

粗密ルート探索による景観アウェアルート推薦システム

川俣 光司^{1,a)} 奥 健太^{1,b)}

概要：本研究では、道路景観に基づいたルートを提示する景観アウェアルート推薦システムを提案する。提案システムでは、出発地と目的地を指定すると田園景観、山林景観、水辺景観、都市景観の4種類の景観ルートを推薦する。本システムでは、探索時の計算量を削減するため、景観ベースのクラスタリング手法、景観クラスタグラフ、粗密ルート探索手法を提案する。実際の道路ネットワークデータを用いて提案システムの性能を評価する。具体的には、生成された景観クラスターの妥当性、道路景観に基づくルート推薦の妥当性、粗ルート探索を用いた時のルート探索時間についての評価を行い、有効性を検証する。

キーワード：ルート推薦システム、地理情報、クラスタリング

KAWAMATA KOJI^{1,a)} OKU KENTA^{1,b)}

1. はじめに

自動車は単なる移動手段だけでなく、ドライブすること自体が娯楽の一つとなっている。ドライブの楽しみ方の一つとして、海沿い景観や田園景観など好きな景観を眺めながら走りたいという要求もある。既存のルート推薦システムは、出発地と目的地が与えられたとき、その間を結ぶ最短ルートや最速ルート、人気ルートを推薦するものが多い[1][2]。しかし、景観を重視したルート推薦システムはあまり見当たらない。

本研究では景観を重視したルートを推薦する景観アウェアルート推薦システムの実現を目指している。例えば、海沿い景観が好きなユーザには海沿い景観を優先したドライブルートを、田園景観が好きなユーザには、田園景観を優先したドライブルートを推薦する。このようなシステムの実現には景観を重視したルート探索手法を開発する必要がある。

我々はこれまでに道路ネットワーク上の道路リンクを景観ベクトル化する手法を提案してきた[3]。我々の先行研究[3]では、予備実験により道路景観において重要な景観要素として、田園系、山林系、水辺系、都市系の4要素を選定した。これら4要素から構成される景観ベクトルを定

義し、道路リンクの景観ベクトルを推定する手法を提案した。本研究では、先行研究[3]で得られた景観ベクトルが付与された道路ネットワークを前提とした景観アウェアルート推薦システムを提案する。

本研究の貢献は以下のとおりである：

- 道路ネットワークにおいて景観クラスタリングを適用することで景観クラスタグラフを作成した。景観クラスタグラフにおいてあらかじめ粗ルート探索を行うことで、ルート探索における処理コストを削減した。
- 景観を重視したルートを提示する景観アウェアルート推薦システムを提案した。提案システムでは、各景観要素を重視した4パターンのルートを提示することでルート選択の多様性を与えている。

2. 関連研究

2.1 ルート推薦システム

最短経路探索アルゴリズムとして、ダイクストラ法[4]やA*アルゴリズム[5]が挙げられる。これらのアルゴリズムでは、入力された始点と終点において、道路ネットワーク上のリンクに付与されたコストに基づき、総コストが最小となるようなルートを選択する。

最速経路探索[1]は、リンクの距離ではなく、リンクを通過するための旅行時間に着目し、旅行時間が最小となるようなルートを選択する。Weiら[1]は、GPSの軌跡データから速度パターンをマイニングすることで旅行時間を推定している。

¹ 龍谷大学
1-5 Yokotani, Seta Oe-cho, Otsu, Shiga, 520-2194, Japan
^{a)} t18m057@mail.ryukoku.ac.jp
^{b)} okukenta@rins.ryukoku.ac.jp

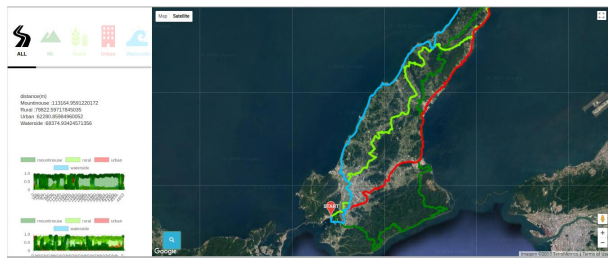


図 1 システムインターフェース。コントロールビューとマップビューから構成される。「Search」ボタンをクリックすることで、出発地から目的地を通る推薦ルートとして、田園景観、山林景観、水辺景観、都市景観の各景観要素をそれぞれ重視したルートが提示される。

人気ルート推薦は、多くの人々が関心をもっているルートを推薦する手法である。Wei ら [2] は、ユーザの軌跡データから多くのユーザが関心をもつルートをマイニングすることで、人気ルートの抽出を行っている。

個人の嗜好に応じたルートを推薦する、個人化ルート推薦の研究もある。MyRoute[6] は、ユーザが熟知しているルートやランドマークに基づき、ユーザ個人向けのルートを作成している。MyRoute では、ランドマークをユーザ自身で入力する必要があるのに対し、Going My Way[7] では、ユーザの個人的な GPS ログデータから、ランドマークを自動的に特定している。

以上のようにこれまでに多くのルート推薦手法が提案されているが、我々の調査した限りでは、景観を考慮したドライブルート推薦手法は少ない。景観を考慮したドライブルート推薦として、Niaraki ら [8] の研究がある。この研究では、オントロジーにより道路属性を定義している。ドライブ景観も属性の一つとして定義されているため、この属性を用いることでドライブ景観を考慮したルート推薦を可能にしている。しかしながら、オントロジーのフレームワークについては詳細に説明されているものの、オントロジーの構築方法については述べられていない。

3. 提案システムの概要

本章では、提案システムである景観アウェアルート推薦システムの概要を説明する。インターフェースおよびシステム構成について説明する。

3.1 インタフェース

図 1 と図 2 に本システムのインターフェースを示す。図 1 よりインターフェースは、大きく分けてコントロールビューとマップビューから構成される。ユーザはマップビューにマーカーを置くことで出発地および目的地を入力する。マップビューの左下の「Search」ボタンをクリックすることで、

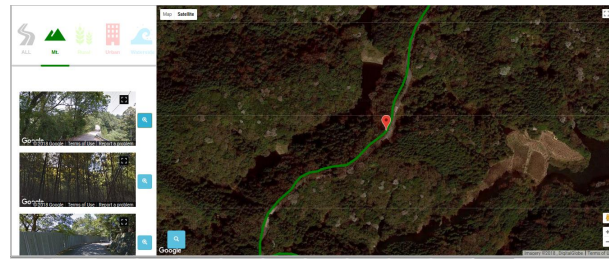


図 2 コントロールビューから好みの景観を選択したときのシステムインターフェース。選択した景観を重視したルートがマップビューに強調表示される。推薦ルート上のマーカーをクリックすると、対象地点の Google ストリートビュー画像が提示される。

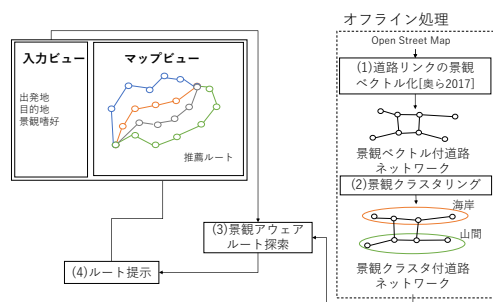


図 3 システム構成。(1) 構成される道路ネットワークデータの各道路片について景観ベクトル化を行う。(2) 景観ベクトル付き道路ネットワークデータより、景観ベクトルに基づいた景観クラスタリングを行う。(3) 景観クラスタ付き道路ネットワークデータにおいて、入力要求（出発地、目的地）を満たす各景観ルートを探査する。(4) 各景観ルートをマップビューに提示する。

出発地から目的地を通る推薦ルートとして 4 パターンのルートがマップビューに提示される。各パターンの推薦ルートは、田園景観、山林景観、水辺景観、都市景観の各景観要素をそれぞれ重視したルートである。また、コントロールビューに各ルートの距離と始点から終点までの景観ベクトルの変動のグラフを提示している。

推薦ルート提示後、コントロールビューから好みの景観を選択することで、その景観を重視したルートが強調表示される。さらに、その推薦ルート上に等間隔にマーカーが提示される。ユーザがこのマーカーをクリックすると、対象地点の Google ストリートビュー画像が提示される。図 2 より、コントロールビューにもマーカーの Google ストリートビュー画像を提示しており、その横に配置しているボタンをクリックすることによりマップビューよりその地点をズームして閲覧することができる。

3.2 システム構成

図 3 にシステム構成図を示す。以下、各処理について説

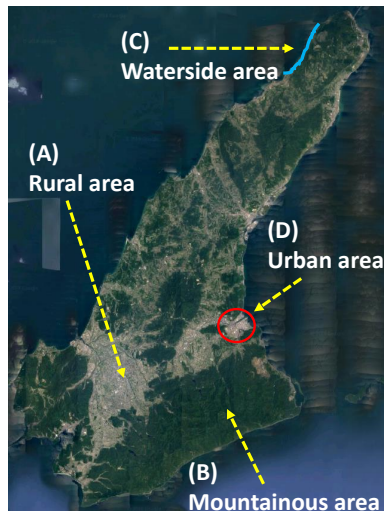


図 4 淡路島の航空画像。A は田園領域，B は山林領域，C は水辺領域，D は都市領域となっている。

明する。項目番号は図中の番号に対応する：

- (1) OpenStreetMap の道路データをノードとリンクから構成される道路ネットワークデータに変換する。各道路リンクについて景観ベクトル化を行い、景観ベクトル付き道路ネットワークデータを得る。
- (2) 景観ベクトル付き道路ネットワークデータより、景観ベクトルに基づいた景観クラスタリングを行い、景観クラスタ付き道路ネットワークデータを得る。
- (3) 与えられた景観クラスタ付き道路ネットワークデータにおいて、入力要求（出発地、目的地）を満たす各景観ルートを探査し、ルート集合を取得する。
- (4) 各景観ルートを推薦ルートとして、マップビューに提示する。また、入力ビューからの景観の選択より、ルートを選択された景観ルートに絞り、Google ストリートビューの画像をマーカーに提示する。

ここで、(1) および (2) の処理については、入力に依存しないためオフライン処理が可能である。なお、(1) の景観ベクトルについては、奥ら [3] の手法を用いる。本研究において、主として取り組む課題は (2) および (3)、(4) となる。これらについては、4 章において詳細に述べる。

4. 景観ルート探索

ダイクストラ法 [4] のような従来のルート探索手法では、道路リンク一つ一つに付与されたコストに基づき、コストの総和が最小となるようなルートが選択される。景観ルート探索において、最も単純なアプローチは、重視する景観要素が強い道路リンクのコストを下げて従来のルート探索手法を適用することである。しかしながら、膨大な道路ネットワークデータに対し、このような探索手法を適用することは計算量が膨大になるという課題がある。

一方で、道路景観は断片的に存在するものではなく、ある一定の領域で構成されるという特性をもつ。例えば、図

4 の例では、A の領域が田園系領域、B の領域が山系領域といったように、類似する景観要素で構成される領域が存在する。このような特性を踏まえ、あらかじめ類似する景観領域を集約しておくことで、計算量の削減が見込まれる。

そこで、提案手法では、まずクラスタリング手法を適用し類似する景観要素で構成される領域（クラスタ）を抽出する。つづいて、抽出されたクラスタ間の接続を考慮した隣接行列を作成することで景観クラスタグラフを作成する。まず景観クラスタグラフにおいて、各景観要素を重視した 4 パターンの大まかなルートを探査—粗ルート探索とよぶ—したうえで、各パターンの詳細なルート探索—密ルート探索とよぶ—を行う。

4.1 定義

定義 1：道路ネットワーク。 道路ネットワークは有向重み付きグラフ $G = (V, E)$ で表現される。ここで、 V は道路ノード集合であり、 $E \subseteq V \times V$ は道路リンク集合である。道路ノード $v_i \in V$ は交差点や道路の終端を表す。道路リンク $e_k = (v_i, v_j) \in E$ は、始点ノード v_i から終点ノード v_j へ向かう有向リンクである。本稿では簡略化のため、 e_k の始点ノードを $e_k.s$ 、終点ノードを $e_k.d$ として表す。また、道路リンク e_k にはリンクの距離等に応じたコスト w_k が付与されている。

定義 2：景観ベクトル。 景観ベクトルは、奥らの先行研究 [3] で定義した 4 種類の景観要素（田園系 (r)、山林系 (m)、水辺系 (w)、都市系 (u)）から構成される 4 次元の確率ベクトルとして定義される。ベクトルの各要素はその景観要素が含まれる確率を表す。したがって、全要素の総和は 1 となる。道路リンク e_i の景観ベクトルを $s(e_i)$ と定義し、 $s(e_i) = (s_i^r, s_i^m, s_i^w, s_i^u)$ と表す。

定義 3：景観クラスタ。 景観クラスタ $C_k \in \mathcal{C}$ は景観ベクトルが類似する道路リンクの集合で表される。景観クラスタ C_k の景観ベクトル $s(C_k)$ は、そのクラスタに属する道路リンクの景観ベクトルの平均ベクトルで表す。すなわち次式で定義する：

$$s(C_k) = \frac{1}{|C_k|} \sum_{i \in C_k} s(e_i). \quad (1)$$

ここで、 $|C_k|$ は景観クラスタ C_k に属する道路リンクの数を表す。

定義 4：景観クラスタグラフ。 景観クラスタグラフは有向重み付きグラフ $G = (\mathcal{V}, \mathcal{E})$ で表現される。ここで、 $\mathcal{V}$ は景観クラスタ C_i の集合であり、 $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ は景観クラスタ間のリンク集合である。リンク $l_k = (C_i, C_k) \in \mathcal{E}$ は、景観クラスタ C_i から C_k へ向かう有向リンクである。また、リンク l_k には終点の景観クラスタ C_k の景観ベクトル $s(C_k)$ に応じたコストベクトル $\omega_k = (\omega_k^r, \omega_k^m, \omega_k^w, \omega_k^u)$ が付与されている。 ω_k の各

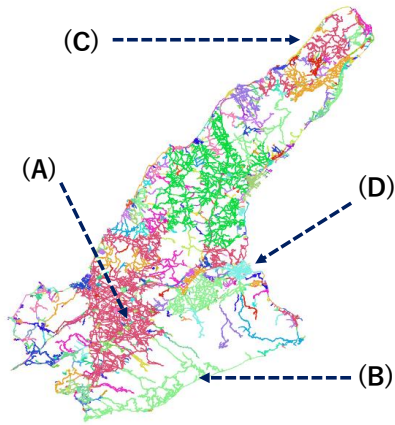


図 5 景観クラスタリングを適用した淡路島の道路ネットワーク。

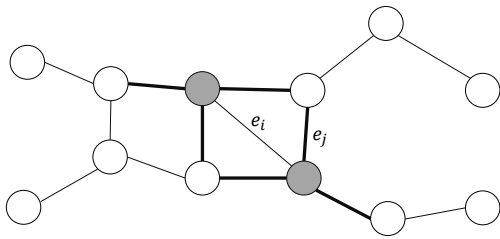


図 6 景観に基づくクラスタリング。

要素は、ルート探索時に重視する景観ごとのコストを表す。例えば ω_k^r は田園景観を重視したルート探索を実行する際に参照されるコストである。

定義 5：景観ベクトルのクラスタ内類似度。 景観クラスタ C_k の景観ベクトルのクラスタ内類似度を $\text{intra_sim}(C_k)$ で表される。 C_k のすべての道路リンク対に対してコサイン類似度を用いて $s(e_j)$ よりクラスタ内類似度を求め全クラスタについてクラスタ内平均類似度を求める。また、クラスタに含まれるリンク数より重みを与える。景観ベクトルのクラスタ内類似度は次式で定義する：

$$\text{intra_sim}(C_k) = \frac{1}{n|C_k|} \sum_{i \in C_k} \sum_{k \in C_k} \cos(s(l_i), s(l_k)). \quad (2)$$

ここで、 n は全リンク数を表す。 $\cos(s(e_j), s(l_m))$ は次式で算出される：

$$\cos(s(l_i), s(l_k)) = \frac{s(l_i) \cdot s(l_k)}{|s(l_i)| |s(l_k)|} \quad (3)$$

4.2 景観クラスタグラフの作成

4.2.1 景観クラスタリング

与えられた道路ネットワークにおいて、道路リンクの近接性および景観ベクトルの類似性に基づき景観クラスタを形成する。隣接する道路リンク間において、各道路リンクの景観ベクトルの類似度が閾値以上である場合、それらの道路リンクを同一のクラスタとみなす。図 5 に淡路島の道

路ネットワークに対し、景観クラスタリングを適用した結果を示す。図 4 の A の領域が田園系領域、B の領域が山林系領域として抽出されている。

また、景観クラスタリングの疑似コードをアルゴリズム 1 に示す。以下、図 6 およびアルゴリズム 1 に沿って景観クラスタリングの処理を説明する。

Algorithm 1 景観クラスタリング

Require: Target link e_i , Cluster ID k

```

1: function ROADScapeCLUSTERING( $e_i, k$ )
2:   Cluster ID of  $e_i \leftarrow k$ 
3:   linkList  $\leftarrow$  getLink( $e_i$ ): Get links adjacent to  $e_i$ .
4:   for each  $e_j$  in linkList
5:     if Cluster ID of  $e_j = 0$  then
6:       if  $\cos(s(e_i), s(e_j)) \geq \alpha$  then
7:         roadscapeClustering( $e_j, k$ )
8:       end if
9:     end if
10:  end for
11:  return 0
12: end function

```

道路ネットワーク上のリンクの中からランダムにリンクを一つ選択する。その選択されたリンクを e_i とする。 e_i と接続されているリンクを e_j とする。それぞれの景観ベクトルを $s(e_i), s(e_j)$ とする。図 6 は e_i と e_j の関係性を示している。さらに、 $s(e_i)$ と $s(e_j)$ は、それぞれのリンクの景観ベクトルとする。

提案手法であるクラスタリングアルゴリズムは、 $\text{roadscapeClustering}((e_i, k))$ という形で呼び出される。まず、 e_i のクラスタ ID として k を付加する。次に、 e_i に隣接するすべてのリンクを取得し、xcode linkList に設定する。 $e_j \in \text{linkList}$ のリンクごとに以下のプロセスを実行する。クラスタ ID が付加されていない場合、 $\cos(s(e_i), s(e_j))$ (式 (3)) を算出する。

$\cos(s(e_i), s(e_j))$ が閾値 α 以上の場合、 e_j のクラスタ ID として、 e_i と同一のクラスタ id k を付加する。 $\text{roadscapeClustering}((e_j, k))$ を再帰的に呼び出す。上記を道路ネットワークにあるすべてのリンクに対しクラスタ ID が付加されるまで繰り返す。

以上により得られた景観クラスタを $C_k \in \mathcal{C}$ とする。また、式 (1) により、景観クラスタの景観ベクトル $s(C_k)$ を算出する。

4.2.2 景観クラスタ間の隣接関係の抽出

景観クラスタ抽出後、景観クラスタ間の隣接行列を作成し、景観クラスタグラフを作成する。景観クラスタ間の隣接行列は、 $|C| \times |C|$ の行列 $\mathbf{A} = [a_{ij}]_{|C| \times |C|}$ で表す。ここで、 $a_{ij} = 1$ のとき景観クラスタ C_i から C_k へ向かうルートが存在することを表し、 $a_{ij} = 0$ のときそれが存在しないことを表す。図 7 は淡路島の道路ネットワークに対し作成された景観クラスタグラフである。ここで、景観クラ

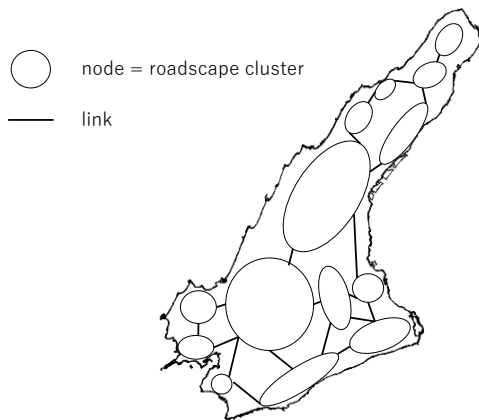


図 7 景観クラスタグラフ.

タグラフにおけるノードが景観クラスタ、リンクがクラスタ間の隣接関係を表す。以下に隣接行列作成手順を示す。

- (1) リンク id の順に中心リンク l_i として設定する。
- (2) 中心リンクと接続しているノードを用いて、ノードに接続している中心リンク以外のリンク l_j を抽出する。
- (3) 中心リンクのクラスタと抽出したリンクのクラスタを比較し、クラスタ id が異なった場合は、隣接するクラスタとして隣接行列 $\mathbf{A}$ に $a_{ij} = 1$ とする。
- (4) 中心リンクを次のリンクに設定し、2~4 をリンク分繰り返す。

4.2.3 景観クラスタグラフへのコストの付与

次節で述べる粗ルート探索を実行するために、あらかじめ景観クラスタグラフのリンクにコストを付与する。リンクのコストはリンクの終点の景観クラスタの景観ベクトルを基に算出する。重視する景観要素に応じて、終点の景観クラスタにおいて、その景観要素が強ければコストが低く、要素が弱ければコストが高くなるように設定する。例えば、田園要素を重視した場合、次に向かう先の景観クラスタの田園要素が強ければコストを低くし、田園要素が弱ければコストを高くする。このようにコストを設定することで、ルート探索時には田園要素が強い景観クラスタへのルートが選ばれやすくなる。

リンク $l_k = (C_i, C_k)$ のコストベクトル ω_k は次式で算出する：

$$\omega_k = d_k(1 - s(C_k)^n), \quad (4)$$

ここで、 d_k はリンク l_k の距離である。

4.3 粗ルート探索

クラスタ間隣接行列とコストベクトルを用いて、ダイクストラ法より各景観の粗ルートの出力方法を説明する。出発地と目的地の位置情報からクラスタを取得し、取得した各クラスタを出発地と目的地にする。クラスタ間のリンクには 4.2.3 節のコストベクトルをコストとして付与している。このコストベクトルは各景観ごとに変動するため、

表 1 α を変化させたときの景観クラスタリング結果。各 α において生成されたクラスタ数、1 クラスタあたりの平均所属リンク数、景観ベクトルのクラスタ内類似度の平均値を示す。

類似度	クラスタ数	平均所属リンク数	クラスタ内類似度
0.95	4896	43.3	0.989
0.90	2912	72.8	0.980
0.85	2111	100.5	0.972
0.80	1538	137.9	0.962
0.75	1158	183.1	0.939
0.70	1094	193.8	0.948
0.65	718	295.3	0.833

ルート探索は各景観ごとで行う。各景観ごとにダイクストラ法を用いて最短ルートを探索する。これにより、景観特徴が表れた 4 パターンの粗ルート探索を行う。

4.4 密ルート探索

粗ルート探索結果より、密ルート探索を行う。リンクは、粗ルート探索の各景観のルート探索結果より、景観クラスタグラフでの通過したノードの景観クラスタに属するリンク、ノードは、リンクと接続しているノードとしている。リンクのコストは、接続されているノード間の距離とした。出発地と目的地のノードより、ダイクストラ法を用いて各景観の最短ルートを算出する。

4.5 ルート提示

密ルート探索より各景観の探索結果のルート提示をする。ルート提示は、密ルート探索結果より出力されたノードから位置情報を抽出し、そのノードの位置情報を探索結果のノード順に直線を引き、GoogleMap 上で提示を行った。

また、ルート探索結果のノードを 6 分割し、その分割される中心のノードにマーカーを置き、Google ストリートビューで探索されたルートの道路景観を見られるようにした。

5. 評価

5.1 データセット

OpenStreetMap の道路ネットワークデータから、淡路島の領域内のみのデータを抽出した。道路ノード数は 102,506 件、道路リンク数は 212,050 件であった。抽出した道路リンクに奥らの先行研究 [3] により景観ベクトルを付加した。

5.2 景観クラスタリングにおける閾値 α の選定

4.2.1 項で述べた景観クラスタリングを実行する際に必要となる景観ベクトルの類似度の閾値 α を選定する。淡路島の道路ネットワークデータに対し、 α を $\alpha = \{0.95, 0.90, 0.85, 0.80, 0.75, 0.70, 0.65\}$ の範囲で変化させながら景観クラスタリングを実行した。

表 1 は、 α を変化させたときのクラスタリングの結果を

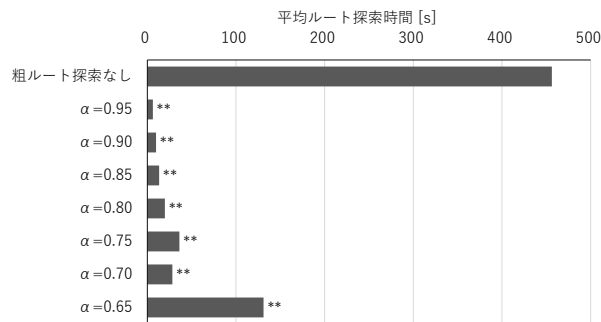


図 8 ルート探索時間の比較. 粗ルート探索を用いないルート探索と粗ルート探索を用いたルート探索の平均ルート探索時間を示している. 粗ルート探索を用いた手法では α を変えたときのルート探索時間を示している. * および ** は粗ルート探索と比較したとき有意差が確認できたことを表す.

示している. 表 1 には, 各 α において生成されたクラスタ数, 1 クラスタあたりの平均所属リンク数, 景観ベクトルのクラスタ内類似度の平均値を示している. α が高いほどクラスタ数が多く生成されていることがわかる. また, $\alpha = 0.95$ のとき景観ベクトルのクラスタ内類似度が最も高くなった. このことから, $\alpha = 0.95$ における各クラスタは景観的に類似する道路リンクにより形成されており, 良いクラスタが生成されているといえる. したがって, 以降の評価では $\alpha = 0.95$ を採用する.

5.3 ルート探索時間の比較: 粗ルート探索あり vs. なし

提案システムでは, 最初から全道路リンクを対象にルート探索を行うのではなく, 事前に景観クラスタリングを適用し, 粗ルート探索を実行することでルート探索時間の削減を行っている. ここでは, 粗ルート探索を導入したときと導入しなかったときとのルート探索時間を比較する.

まず, 出発地点と目的地の組合せとして次の 5 組を用意する:

- (a) (34.257575, 134.722549) $\rightarrow$ (34.574902, 134.959632)
- (b) (34.317774, 134.676412) $\rightarrow$ (34.348304, 134.896255)
- (c) (34.499798, 134.938260) $\rightarrow$ (34.293801, 134.788816)
- (d) (34.545838, 134.923368) $\rightarrow$ (34.440009, 134.912038)
- (e) (34.208185, 134.814500) $\rightarrow$ (34.430861, 134.830634)

各組について, 各景観要素を重視したルート探索を実行し, ルート探索時間を計測する. これを 1 試行とし, 各組 10 回試行し, 1 試行あたりの平均ルート探索時間を算出する.

ルート探索アルゴリズムは Java で実装し, 道路ネットワークデータは PostgreSQL 9.5 で管理している. 実験は Intel Core i5-6200U CPU (2.8GHz), 8GB メモリ, Linux Mint 18.2 を搭載したコンピュータ上で実施した.

図 8 に各手法によるルート探索時間を示す. 図 8 には, 粗ルート探索を用いないルート探索と粗ルート探索を用いたルート探索の平均ルート探索時間を示している. 粗ルー

ト探索を用いた手法では α を変えたときのルート探索時間を示している. 図 8 中の * および ** は粗ルート探索と比較したとき, 対応のある t 検定 (片側検定) による有意差 ($p < 0.05, p < 0.01$) が確認できたことを表す.

図 8 より, 粗ルート探索を用いることによりルート探索時間を短縮できることがわかる. また, 景観クラスタリングの α が高いほど, ルート探索時間が短くなった. 特に, $\alpha = 0.95$ のとき探索時間は 6.24 秒であった. 粗ルート探索を用いないときの探索時間が 456 秒であったことと比べると, 大幅な時間短縮が実現できたといえる.

6. おわりに

本研究では, 景観を重視したルートを探る景観ウェアルート推薦システムを提案した. 提案システムでは, 各景観要素を重視した 4 パターンのルートを提示することでルート選択の多様性を与える.

OpenStreetMap から抽出された淡路島領域内の道路ネットワークデータ (道路ノード 102,506 件, 道路リンク 212,050 件) を用いた評価実験を行った. 粗ルート探索を導入することで, ルート探索時間の大幅な削減を実現した.

謝辞

本研究は JSPS 科研費 JP15K12151, JP16HO5932 の助成を受けたものです. ここに記して謝意を表します.

参考文献

- [1] L. Wei, W. Peng, C. Lin, C. Jung. Exploring Spatio-Temporal Features. In *Advances in Spatial and Temporal Databases, Lecture Notes in Computer Science*, pp. 399–404, 2009.
- [2] L. Wei, W. Peng, B. Chen, T. Lin. Eleventh International Conference on Mobile Data Management PATS : A Framework of Pattern-Aware Trajectory Search. In *Mobile Data Management*, pp. 362–377, 2010.
- [3] 奥健太, 山西良典. 土地被覆図からの景観要素抽出に基づく道路リンクの景観ベクトル化. 情報処理学会研究報告, 第 2017-DBS-1 巻, pp. 1–6, 2017.
- [4] E. Dijkstra. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, Vol. 1, pp. 269–271, 1959.
- [5] R. Dechter, J. Pearl. Generalized best-first search strategies and the optimality of A*. *Journal of the ACM*, Vol. 32, No. 3, pp. 505–536, jul 1985.
- [6] K. Patel, M. Chen, I. Smith, J. Landay. Personalizing Routes. In *UIST 2006: Proceedings of the 19th annual ACM symposium on User interface software and technology*, pp. 187–190, 2006.
- [7] J. Chung, C. Schmandt. Going My Way : A User-aware Route Planner. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1899–1902, 2009.
- [8] A. Niaraki and K. Kim. Ontology based personalized route planning system using a multi-criteria decision making approach. *Expert Systems with Applications*, Vol. 36, No. 2, pp. 2250–2259, mar 2009.