

(注3): <http://www.flickr.com/services/api/>

3.1.2 画像データの選抜

次に、Flickr API を用いて取得した位置情報付きの画像から本研究での軌跡抽出に適した画像を選抜する。我々のシステムでは、ランドマークや歴史的な建物のようなレベルの観光ルート軌跡を収集する必要がある。そこで、本研究では accuracy が 12 から 16 の画像のみを利用することとする。この 12 から 16 という基準は street レベルを指す精度である。

位置情報には GPS により機械的につけられたものと、撮影者がアップロード後に地図上で撮った位置を付加するものがある。後者の場合、正確な位置である可能性は低くなり、場合によってはある日に撮られた写真すべてに全く同じ位置を付加する場合もある。それを除外するために違う時間に撮られていながら全く同じ位置のデータを除くことにする。

ユーザの移動した軌跡を得ることが重要であるので、それが得られないもの、つまり、一日に一枚しか取られなかったものも除く。

これによる利用できる画像の変化を表 1 にまとめる。

表 1 除外によるデータ数の変化

対象	元のデータ数	除外後のデータ数
ユーザ A	25189	192
ユーザ B	4793	4766

これにより不要なデータを減らすことができ、また、ユーザ A の画像の位置情報の精度はおそらく低軌跡を取ることは難しく、ユーザ B は逆に精度が高く、軌跡が取れる期待が高くなる。

3.2 撮影位置のクラスタリング

撮影位置のクラスタリングは、以下に示す階層的クラスタリング手法によって行う。

(1) すべての写撮影位置のそれぞれをクラスタの位置とする。

(2) もっとも距離の近い二つのクラスタをまとめて新たなクラスタとする。新たなクラスタの位置はクラスタに含まれる撮影位置の重心とする。

(3) クラスタをまとめる作業をもっとも近い距離がある閾値以下になったところでクラスタリングを終了する。

図 3 に階層的クラスタリングの例を示す。

この時、位置は緯度経度で表され、距離 D は地球を完全な球として球面三角法を利用して求める。地点 A, B の緯度経度をそれぞれ $\delta_A, \lambda_A, \delta_B, \lambda_B$ とし、地球の半径を R とすると、球面三角法から、

$$\cos \rho = \sin \delta_A \times \sin \delta_B + \cos \delta_A \times \cos \delta_B \times \cos(\lambda_A - \lambda_B) \quad (1)$$

となり、距離 D は、

$$D = R \times \rho \quad (2)$$

となる。式 2 において ρ の単位はラジアンであることに注意が必要である。

京都近辺において、図 4 にクラスタの中心にマーカを置いた

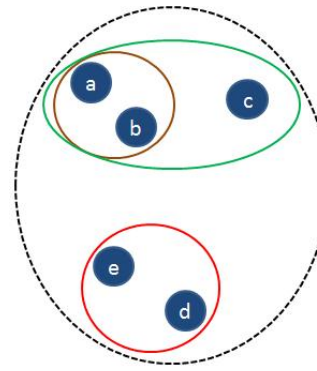


図 3 階層的クラスタリングの例

(1) 今、5つのクラスタ a, b, c, d, e がある。

(2) もっとも近いクラスタ a とクラスタ b を一つのクラスタ ab に統合する。

(3) 4つになったクラスタ間でもっとも近いクラスタ d とクラスタ e を統合しクラスタ de にする。

(4) 残りのクラスタ間でも同様に、クラスタ ab とクラスタ c をクラスタ abc に統合する。

(5) クラスタ間の最短距離がある閾値になるまで続ける。

ものを表示した。クラスタ間の最低距離を 100m から 400m に、この範囲内にあった約 2500 箇所の撮影位置から 100m では約 400ヶ所、400m では約 180ヶ所の観光地のクラスタに絞られていった。



図 4 左：最低距離 100 m の場合の得られた観光地クラスタ。右：最低距離 400 m の場合の得られた観光地クラスタ。

3.3 観光地クラスタの解析

観光地クラスタ内の写真の持つテキストタグと画像データを利用してその観光地クラスタの意味的、視覚的な説明を得るための手法を述べる。

3.3.1 テキストタグによる観光地情報の抽出

テキストタグはそのものを具体的に表現するものもあれば抽象的なものや大雑把なものもある。例えば、金閣寺の写真には「金閣寺、京都、日本、寺」のようなテキストタグが付いたとする。しかし、日本や京都といった広い範囲でつけられるものや、寺という各地につけられるだろうテキストタグはその場所をうまく表現できていない。金閣寺のようなできるだけユニークなテキストタグをつける必要がある。

そこで次のようなスコアづけを行う。あるクラスタ c 内でのテキストタグ t のスコアを $score_{c,t}$ とすると、

$$score_{c,t} = \frac{\text{クラスタ } c \text{ 内のテキストタグ } t \text{ の出現数}}{\text{全体のテキストタグ } t \text{ の出現数}}$$

この $score_{c,t}$ がある閾値以上であればそのクラスタを説明するものとしてクラスタに対応付けを行う。

表 2、表 3 に金閣寺と清水寺を含むクラスタに付加されたテキストタグをスコア付けで処理を行ったものと行わなかったものの例を示す。

このように広く使われる japan や kansai といったものの除去をすることができる。

表 2 金閣寺を含むクラスタのテキストタグの変化

処理前	kinkakuji japanesebuses japan kansai kyoto ofriceandzen
処理後	kinkakuji japanesebuses

表 3 清水寺を含むクラスタのテキストタグの変化

処理前	october japan oktober kyoto kiyomizudera 2009 kiyomizutemple
処理後	kiyomizudera kiyomizutemple

3.3.2 クラスタの代表画像の抽出

画像の視覚的な特徴量を求めクラスタ内の画像の中でもっともそのクラスタを表現していると思われる画像を抽出する。

a) Speeded Up Robust Features

画像の特徴として SURF (Speeded Up Robust Features) [5] を用いる。物体認識に用いる特徴点を求める画像処理のアルゴリズムである。類似する SIFT などよりも高速に求めることができ、照明変化、スケール変化、回転に対して頑健な 64 次元の特徴量である。その頑健さから、観光地での形の定まったオブジェクトやランドマークの特徴点の検出に優れている。

b) 代表画像の選出方法

クラスタ内の各画像間における特徴点マッチングの投票で代用画像を決定する。

各画像において SURF を抽出し、画像それぞれに対して特徴点同士の特徴量がマッチしたら投票を行う。例えば、10 個の特徴点がマッチしたら 10 票得たことにある。他の画像とも投票を行い総得票数がもっとも高いものがクラスタの代表画像となる。

図 5 に画像間の特徴点マッチングの例を示した。二種類の金閣寺の写真間での特徴点のマッチングを示した。

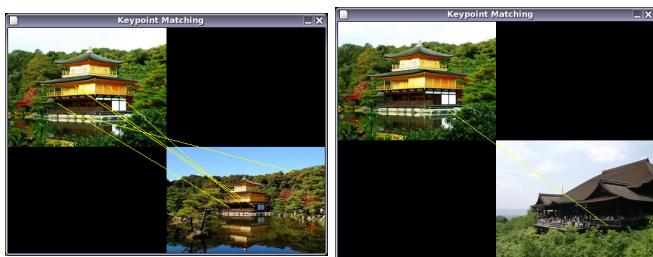


図 5 マッチングの例。左：金閣寺同士。右：金閣寺と清水寺

図 5 左は二パターンの金閣寺の写真間での特徴点マッチングを示した。一方、図 5 右は金閣寺と清水寺という異なるオブジェクトの特徴点マッチングを示した。マッチングしている点の数は金閣寺同士の方が多く、寺という点で似ているが、清水寺のマッチングしている点は少なかった。

このことから、同様のオブジェクトの方がマッチングする数が多い、クラスタに多く含まれるオブジェクトを写した画像がより多くの票を集められることが期待される。

図 6 にクラスタの代表画像例を示す。左が二条城を含むクラスタの代表画像、右が伏見稲荷大社を含むクラスタの代表画像である。それぞれ特徴的な門や鳥居が写っている。



図 6 観光地クラスタの代表画像。左：二条城を含むクラスタの代表画像。右：伏見稲荷大社を含むクラスタの代表画像

4. 軌跡の抽出とモデル化

Flickr から集めたデータからユーザの実際にどのように動いたかをもとめ、クラスタの移動という形でのモデル化を行う。さらに、モデルを増やすために得られたモデルをつなぎ合わせる事で新しいモデルを生成する方法を述べる。

4.1 軌跡の抽出

Flickr から集めたデータのユーザ ID と撮影日時を確認し、あるユーザのある一日に撮影された写真の撮影位置を時系列順に結ぶことで、そのユーザの一日の移動の軌跡を求める。これ以降、この移動軌跡の一つの単位のことを「トリップ」と表す。

図 7 左に、あるユーザの撮影位置の分布例を示す。そしてその点を同一日付内で順に結んだ移動の軌跡例を図 7 右に示す。

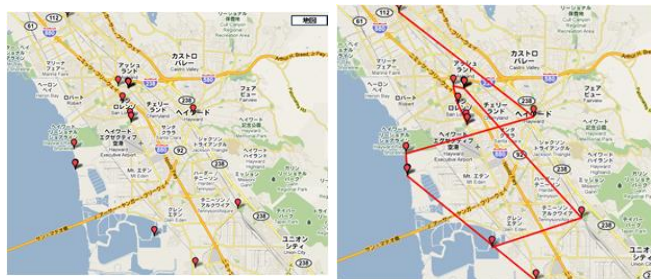


図 7 あるユーザの撮影位置分布 (左) と移動経路 (右)

4.2 軌跡のモデル化

軌跡をそのまま利用するのは難しい。そこで「トリップ」中の各撮影位置を観光地クラスタに割り当てることで「トリップ」のモデル化を行う。このモデルを「トリップモデル」と呼ぶこととする。もしも、連続する撮影位置が同じ観光地クラスタに割り当てられた場合は、その場所に留まっているとし、その移動は無効として扱う。

具体例を図 8 に示す。

4.3 仮想的なトリップモデルの生成

「トリップモデル」を作ったことで軌跡が扱えるようになった。しかし、実際にユーザが動いた軌跡だけでは存在しない経路がある。ここではすでにある「トリップモデル」を基にその組み合わせで新しい「トリップモデル」を作成する。

その手法を以下に示し、図 9 で例を示す。

(1) 同じ観光地クラスタを持つ二つの「トリップモデル」を見つける。

(2) 同一の観光地クラスタを中心に、片方の「トリップモデル」の前半の経路と、もう一方の後半の経路を繋げる。



図 8 軌跡のモデル化の視覚的な例

(3) 新しい「トリップモデル」に重複する観光地クラスタがなければ、この「トリップモデル」登録する。
 重複する観光地クラスタがあるとその「トリップモデル」を繋ぐことで無限に経路が延び続ける事があるため除外している。

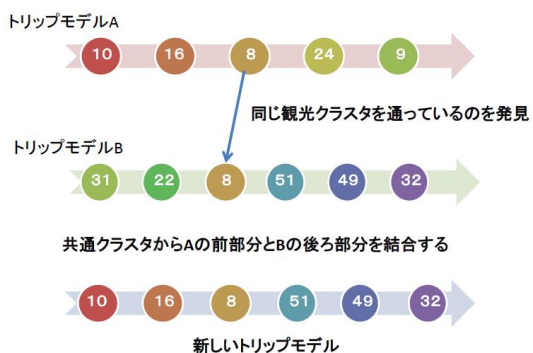


図 9 新しい「トリップモデル」の生成例

図 9 では新しい「トリップモデル」は一つしか示していないが、「トリップモデル B」の前半と「トリップモデル A」の後半を結ぶことでもう一つ作成することができる。または「トリップモデル A」の前半と「トリップモデル B」の前半を繋げることや逆向きにすることも考えられる、しかし、本研究では時間に逆らう向きで結んだ「トリップモデル」は作成しない。なぜなら、そのルートを通ってきた人は何らかの意図や思惑があった可能性があるからである。

例えば、花見をした人の経路がスーパーで買い出しをしてから公園に向かったとする、これが逆では意味が全く違ってくる。この経路を提案された人は飲まず食わずで花見をして、帰りにスーパーで買い出し家でテレビを見ながらそれを食べることになる。また、あるいはデートの経路を求める人が、あるユーザーのデートの経路を逆にたどったとしたら、その印象や結果は同じにはならない。

したがって本研究では経路順を時系列に沿うようにして生成する事にした。

4.4 ユーザに提供する機能

本研究の目的となるシステムの二つの機能について記述する。

4.4.1 観光ルート推薦

この機能は次の 3 つのステップからなる。例を図 10 に示す。

a) ユーザの入力

ユーザには観光地クラスタの解析によって得られた各観光地クラスタのテキストタグや代表画像、また Yahoo! API^(注4)のローカルサーチを利用して住所および近隣の施設情報を提示する。そこでユーザの興味を持ったいくつかの観光地クラスタを選択させる。

b) 候補「トリップモデル」の検索

ユーザが要求した観光地クラスタ集合が「トリップモデル」にすべて含まれていればその「トリップモデル」は候補の経路とする。この候補の検索にはあらかじめ観光地クラスタをインデックスとして「トリップモデル」を管理することですばやい候補経路の抽出が可能になっている。

c) 候補経路の表示

経路の表示には Google Maps API^(注5)を利用する。

Google Maps API により地図を表示し、観光地クラスタの場所(緯度経度)を順番に格納した配列を作り、それをリクエストすることによりマーカを立て、さらにルート案内の API により結び、ルートの記述を得る事ができる。

4.4.2 旅行記録の整理

以下のステップで動作する。例を図 11 に示す。

- (1) あるユーザーの移動軌跡を読み込む。
- (2) トリップからトリップモデルを作成すると同様に、作成した観光地クラスタ間の移動に置き換える。クラスタの代表画像をユーザ移動先の情報として付加し、近隣の観光地クラスタの代表画像も表示する。
- (3) ユーザの通ったとされる観光地クラスタの情報を表示する。

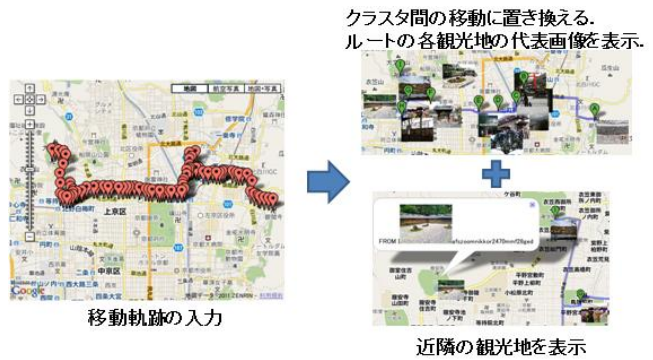


図 11 旅行記録の整理例

5. 実 験

5.1 データセット

データセットは Flickr の Web API を利用することで集めた位置情報付きの画像約 200 万枚の情報を用いた。

位置情報付き画像の分布には偏りがあり、軌跡を得るためにはある程度の画像数が必要になる。そこで今回はある程度の観光地クラスタ間の軌跡が得られた「京都」付近に絞って実験を

(注4): <http://developer.yahoo.co.jp/webapi/map/>
 (注5): <http://code.google.com/intl/ja/apis/maps/>



図 10 観光ルート推薦例

行った．図 12 に京都駅付近の観光地クラスタ間のつながりの様子を示した．



図 12 京都の観光地間につながり

京都付近の写真をアップロードしてた 162 人のユーザ, 1,805ヶ所の撮影位置を閾値 400m でクラスタリングを行い, 154 の観光地クラスタを得れ, 18,752 の重複のないルートを導出した．

5.2 提示される経路の距離的な比較

「トリップモデル」の性能を調べるためにトリップモデル内の観光クラスタをランダムに組み替えた経路を 10,000 経路生成し, 比較を行った．比較に用いる尺度は経路の隣接する観光地クラスタ間の距離の平均 (km) で行った．これによりクラスタ間の平均移動距離がどうなるかを調べた．

ここでの距離とは式 1, 式 2 によって求められる地球を球として見た時の球表面の二点間の最短距離である．また地球の半径は 6,400km として計算した．観光地間の平均距離を表 4 に示す．表 4 より, ランダムよりも本手法がクラスタ間の平均移

表 4 経路選択による違い

手法	観光地間平均距離 (km)
ランダム組み替え	3.810
本手法	2.512

動距離に長距離は短いことが分かった．

ランダムは行き先は決まっているが, 行く順番が分からないので適当に行くという場合であり, 本手法は同じ場所はめぐりつつ, それの約 66% の移動距離で済むといえる．

単純に距離を短くするように観光地を巡るということに関し
て言えば, トラベリングセールスマン問題を解くことが最良で

あるが, そうしたルートは必ずしも観光に適したものではなく, 実際に人が旅行したルートの方が旅行経路として適切であると我々は考えている．また, 本手法では経路の推薦を行っており, そのような問題解決法では得られない新しい観光地の候補の発見ができる．

5.3 観光ルート推薦の例

本システムの機能の一つ「観光ルート推薦」によるルート推薦例を示す．

図 13 はユーザの選択画面を示す．枠に囲まれた一つ一つが観光地クラスタであり, b) の代表画像選出の際の得票数上位 5 枚の画像とテキストタグの情報, ローカルサーチによる住所と施設の情報が示してある．また, 表示が上にあるクラスタほど多くの撮影位置を含む, 人気の高い観光地クラスタになっている．

図 13 に表示されている 4 つのクラスタを選択した．それらはそれぞれ「京都駅」「稲荷伏見大社」「宇治橋」「二条城」を含んでいる．



図 13 ユーザの選択画面

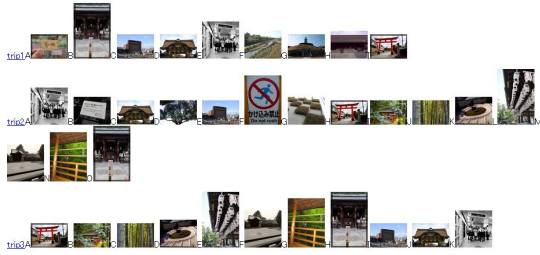


図 14 ルート候補の提示



図 15 trip1 のルート . B:宇治橋, C:京都駅, D:二条城, I:稲荷伏見大社 . 総移動距離 29.8km, 平均区間距離 3.73km



図 16 trip2 のルート . C:二条城, E:京都駅, H:稲荷伏見大社, O:宇治橋 . 総移動距離 52.7km, 平均区間距離 3.76km



図 17 trip3 のルート . A:稲荷伏見大社, H:宇治橋, I:京都駅, J:二条城 . 総移動距離 54.1km, 平均区間距離 5.41km

4つの観光地を通るルートの候補として、図14のような3つのルート (trip1, trip2, trip3) を代表画像を並べた形で提示された。

そして、図14の trip1, trip2, trip3 と書かれたルートを選択するとそれぞれ図15, 図16, 図17のようにマップ上にルートが提示される。これらルートの距離の比較を表5にまとめる。ここでの距離は Google Maps のルート案内 API によって求められた距離である。

表 5 候補ルートの経路距離

候補	観光地数	経路の総和 (km)	観光地間平均距離 (km)
trip1	9	29.8	3.73
trip2	15	52.7	3.76
trip3	11	54.1	5.41

表5より、これらのルート候補を選択する際に、例えば、観光地間であまり移動したくない場合は、trip1 が trip2 を選択するであろうし、できる限り色々な観光地を巡りたいなら、trip2 を選択すればよい。また移動距離はどうでもいいという人ならば、代表画像を眺めて決めることになる。しかし、このルート選択は利用者のそれぞれによって異なり、明確にどのルートが素晴らしいかどうかということを決めることは難しく、ユーザーそれぞれに合わせて検索時の検索オプション (距離的なもの、時間的なもの、観光地数など) で対応するような機能の追加が必要である。

5.4 旅行記録の整理の例

本システムの機能の一つ、「旅行記録の整理」の例を示す。

図18は入力となるユーザのGPSログを用いた移動軌跡の例である。



図 18 GPS データの例

図18の軌跡データ読み込み観光地クラスタに割り振る。また、同じ観光地クラスタに移動した場合はまとめている。

図19に旅行記録の整理したものを示す。図19上の地図は移動軌跡を観光地クラスタの移動として要約したものであり、下の画像列が通って来た観光地の代表画像を示している。.. アルファベットのついたマークは移動の順序を表している。通った観光地クラスタ以外の地図上の写真は、通った観光地クラスタから行くことができる、観光地クラスタを示している。これにより、次に旅行に行くときやプランの変更を考える時の参考となる。

図19で表示した、自分のルート近くの観光地のマークをクリックすると図20のように、その観光地の代表画像と、どの観光地クラスタから行くことができたかを示している。図20

6. おわりに

6.1 まとめ

本研究では、位置情報付きの写真大量の写真を利用し観光旅行を支援する二つの機能を持つシステムを開発した。機能の一つ目は観光旅行に行く前のユーザを対象として、仮定の観光ルートを提案する「観光ルート推薦」機能である。二つ目は観光旅行から戻ってきた人を対象とする、その人の移動軌跡を要約し画像を表示する「旅行記録の整理」機能である。

実現のため、Flickr より位置情報付き画像を約 200 万枚集め、軌跡が多く取れる人気の観光地域「京都」に関して、画像の分布を階層的にクラスタリングすることで観光地の発見とテキストタグや画像の視覚特徴から観光地の特徴発見、ユーザの移動経路から軌跡のモデル化を行った。複数のモデルを時系列を守り連結することで新しい経路を作成した。

6.2 今後の課題

現在は「京都」のデータしか扱っていないので、他の経路が得られる程度の人気のある観光地域のデータも扱う。

経路検索の際に様々な要求を与えられるようなシステムに機能を追加できる。例えば、出発地点やゴール地点を設定することや、観光地数の制限、移動距離、旅行時間による制限、人気観光地、人気ルートによるランク付けなどを考えている。

実際に利用するユーザの使い勝手や、その推薦経路の妥当性や有効性、その経路で旅行がしたいかという主観的な印象などの評価のために被験者によるシステム評価も課題である。

表示する写真に普遍的ではないもの、例えば人物や季節限定のものなどが含まれた場合、ユーザはそれが見たくて観光ルートを決定してしまう可能性もある。したがって人物や動物が主な被写体であったなら写真を除くことや、時期的に偏りを持つ被写体を検出したら代表画像として選出しないなど、代表画像の選び方を工夫することも課題である。

山や巨大な建造物の場合、被写体の位置と撮影位置が大きく異なる可能性があり、撮影場所に行くことでその写真の風景は見られるが、目的がその場所への訪問だった場合、ユーザの要求とズレが生まれてしまう。したがって、被写体との距離関係の考慮を行うことで、それらの問題に対処していく必要がある。

文 献

- [1] Y. Zheng, L. Zhang, X. Xie, and W. Y. Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proc. of ACM World Wide Web Conference*, 2009.
- [2] L. Cao, J. Luo, A. Gallagher, X. Jin, J. Han, and T.S. Huang. A worldwide tourism recommendation system based on geotagged web photos. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2010.
- [3] X. Lu, C. Wang, J.M. Yang, Y. Pang, and L. Zhang. Photo2trip: Generating travel routes from geo-tagged photos for trip planning. In *Proc. of ACM International Conference Multimedia*, 2010.
- [4] Y. Arase, X. Xie, T. Hara, and S. Nishio. Mining people's trips from large scale geo-tagged photos. In *Proc. of ACM International Conference Multimedia*, 2010.
- [5] B. Herbert, E. Andreas, T. Tinne, and G. Luc. Surf: Speeded up robust features. *Computer Vision and Image Understanding*, pp. 346–359, 2008.



図 19 上: 図 18 のルートを整理したもの. 通った観光地クラスタの代表画像と近隣の観光地クラスタの代表画像を地図上に表示. 下: 図 18 を整理した観光地クラスタの代表画像列.

左の場合は図 19 の「D」が「E」のマークが指し示す観光地クラスタから行けた, 図 20 右の場合は図 19 の「A」のマークが指し示す観光地クラスタから行けたという意味である。



図 20 そのルートに近い観光地紹介の例. 左: D, E から来れる観光地クラスタ. 右: A から来れる観光地クラスタ

GPS を持たなくとも巡った観光地を地図上にマッピングするだけでも作成が可能となっている。その例を図 21 に示す。図 21 は「京都駅、清水寺、銀閣寺、二条城」を巡った人がマークを置いたものである。



図 21 左: 巡った観光地「京都駅、清水寺、銀閣寺、二条城」にマークを置く. 右: ルート表示:15 の近隣の観光地クラスタを発見.

この例では通った 4ヶ所の観光地クラスタから行くことができた観光地クラスタを 15ヶ所発見した。これにより、もし、他に行くべき観光地を発見したい場合の手助けができる。